

- ==, 8, 32, 100
- ?, 17–19, 22, 23, 25, 37, 48, 58, 76
- @SuppressWarnings, 81
- &, 48, 58
- <, 32
- <=, 32
  
- abstract, 170
- abstract class, 127, 132
- abstract data types, 149
- abstract method, 127
- AbstractCollection, 55, 176
- AbstractList, 92, 118, 119, 176
  - add, 92
  - get, 92
  - remove, 92
  - set, 92
- AbstractSequentialList, 118, 119
  - listIterator, 118
  - size, 118
- AbstractSet, 176
- acquire
  - Semaphore, 207
- Ada, xiii
- add
  - AbstractList, 92
  - Collection, 16, 20, 55, 167, 213
  - CopyOnWriteArrayList, 181
  - EnumSet, 182
  - HashSet, 180
  - List, 219, 220
  - ListIterator, 220
  - PriorityQueue, 203
  - Queue, 198, 199
  - Set, 177, 227
- addAll
  - BlockingQueue, 205
  - Collection, 12, 17, 167, 250
  - Collections, 12, 172, 253
  - List, 219
  - Set, 227
- addAllAbsent
  - CopyOnWriteArrayList, 227
- addFirst
  - Deque, 212, 213
- addIfAbsent
  - CopyOnWriteArrayList, 227
- addLast
  - Deque, 212, 213
- addObserver
  - Observable, 141
- allOf
  - EnumSet, 183
- alphabetic ordering, 32, 40
- ampersands, 47
- angle brackets, 4, 12
- Annotation, 101
- annotations, 68, 81, 101
  - reflection of, 99
- Anything Goes Behind Closed Doors, Principle of, 92
- append
  - Appendable, 47
- Appendable, 47, 48, 58
  - append, 47
- argument, 8, 15, 21, 27, 92
- Array, 3
  - asList, 169, 170
  - newInstance, 104
- array, 1, 12, 17, 22, 24, 75, 76, 83, 84, 86, 87, 92, 95–97, 108, 154, 155, 168, 170, 248
  - argument, 88

- as implementation for List, 224
  - component type, 108
  - creation, 83, 97, 116, 118, 120
  - deprecated type, 25, 95
  - store exception, 23, 24, 89
  - subtyping, 24
  - types, 103
  - vs collections, 95
- ArrayBlockingQueue, 208, 209, 214, 217, 225, 226
  - blocked requests, 209
  - constructors, 176, 209
  - FIFO, 209
  - performance, 208, 217
  - weakly consistent iterators, 209
- ArrayDeque, 199, 201, 212, 214–216, 226, 228, 231
  - fail-fast iterators, 214
  - performance, 214
- ArrayList, 5, 12, 15, 62, 66, 75, 76, 84, 91, 92, 96, 102, 118, 154, 155, 157, 158, 162, 172, 208, 209, 224, 226–228
  - capacity, 225
  - constructors, 226
  - fail-fast iterators, 226
  - get, 225
  - initial capacity, 226
  - performance, 225, 226
  - set, 225
- Arrays, 24
  - asList, 12, 94, 95, 158
- ArrayStoreException, 169
- asLifoQueue
  - Collections, 253
- asList
  - Array, 169, 170
  - Arrays, 12, 94, 95, 158
- assert, 3, 13, 174
- AssertionError, 13, 174
- atomic operations, 163, 243, 251
  - compound, 243
- atomic variables, 164
  
- backward compatibility, 61
- big-O notation, 156
- BigDecimal, 32, 33
- binary compatibility, 61, 73, 121, 123, 126
- binary search, 158, 249
- binary tree, 127, 191, 193, 194, 202
- binarySearch
  - Collections, 158, 249
- Bloch, Joshua, 22, 46, 113, 136, 171, 176, 180, 192
- blocking
  - lock, 161
  - queue, 200, 217
- BlockingDeque, 151, 212, 215, 216
- BlockingQueue, 151, 203–205, 212, 215
  - addAll, 205
  - containsAll, 205
  - drainTo, 205
  - implementations, 208
  - offer, 204
  - poll, 204, 205
  - put, 204
  - remainingCapacity, 205
  - remove, 205
  - removeAll, 205
  - retainAll, 205
  - take, 204, 205
- bounds, 1, 31, 76, 122
  - recursive, 35
- boxing, 3, 4, 7, 8, 11, 13, 24
- Bracha, Gilad, xiv
- bridge, 48, 122
- bridge, 48, 50, 51, 123
- bucket, 179, 180
- buffer, 236
- bytecode, 5, 64, 122, 123
  
- C, 81
- C#, 61, 73, 81
- C++, xiii, 6, 81
- cache, 8, 234, 237, 243, 246
- caching, 8
- capture of ?, 28
- CAS, *see* compare-and-swap
- cast-iron guarantee, 5, 86, 95, 105, 114
- casts, 1, 5, 76, 97, 116, 118, 120, 129
- catch, 81, 82, 134
- ceiling
  - NavigableSet, 189
- ceilingEntry
  - NavigableMap, 242
- chained overflow, 179
- Character, 179

- checked collections, 113, 116
  - for security, 115
  - see also* checkedCollection, 254
- checked exceptions, 132, 134
- checked lists, 115
  - see also* checkedList, 254
- checkedCollection
  - Collections, 252
- checkedList
  - Collections, 101, 114, 119, 120, 252
- checkedMap
  - Collections, 252
- checkedSet
  - Collections, 252
- checkedSortedMap
  - Collections, 252
- checkedSortedSet
  - Collections, 252
- circular array, 208, 209, 212, 226
  - performance, 214, 226
- Class, 87, 88, 99, 100, 102, 105, 107, 108
  - getComponentType, 87, 104
  - newInstance, 87, 88, 105
- class
  - anonymous, 56
  - argument, 105
  - browsers, 99
  - cast exception, 79, 80, 91, 95, 96, 101, 114, 116
  - declarations, 67
  - files, 64, 70, 73, 142
    - legacy, 70
  - generic, 73
  - inner, 55
  - literals, 99, 102, 103
  - loader, 100
  - nested, 55, 57
  - nested static, 56, 57
  - outer, 55
  - singleton, 250
  - token, 88, 97, 100, 101, 103, 239
  - wrapper, 71
- ClassCastException, 79, 85
- ClassNotFoundException, 134, 135
- clear
  - Collection, 168
  - Map, 230
- client, 67
- client-side locking, 163, 164, 243
- clone
  - Object, 46, 50
  - without a cast, 51
- close
  - Closeable, 47
- Closeable, 47, 48, 58
  - close, 47
- closest matches, 189
- code bloat, 6
- coercions, 7
- Collection, 15–17, 25, 34, 36, 37, 85, 151, 153, 154, 158, 167, 170, 172, 174, 177, 184, 198, 200, 209, 212, 219, 229, 230, 247, 249
  - add, 16, 20, 55, 167, 213
  - addAll, 12, 17, 167, 250
  - clear, 168
  - contains, 25, 168
  - containsAll, 25, 26, 168, 250
  - isEmpty, 168
  - iterator, 55, 153, 168
  - maximum of, 34
  - no implementations, 176
  - remove, 168, 213
  - removeAll, 168, 250
  - retainAll, 168, 250
  - size, 55, 168, 230
  - synchronizedSet, 195
  - toArray, 84, 85, 88, 96, 97, 165, 168, 169, 228
- collection, 3, 4, 24, 25, 86, 87, 168
  - and thread safety, 159
  - checked, 252
  - classes, 84
  - constructors, 176
  - elements, 153
  - vs array, 95
- Collections, 24, 27, 34, 62, 101, 247, 250, 251, 253, 254
  - addAll, 12, 172, 253
  - asLifoQueue, 253
  - binarySearch, 158, 249
  - checkedCollection, 252
  - checkedList, 101, 114, 119, 120, 252
  - checkedMap, 252
  - checkedSet, 252
  - checkedSortedMap, 252

- checkedSortedSet, 252
- copy, 18, 19, 248
- disjoint, 253
- emptyList, 250
- emptyMap, 250
- emptySet, 250
- fill, 248
- frequency, 253
- indexOfSubList, 249, 250
- lastIndexOfSubList, 249, 250
- max, 42, 48, 122, 249
- min, 42, 249
- nCopies, 250
- newSetFromMap, 253, 254
- replaceAll, 248
- reverse, 27, 247
- reverseOrder, 254
- rotate, 247
- shuffle, 247, 248
- singleton, 232, 250
- singletonList, 250
- singletonMap, 250
- sort, 247, 248
- swap, 247, 248
- synchronizedCollection, 251
- synchronizedList, 251
- synchronizedMap, 243, 251
- synchronizedSet, 251
- synchronizedSortedMap, 251
- synchronizedSortedSet, 251
- unmodifiableCollection, 252
- unmodifiableList, 252
- unmodifiableMap, 252
- unmodifiableSet, 252
- unmodifiableSortedMap, 252
- unmodifiableSortedSet, 252
- Collections Framework, xiv, 5, 12, 15, 25, 26, 32, 34, 42, 62, 64, 66, 84, 85, 88, 118, 149, 151, 155, 158, 162, 168, 170, 177, 178, 180, 190, 197, 203, 208, 214, 219, 224, 229, 232, 238, 243, 252, 254
- Comparable, 12, 31, 33–37, 40, 46, 48, 122, 135, 184, 201, 210, 254
  - compareTo, 31, 32, 36, 48, 49, 122, 123, 132, 186
- Comparator, 31, 37, 40, 42, 127, 132, 184, 186, 192, 200, 201, 240, 243, 249, 254
  - compare, 37, 186, 240
  - reverse, 254
- comparator
  - on lists, 42
- comparator
  - SortedSet, 186
- compare
  - Comparator, 37, 186, 240
- compare-and-swap, 164, 165, 203, 216
- compareTo
  - Comparable, 31, 32, 36, 48, 49, 122, 123, 132, 186
  - Delayed, 210
  - Enum, 46
- comparison, 31
  - anti-symmetric, 33
  - compatible with equality, 33
  - congruence, 33
  - permit, 36
  - prohibit, 36
  - reflexive, 33
  - transitive, 33
- compatibility, 24, 48
- compile time, 113
- compile-time errors, 18, 32, 55, 59, 79, 84, 137
- compiler flag
  - ea, 13
  - enableassertions, 13
  - source 1.4, 67
  - Xbootclasspath, 148
  - Xlint:unchecked, 53
- complementOf
  - EnumSet, 183
- Component, 101
  - getListeners, 101
- component type, 75, 83, 89
- concurrency, 159, 162, 205, 223
  - policy, 161, 162
  - utilities, 197
- concurrent
  - collections, 155, 164
  - modification, 155
- concurrent collections, 164
- Concurrent Programming in Java*, 161
- ConcurrentHashMap, 156, 243, 244, 246

- constructors, 244
  - performance, 244
  - segment locking, 244
  - size, 244
  - weakly consistent iterators, 244
- ConcurrentHashMap thread safety, 244
- ConcurrentLinkedQueue, 156, 165, 203, 217
  - constructors, 203
  - removal, 203
  - size, 203
  - weakly consistent iterators, 203
- ConcurrentMap, 151, 232, 243, 244
  - putIfAbsent, 164, 243
  - remove, 243
  - replace, 244
- ConcurrentModificationException, 154, 155, 162, 165, 174, 190, 210, 220, 251
- ConcurrentNavigableMap, 151, 232, 245
  - keySet, 245
  - range-view methods, 245
- ConcurrentSkipListMap, 156, 165, 193, 245
  - fail-fast iterators, 245
  - get, 245
  - performance, 245
  - put, 245
  - remove, 245
- ConcurrentSkipListSet, 156, 178, 179, 190, 193, 195, 245
  - weakly consistent iterators, 194
- consistent with equals, 32
- constant (time), 157
- Constructor, 107
  - newInstance, 105
  - toGenericString, 49, 107
- constructor, 53, 71
- consumer threads, 212
- contains
  - Collection, 25, 168
  - CopyOnWriteArrayList, 181
  - EnumSet, 182
  - HashSet, 180
  - List, 25
  - PriorityQueue, 203
  - Set, 177
  - TreeSet, 191
- containsAll
  - BlockingQueue, 205
  - Collection, 25, 26, 168, 250
  - List, 25
- containsKey
  - Map, 230
- containsValue
  - Map, 230
- contract, 157, 159, 213
- contravariant subtyping, 23
- conversion constructor, 176, 192, 193
- Converter, 144
- copy
  - Collections, 18, 19, 248
- copy constructor, 176
- copy-on-write, 164, 165
- copyOf
  - EnumSet, 183
- CopyOnWriteArrayList, 155, 164, 181, 223, 227
  - addAllAbsent, 227
  - addIfAbsent, 227
  - constructors, 227
  - snapshot iterators, 227
  - thread safety, 227
- CopyOnWriteArraySet, 164, 177, 181, 182, 195, 227
  - add, 181
  - constructors, 182
  - contains, 181
  - immutable, 181
  - iterators, 182
  - performance, 181
  - snapshot iterators, 182
  - thread safety, 181
- covariant
  - array subtyping, 24
  - arrays, 22, 24, 83
  - return types, 46, 50, 51, 124, 242
  - subtyping, 23
- covariant overriding, *see* covariant, return type
- critical section, 161
- data structure, 127
- debuggers, 99
- defensive copy, 252
- delay time
  - DelayQueue, 210
- Delayed, 210
  - compareTo, 210

- getDelay, 210
- DelayQueue, 165, 210, 211, 216
  - delay time, 210
  - fail-fast iterators, 211
  - performance, 211
  - poll, 210
- delegation, 71, 116, 118, 120, 251
  - see also wrapper, 254
- deprecated type, 97
- Deque, 151, 155, 199, 212–215, 226, 228
  - addFirst, 212, 213
  - addLast, 212, 213
  - descendingIterator, 212
  - getFirst, 214
  - getLast, 214
  - offer, 212
  - offerFirst, 213
  - offerLast, 212, 213
  - peek, 212
  - peekFirst, 212, 214
  - peekLast, 214
  - performance, 216
  - pollFirst, 214
  - pollLast, 214
  - pop, 214
  - push, 212, 213
  - removeFirst, 214
  - removeFirstOccurrence, 212, 213
  - removeLast, 214
  - removeLastOccurrence, 212, 213
- deque, 212, 214
  - concurrent, 212
  - head, 214
  - insertion, 212
  - removal, 212
  - tail, 214
- Dequeue, 253
- dequeueing
  - BlockingQueue, 204
- descendingIterator
  - Deque, 212
- descendingKeySet
  - NavigableMap, 242
- descendingMap
  - NavigableMap, 242
- descendingSet
  - NavigableSet, 190
- design pattern, 1, 46, 127
  - Function, 1, 127, 132
  - Interpreter, 1, 127, 131, 132
  - Proxy, 251
  - Singleton, 46
  - Strategy, 1, 46, 127, 135, 138
  - Subject-Observer, 1, 46, 127, 141, 182
  - Visitor, 1, 127, 128, 130
- Design Patterns*, 46, 127, 251
- disjoint
  - Collections, 253
- dot product, 10
- Double, 15
  - subtype of Number, 16
- double-ended queue, 212
- drainTo
  - BlockingQueue, 205
- ea flag, 13
- Effective Java*, 22, 46, 113, 136, 171, 192
- efficiency, 35, 42, 88, 95, 164, 181, 200
  - see also performance, 254
- element
  - PriorityQueue, 203
  - Queue, 199, 214
- element order, 247
- empty deque, 214
- empty loop technique, 116
- emptyList
  - Collections, 250
- emptyMap
  - Collections, 250
- emptySet
  - Collections, 250
- enableassertions flag, 13
- end
  - of deque, 212
- enqueueing
  - BlockingQueue, 204
- entrySet
  - Map, 230, 231
- Enum, 43, 46, 135, 182, 204
  - compareTo, 46
  - name, 46
  - ordinal, 46, 182
  - toString, 46
- enum, 43, 182, 183, 195, 231, 239
- enumerated constant, 43, 46
- enumerated type, 43, 46, 143, 183

- valueOf, 46
- values, 46
- enumeration class, 46
- EnumMap, 155, 231, 232, 239, 245
  - constructors, 232, 239
  - get, 239
  - iterators not fail-fast, 239
  - put, 239
  - thread safety, 239
  - weakly consistent iterators, 239
- EnumSet, 155, 177, 182, 195, 239
  - add, 182
  - allOf, 183
  - complementOf, 183
  - contains, 182
  - copyOf, 183
  - iterators not fail-first, 183
  - noneOf, 183
  - of, 183
  - range, 183
  - remove, 182
  - snapshot iterators, 183
  - thread safety, 183
  - weakly consistent iterators, 183
- equality, 32, 76–78, 100, 170, 237
  - see also* Object, equals, 254
- equals
  - Object, 32, 100, 186, 201, 236, 237, 240, 253
- erasure, 5, 6, 48, 58, 59, 61, 76, 80, 85, 86, 91, 97, 114, 121–123
  - signature, 58
- Ernst, Erik, xiv
- Error, 132
- error messages, 28
  - syntax, 82
- evolution, 61, 64, 66, 67, 69, 75, 97, 121
- Exception, 132, 134
- exception, 1, 81, 103, 114, 137, 163, 198, 205, 251
  - store, 24
  - array store, 23, 24, 89
  - ArrayStoreException, 169
  - class cast, 79, 80, 91, 95, 96, 101, 114, 116
  - ClassCastException, 79, 85
  - ClassNotFoundException, 134, 135
  - ConcurrentModificationException, 154, 155, 162, 165, 174, 190, 210, 220, 251
  - Exception, 132, 134
  - handling, 81
  - illegal argument, 105
  - IllegalArgumentException, 46
  - IllegalStateException, 163, 213, 221
  - InvocationTargetException, 105
  - NoSuchElement, 35
  - NoSuchElementException, 186, 190, 221, 240
  - NoSuchMethodException, 134, 135
  - NullPointerException, 32, 175, 230, 254
  - RuntimeException, 132, 134
  - StubException, 70
  - UnsupportedOperationException, 70, 159, 230, 252
- exclusive access, 243
- expansion, 6
- explicit representation, 75
- explicit type parameters, 19, 28
- extends, 15, 17, 19, 28, 35–37
  - wildcard, 19, 21–23, 28
- factory methods, 251, 252
- fail-fast iterators, 155, 162, 165
  - ArrayDeque, 214
  - ArrayList, 226
  - ConcurrentSkipListMap, 245
  - DelayQueue, 165, 211
  - HashMap, 233
  - HashSet, 180
  - LinkedHashMap, 235
  - LinkedHashSet, 181
  - LinkedList, 215
  - PriorityBlockingQueue, 209
  - PriorityLockingQueue, 165
  - PriorityQueue, 203
  - TreeMap, 243
  - TreeSet, 190, 193
  - WeakHashMap, 237
- fair scheduling, 209
- Field, 107
- FIFO, *see* First In First Out
- LinkedHashMap, 235

- vs LRU, 235
- fill
  - Collections, 248
- final
  - classes, 171
  - fields, 171
- first
  - SortedSet, 186, 190, 240
- First In First Out, 197, 199, 203, 208, 212, 214, 216, 217, 231
- firstEntry
  - NavigableMap, 241
- firstKey
  - SortedMap, 240
- floor
  - NavigableSet, 189
- floorEntry
  - NavigableMap, 242
- for, 9
- foreach, 1, 3, 4, 9, 10, 13, 35, 153, 154, 174
- Fortran, 155
- frequency
  - Collections, 253
- front
  - of deque, 212
- Fuller, Thomas, 86
  
- Gafter, Neal, xiv, 92
- Gamma, Erich, 46, 127, 251
- garbage collection, 235–237
  - tracing, 236
- generalized abstract data types, 132
- generic
  - algorithms, 118, 247
  - array creation, 84, 90, 94, 95
  - array creation error, 85
  - class, 48, 53, 54, 73, 87, 88, 108
  - client, 62, 67, 70, 73
  - code, 73, 80, 121
  - declarations, 142
  - functions, 103
  - interface, 48, 71, 108
  - library, 62, 64, 73
  - method, 10–12, 27, 34, 40, 67
  - method invocation, 12, 28, 41
  - objects, 73
  - reflection
    - library, 105
    - signature, 122, 144
    - signatures, 25, 69, 70, 141, 142
    - type, 1, 5, 12, 73, 83, 101, 113, 115, 135
    - types, recursive, 138
  - generic array creation error, 83, 84
  - generic array creation warning, 95
  - GenericArrayType, 108
  - generics, xiii, xiv, 1, 3–6, 13, 15, 23–26, 31, 48, 50, 58, 61, 64, 69, 80, 86, 96, 99, 119, 127, 129, 131, 137
    - compilation, 48
    - for reflection, 99, 100
- get
  - AbstractList, 92
  - ArrayList, 225
  - ConcurrentSkipListMap, 245
  - EnumMap, 239
  - HashMap, 233
  - List, 219, 228
  - Map, 230
  - TreeMap, 243
  - WeakReference, 236
- Get and Put Principle, 19, 22, 23, 28, 36, 248
- getClass
  - Object, 87, 99, 100, 102, 105
- getComponentType
  - Class, 87, 104
- getDelay
  - Delayed, 210
- getFirst
  - Deque, 214
- getLast
  - Deque, 214
- getListeners
  - Component, 101
- Goetz, Brian, 161
- graph, 237
- graphical user interface, *see* GUI
- GUI, 159
  
- half-open interval, 187
- Hamming, Richard, xiv
- Hansen, Christian Plesner, xiv
- hash code, 36, 178
- hash function, 178, 179
- hash method, 36
- hash table, 4, 155, 156, 178, 180, 191
  - access by content, 156

- insertion, 156
- iterating, 180
- key, 179
- load, 179
- removal, 156
- value, 179
- HashMap, 154, 156, 179, 232–238
  - constructors, 232, 233
  - fail-fast iterators, 233
  - get, 233
  - performance, 233
  - put, 233
- HashMap
  - load factor, 233
- HashSet, 156, 177–181, 195, 233, 238, 254
  - add, 180
  - constructors, 176, 180
  - contains, 180
  - fail-first iterators, 180
  - iteration, 180
  - not thread-safe, 180
  - remove, 180
  - size, 180
  - unsynchronized, 180
- Hashtable, 161, 251
- hashtable
  - load factor, 180
  - rehashing, 180
- Haskell, xiii, 132
- hasNext
  - Iterator, 9, 153, 220
  - ListIterator, 221
- hasPrevious
  - ListIterator, 220, 221
- head, 197
  - of deque, 212
- headMap
  - NavigableMap, 242
  - SortedMap, 240
- headSet
  - NavigableSet, 189
  - SortedSet, 186, 240
- Helm, Richard, 46, 127, 251
- hierarchy, 191
- higher
  - NavigableSet, 189
- higherEntry
  - NavigableMap, 242
- identity, 100, 237
- IdentityHashMap, 156, 232, 237, 238, 245, 254
  - constructors, 238
  - keys, 237
  - load factor, 239
- Igarashi, Atsushi, xiv
- illegal argument exception, 105
- IllegalArgumentException, 46
- IllegalStateException, 163, 221
- immutable, 185, 250
- implements, 15, 28
- in, 208
- inconsistent state, 161
- Indecent Exposure, Principle of , 75, 84, 89, 91, 92, 95, 97, 103, 105
- indexOf
  - List, 220
- indexOfSubList
  - Collections, 249, 250
- inequality assertion, 8
- inheritance, 116, 119, 120
  - specialize by, 119
  - see also* subclass, 254
- initializer, 30
- inner class, 55
- insertion
  - ConcurrentLinkedQueue, 203
- insertion by position, 155
- instance creation, 28–30
- instance tests, 76, 97, 108, 116, 118, 120
- int, 11
- Integer, xiii, 3, 9, 11, 15, 17–19, 48, 49
  - implements Comparable<Integer>, 35
  - subtype of Number, 16
- interface, 67, 158
- interpreters, 99
- intervals
  - half-open, 186
- invariant subtyping, 23
- InvocationTargetException, 105
- isEmpty
  - Collection, 168
  - Map, 230
- Iterable, 9, 15, 151, 153, 154
  - iterator, 9
- iteration, 155
- Iterator, 9, 55, 79, 153, 154, 159, 220

- hasNext, 9, 153, 220
  - next, 9, 153, 220, 245
  - remove, 9, 10, 153, 182, 208, 220
- iterator, 10, 35, 153, 162, 165, 174, 180
- iterator
  - Collection, 55, 153, 168
  - Iterable, 9
  - Queue, 200
  - SynchronousQueue, 212
- Java
  - 1.0 (version), 99, 162
  - 1.1 (version), 99
  - 1.2 (version), xiii, 162
  - 1.4 (version), 3, 5, 50, 64
  - 5 (version), 5, 50, 62, 64, 67, 68, 96, 97, 100, 153, 154, 164, 174, 184, 203, 214
  - 6 (version), 5, 184, 188, 193, 212, 214, 215, 228, 231, 240
  - vs C#, 73
- Java Concurrency in Practice*, 161
- Java Specialists' Newsletter, The*, 135, 226
- java.awt, 101
- java.lang, 9
- java.lang.ref, 236, 237
- java.util, 9, 12, 149
- java.util.concurrent, 149, 197, 206
- JavaBeans, 99
- Johnson, Ralph, 46, 127, 251
- Kabutz, Heinz, 135, 226, 237
- keys, 182
  - Map, 230
- keySet
  - ConcurrentNavigableMap, 245
  - Map, 230
  - NavigableMap, 242
  - TreeMap, 242
- Knuth, Donald, 156
- last
  - SortedSet, 186, 190
- Last In First Out, 212, 253
- lastEntry
  - NavigableMap, 241
- lastIndexOf
  - List, 220
- lastIndexOfSubList
  - Collection, 249, 250
- lastKey
  - SortedMap, 240
- Lea, Doug, 161
- Least Recently Used, 235
  - vs FIFO, 235
- legacy
  - class, 71, 73, 123, 161
  - client, 62, 64, 73
  - code, 1, 5, 61, 62, 64, 73, 80, 113, 116, 120, 121
  - interface, 71
  - library, 62, 67, 73, 113, 114
  - method, 67
  - object, 73
  - signature, 71
- legacy collections, 161–163
- legacy data type, 168
- library, 5, 61, 64, 67, 68, 89
  - generic reflection, 103
- LIFO, *see* Last In First Out
- lightweight process, 159
- linear (time), 157
- linear probing, 238
- linked list, 4, 55, 155, 180, 193, 214
- LinkedBlockingDeque, 216
  - constructors, 216
  - performance, 216
  - weakly consistent iterators, 216
- LinkedBlockingQueue, 156, 165, 208, 214, 216, 217
  - constructors, 208
  - performance, 217
  - weakly consistent iterators, 208
- LinkedHashMap, 156, 232–235, 246
  - access-ordered, 234
  - constructors, 234
  - fail-fast iterators, 235
  - insertion-ordered, 234, 235
  - iteration orders, 233
  - performance, 246
  - put, 235
  - putAll, 235
  - removeEldestEntry, 234, 235
- LinkedHashSet, 156, 177, 180, 181, 195
  - constructors, 181
  - fail-first iterators, 181
  - not thread-safe, 181

- performance, 181
  - unsynchronized, 181
- LinkedList, 118, 156, 199, 214–216, 227, 228
  - constructors, 215
  - fail-fast iterators, 215
  - thread safety, 215
- List, xiii, 3, 5, 8, 9, 15–19, 23, 27, 54, 58, 75, 77–79, 85, 90, 96, 103, 114, 151, 158, 162, 200, 219, 221, 223, 224, 226–228, 247–249
  - add, 219, 220
  - addAll, 219
  - contains, 25
  - containsAll, 25
  - get, 219, 228
  - implementations of, 224
  - indexOf, 220
  - lastIndexOf, 220
  - operations, 219
  - performance, 224, 227
  - remove, 219
  - set, 219, 227
  - subList, 220
- List
  - LinkedList implementation, 214
- list, xiv, 219
  - adding elements, 220
  - changing contents, 247, 248
  - changing elements, 220
  - iterator, 224
  - linked, 55
  - Iteration, 220
  - positional access, 219
  - range-view, 220, 224
  - search, 220
  - singly-linked, 55
  - traversing, 220
  - traversing in reverse, 220
  - usual ordering, 43
- ListIterator, 159, 220
  - add, 220
  - hasNext, 221
  - hasPrevious, 220, 221
  - listIterator, 220
  - next, 221
  - nextIndex, 220, 221
  - previous, 221
  - previousIndex, 221
  - remove, 221
  - set, 221
- listIterator
  - AbstractSequentialList, 118
  - ListIterator, 220
- Lock, 165
  - unlock, 165
- lock, 161, 163, 243
- lock striping, 243
- lock-free deletion, 194
- lock-free insertion, 194
- locking
  - client-side, 163, 164, 243
- locks, 164
- logarithmic (time), 157
- loop, 10
- loop condition, 10
- lower
  - NavigableSet, 189
- lowerEntry
  - NavigableMap, 242
- LRU, *see* Least Recently Used
- Iteration
  - lists, 220
- Map, 151, 156, 158, 162, 164, 178, 229–231, 237, 242
  - access order, 246
  - adding associations, 229
  - clear, 230
  - containsKey, 230
  - containsValue, 230
  - entrySet, 230, 231
  - get, 230
  - implementations of, 232
  - insertion order, 246
  - isEmpty, 230
  - iterator over view, 230
  - keySet, 230
  - performance, 245, 246
  - put, 229
  - putAll, 229, 232
  - querying contents, 230
  - remove, 230
  - removing associations, 230
  - size, 230
  - thread safety, 243, 245

- unreachable keys, 235
  - values, 230
  - views, 230
- map, xiv
- Map.Entry, 230, 231, 241
  - setValue, 230, 241
- max
  - Collections, 42, 48, 122, 249
- maximum, 40
  - Collection, 34
- memory, 237
- memory locations, 159
- merge sort, 247
  - performance, 248
- Method, 107
- method body, 11
- method signature, 11, 67
- methods, 58
- migration compatibility, 61
- min
  - Collections, 42, 249
- minimum, 40
- ML, xiii, 132
- multicore, 159
- multiple bounds, 47, 58, 122
- multiprocessor, 159
- multithread, 159
- multithreading, 159
- mutually recursive bounds, 35
  
- name
  - Enum, 46
- natural order, 175
  - reverse, 254
- natural ordering, 31, 32, 37, 40, 41, 170, 184, 185, 193, 240, 243
  - consistent with equality, 170
  - reverse of, 41
- navigableKeySet
  - NavigableMap, 242
- NavigableMap, 151, 231, 232, 240, 241
  - ceilingEntry, 242
  - closest matches, 242
  - descendingKeySet, 242
  - descendingMap, 242
  - firstEntry, 241
  - floorEntry, 242
  - headMap, 242
  - higherEntry, 242
  - keySet, 242
  - lastEntry, 241
  - lowerEntry, 242
  - navigableKeySet, 242
  - pollFirstEntry, 241
  - pollLastEntry, 241
  - range views, 242
  - subMap, 242
  - tailMap, 242
- NavigableSet, 177, 178, 184, 185, 187–190, 200, 201
  - ceiling, 189
  - descendingSet, 190
  - floor, 189
  - headSet, 189
  - higher, 189
  - lower, 189
  - pollFirst, 188
  - pollLast, 188
  - reverse order navigation, 190
  - subSet, 189
  - tailSet, 189
  - vs NavigableMap, 241
- NavigableSkipListSet, 253
- nCopies
  - Collections, 250
- nested classes, 55, 57
- new, 28
- newInstance
  - Array, 104
  - Class, 87, 105
  - Constructor, 105
- newSetFromMap
  - Collections, 253, 254
- Newton, Isaac, xiv
- next
  - Iterator, 9, 153, 220, 245
- nextIndex
  - ListIterator, 220, 221
- Node, 56
- noneOf
  - EnumSet, 183
- nonparameterized, 76
- nonreifiable
  - casts, 79
  - type, 78, 79, 89, 95–97, 118
- nonreifiable type, 118

- NoSuchElement, 35
- NoSuchElementException, 186, 190, 221, 240
- NoSuchMethodException, 134, 135
- notifyObservers
  - Observable, 141
- null, 21, 32, 87, 167, 168, 175, 190, 199, 210, 236, 248, 253, 254
- NullPointerException, 32, 175, 230, 254
- Number, 15, 17, 19
- numerical ordering, 31
  
- O*-notation, 156
- Object, xiii, 7, 12, 15, 19, 22, 48, 50, 77, 87, 105, 122, 123, 129, 141, 168, 230
  - clone, 46, 50
  - equals, 32, 100, 186, 201, 236, 237, 240, 253
  - getClass, 87, 99, 100, 102, 105
  - toString, 12, 16–18, 20, 22, 23, 29, 107, 119
- object identity, 73
- object inspectors, 99
- object serialization, 99
- Observable, 141–144
  - addObserver, 141
  - notifyObservers, 141
  - setChanged, 141
- Observer, 141–144
  - update, 141
- Odersky, Martin, xiv
- of
  - EnumSet, 183
- offer
  - BlockingQueue, 204
  - Deque, 212
  - PriorityQueue, 203
  - Queue, 198, 199, 213
- offerFirst
  - Deque, 213
- offerLast
  - Deque, 212, 213
- optional operations, 159
- ordered tree, 4
- ordinal
  - Enum, 46, 182
- outer class, 55
- override, 124
  
- parallel class hierarchies, 135, 137, 141
- parallelism, 159
- parameter, 15
- parameter types, 75
- parameterized
  - collection, 248
  - interface, 48
  - type, 28, 53, 58, 64, 66–68, 76, 84, 90, 101, 114, 116, 118
  - type hierarchy, 141
- ParameterizedType, 108
- parametric polymorphism, xiii
- parametric type, 173
- pattern, *see* design pattern
- peek
  - Deque, 212
  - PriorityQueue, 203
  - Queue, 199, 210, 214
- peekFirst
  - Deque, 212, 214
- peekLast
  - Deque, 214
- performance, 24, 118, 156, 159, 162, 164, 170, 180, 182, 190, 191, 194, 195, 227, 248
  - ArrayBlockingQueue, 208, 217
  - ArrayDeque, 214
  - ArrayList, 225, 226
  - binary search, 249
  - circular array, 214, 226
  - ConcurrentHashMap, 244
  - ConcurrentSkipListMap, 245
  - CopyOnWriteArraySet, 181
  - DelayQueue, 211
  - Deque, 216
  - HashMap, 233
  - LinkedBlockingDeque, 216
  - LinkedBlockingQueue, 217
  - LinkedHashMap, 246
  - LinkedHashSet, 181
  - List, 224, 227
  - Map, 245, 246
  - merge sort, 248
  - PriorityBlockingQueue, 209
  - PriorityQueue, 211
  - Queue, 216
  - Set, 177
  - TreeMap, 243

- WeakHashMap, 237
  - see also* efficiency, 254
- performance cost, 162
- platform compatibility, 61
- Point, 50
- poll
  - BlockingQueue, 204, 205
  - DelayQueue, 210
  - PriorityQueue, 203
  - Queue, 199, 214
- pollFirst
  - Deque, 214
  - NavigableSet, 188
- pollFirstEntry
  - NavigableMap, 241
- pollLast
  - Deque, 214
  - NavigableSet, 188
- pollLastEntry
  - NavigableMap, 241
- pop
  - Deque, 214
- previous
  - ListIterator, 221
- previousIndex
  - ListIterator, 221
- primitive type, 7, 76, 105, 108, 170
  - Arrays of, 24
- primitive types, 8, 96, 103
  - reflection, 103
- Principle of Anything Goes Behind Closed Doors, 92
- Principle of Indecent Exposure, 75, 84, 89, 91, 92, 95, 97, 103, 105
- Principle of Truth in Advertising, 75, 84, 86, 87, 92, 95, 97, 103, 105
- print spooling, 204
- PrintStream, 48
- priority heap, 202
- priority queue, 200, 210
- PriorityBlockingQueue, 156, 203, 205, 209–211, 216
  - fail-fast iterators, 209
- PriorityLockingQueue, 165
- PriorityQueue, 156, 193, 200, 201, 211, 216
  - add, 203
  - constructors, 201
  - contains, 203
  - element, 203
  - fail-fast iterators, 203
  - offer, 203
  - peek, 203
  - performance, 211
  - poll, 203
  - remove, 203
  - size, 203
  - unsuitable for concurrent use, 203
- probability, 194
- producer threads, 212
- producer-consumer queue, 204, 207
- protection proxies, 251
- proxy, 251
- pull
  - Queue, 199
- push
  - Deque, 212, 213
- put
  - BlockingQueue, 204
  - ConcurrentSkipListMap, 245
  - EnumMap, 239
  - HashMap, 233
  - LinkedHashMap, 235
  - Map, 229
  - TreeMap, 243
- putAll
  - LinkedHashMap, 235
  - Map, 229, 232
- putIfAbsent
  - ConcurrentMap, 164, 243
- quadratic (time), 157
- querying, 174
- Queue, 151, 155, 197–200, 203, 204, 210, 212–214, 228, 231, 253
  - add, 198, 199
  - adding an element, 198
  - choosing a, 216
  - element, 199, 214
  - implementations of, 201
  - iterator, 200
  - nonblocking implementation, 203
  - offer, 198, 199, 213
  - peek, 199, 210, 214
  - performance, 216
  - poll, 199, 214
  - pull, 199

- remove, 199, 210, 214, 225
  - retrieving an element, 198
  - thread safety, 214, 216
- queue, xiv, 197, 212
  - blocking, 217
  - concurrent, 207
  - head, 212
  - tail, 212
  - traversal, 200
  - unbounded, 207
- race condition, 161
- random access, 149, 215, 225, 226, 228, 248
- RandomAccess, 92, 118, 247–249
- range, 187
  - EnumSet, 183
- range views, 186
- raw type, 53, 64, 66, 76, 79, 96, 101, 103, 108, 120, 250
- read
  - Readable, 47
- Readable, 47, 48, 58
  - read, 47
- Reader, 48
- recursion, 127
- recursive
  - bounds, 35, 135
  - calls, 129
  - declaration, 138
  - generics, 138
  - type parameters, 141
- red-black tree, 192, 243
- reference type, 7, 8, 15, 21, 22, 24
- ReferenceQueue, 236
- reflection, 1, 49, 54, 85, 86, 99, 101, 103, 104, 114, 119
  - for generics, 99, 106
  - library, 106
- reifiable type, 75–78, 81, 84, 89, 91, 92, 94, 96, 97, 99, 102, 103, 105, 115, 116, 118, 168
- reification, xiv, 6, 73, 75, 76, 83, 89, 94
- reified component type, 88, 105
- reified type, 23, 83, 86, 89, 91, 97, 239
  - information, 86, 101
- Reinhold, Mark, 122
- remainingCapacity
  - BlockingQueue, 205
- removal by position, 155
- remove
  - AbstractList, 92
  - BlockingQueue, 205
  - Collection, 168, 213
  - ConcurrentMap, 243
  - ConcurrentSkipListMap, 245
  - EnumSet, 182
  - HashSet, 180
  - Iterator, 9, 10, 153, 182, 208, 220
  - List, 219
  - ListIterator, 221
  - Map, 230
  - PriorityQueue, 203
  - Queue, 199, 210, 214, 225
  - TreeMap, 243
- removeAll
  - BlockingQueue, 205
  - Collection, 168, 250
- removeEldestEntry
  - LinkedHashMap, 234, 235
- removeFirst
  - Deque, 214
- removeFirstOccurrence
  - Deque, 212, 213
- removeLast
  - Deque, 214
- removeLastOccurrence
  - Deque, 212, 213
- rendezvous, 211
- replace
  - ConcurrentMap, 244
- replaceAll
  - Collections, 248
- retainAll
  - BlockingQueue, 205
  - Collection, 168, 250
- Retention, 101
- retention annotation, 101
- retrieval by content, 155
- return clause, 68
- return type, 68
- reverse
  - Collections, 27, 247
- reverseOrder
  - Collections, 254
- rotate

- Collections, 247
  - runtime
    - type violations, 252
  - runtime type information, 75
  - RuntimeException, 132, 134
- security properties, 120
- segment locking
  - ConcurrentHashMap, 244
- segments, 244
- Semaphore
  - acquire, 207
  - tryAcquire, 207
- semaphore, 207
  - permit, 207
- Sequential, 118
- Serializable, 12
- serialization, 237
- server applications, 243
- Set, 151, 156, 162, 177, 178, 181, 184, 232, 253, 254
  - add, 177, 227
  - addAll, 227
  - contains, 177
  - implementations of, 177
  - iteration, 177
  - performance, 177
- set, xiv, 177
- set
  - AbstractList, 92
  - ArrayList, 225
  - List, 219, 227
  - ListIterator, 221
- setChanged
  - Observable, 141
- setValue
  - Map.Entry, 230, 241
- sgn, 33
- shuffle
  - Collections, 247, 248
- sign function, 33
- signature, 19, 27, 58, 67
  - erasure, 58
- singleton
  - Collections, 232, 250
- singletonList
  - Collections, 250
- singletonMap
  - Collections, 250
- size
  - AbstractSequentialList, 118
  - Collection, 55, 168, 230
  - ConcurrentHashMap, 244
  - ConcurrentLinkedQueue, 203
  - HashSet, 180
  - Map, 230
  - PriorityQueue, 203
- size ordering, 40
- skip list, 193, 194
  - inserting, 194
- skip lists, 214
- snapshot iterators, 165, 224
  - CopyOnWriteArrayList, 227
  - CopyOnWriteArraySet, 182
  - EnumSet, 183
- snapshot objects, 241
- SoftReference, 237
- software engineering, 157
- sort
  - Collections, 247, 248
- sorted lists
  - merging, 184
- sorted map, 32
- sorted set, 32
- SortedMap, 32, 151, 186, 210, 239, 240, 243
  - ascending key order, 239
  - firstKey, 240
  - headMap, 240
  - lastKey, 240
  - obsolete, 240
  - subMap, 240
  - subsequences, 240
  - tailMap, 240
- SortedSet, 32, 177, 178, 184–188, 192, 193, 210, 220, 253
  - comparator, 186
  - first, 186, 190, 240
  - headSet, 186, 240
  - last, 186, 190
  - subSet, 186, 220
  - tailSet, 186
  - vs SortedMap, 240
- Sorting and Searching, 156
- source 1.4 flag, 67
- specialization, 116
- specialize, 116

- Stack, 160
  - thread safe, 162
- stacks, 160, 212
- static member, 54
- static methods, 132
- static typing, 79
- Stoutamire, David, xiv
- String, xiii, 9, 178
  - implements Comparable<String>, 35
  - subString, 236
- StringBuffer, 162
- StringBuilder, 162
- stub files, 67, 144
- StubException, 70
- stubs, 70, 71, 114, 142
- subclass, 101, 116, 127–129, 143
  - see also* inheritance, 254
- subList
  - List, 220
- subMap
  - NavigableMap, 242
  - SortedMap, 240
- subSet
  - NavigableSet, 189
  - SortedSet, 186, 220
- Substitution Principle, 15, 16, 22, 23, 66
- subString
  - String, 236
- subtype, 15, 17, 22–24, 34, 50, 64, 249
- subtyping, 1, 15
  - contravariant, 23
  - covariant, 23
  - invariant, 23
  - transitive, 15
- super, 18
- super, 19, 35–37, 248
- super wildcard, 19–21, 28
- superclass, 101, 108, 119
- superinterface, 30, 108
- supertype, 15, 18, 22, 23, 28, 30, 64, 186
- suppress warnings annotation, 68
- SuppressWarnings, 68
- swap
  - Collections, 247, 248
- synchronization, 161–164
  - coarse-grained, 252
  - overhead, 251
- synchronized, 54, 161, 211
  - block, 207
  - collections, 162, 164
  - wrapper, 162
- SynchronizedArrayStack, 163
- synchronizedCollection
  - Collections, 251
- synchronizedList
  - Collections, 251
- synchronizedMap
  - Collections, 243, 251
- synchronizedSet
  - Collection, 195
  - Collections, 251
- synchronizedSortedMap
  - Collections, 251
- synchronizedSortedSet
  - Collections, 251
- synchronizing, 161, 211
- SynchronousQueue, 211, 212, 217
  - constructors, 176, 211
  - iterator, 212
- syntactic rat poison, 97
- syntax errors, 102
- system resources, 159

  

- tail, 197
  - of deque, 212
- tailMap
  - NavigableMap, 242
  - SortedMap, 240
- tailSet
  - NavigableSet, 189
  - SortedSet, 186
- take
  - BlockingQueue, 204, 205
- templates, 6
- Thorup, Kresten Krab, xiv
- thread, 159, 160, 207, 211
  - producer, 207
- thread safety, 161–164, 208, 251
  - collections, 164
  - ConcurrentHashMap, 244
  - CopyOnWriteArrayList, 227
  - CopyOnWriteArraySet, 181
  - EnumMap, 239
  - LinkedList, 215
  - Map, 243, 245
  - Queue, 214, 216

- Throwable, 81, 83, 97, 132
  - parameterized subclass of, 81
- throws clause, 83, 132
  - type variable in, 132
- time slicing, 159
- timeout, 204
- TimeUnit, 204
- toArray
  - Collection, 84, 85, 88, 96, 97, 165, 168, 169, 228
- toGenericString
  - Constructor, 49, 107
- top-level parameters, 29
- Torgersen, Mads, xiv
- toString
  - Enum, 46
  - Object, 12, 16–18, 20, 22, 23, 29, 107, 119
- tree, 127, 156
  - balanced binary, 191
  - binary, 191, 193, 194, 202
  - descendants, 191
  - expression, 131
  - ordered, 4
  - red-black, 192, 243
- TreeMap, 156, 193, 243, 245
  - constructors, 243
  - get, 243
  - keys, 243
  - keySet, 242
  - performance, 243
  - put, 243
  - remove, 243
  - values, 243
- TreeSet, 156, 178, 184, 190, 193, 195, 243, 253
  - constructors, 192
  - contains, 191
  - fail-fast iterators, 190, 193
  - not thread-safe, 193
  - unsynchronized, 193
- Truth in Advertising, Principle of, 75, 84, 86, 87, 92, 95, 97, 103, 105
- try, 81, 82, 134
- tryAcquire
  - Semaphore, 207
- type, 43
- type declaration, 4, 79
- type error, 139
- type parameter, 4, 5, 8, 12, 18, 25–28, 32, 36, 41, 53–56, 67, 68, 75, 97, 99–102, 108, 129, 141, 142, 249
  - explicit, 19
  - implicit, 19
  - mutually recursive, 142
  - recursive, 142
- type safety, 252
- type variable, 31, 34, 35, 40, 47, 76, 108
  - bound, 34
- type violations
  - runtime, 252
- TypeVariable, 108
- unbalanced binary tree, 191, 192
- unbounded wildcard, 79, 96, 100, 103
- unboxing, 3, 4, 7–9, 13
- unchecked cast, 68, 71, 79–81, 84, 87–92, 94, 96, 99, 103–106
- unchecked conversion, 64
- unchecked warning, 5, 12, 53, 66, 68, 71, 78, 79, 85, 86, 90, 94, 97, 114–116, 120, 250, 252
- Unicode, 179
- unlock
  - Lock, 165
- unmodifiableCollection
  - Collections, 252
- unmodifiableList
  - Collections, 252
- unmodifiableMap
  - Collections, 252
- unmodifiableSet
  - Collections, 252
- unmodifiableSortedMap
  - Collections, 252
- unmodifiableSortedSet
  - Collections, 252
- unnatural ordering, 37
- unparameterized type, 96, 118
- UnsupportedOperationException, 70, 159, 230, 252
- update
  - Observer, 141
- valueOf
  - enumerated type, 46
- values

- finding extreme, 247, 249
  - finding specific, 247, 249
- Map, 230
- values
  - enumerated type, 46
  - Map, 230
- varargs, 4, 10–13, 84, 92, 94, 95, 97
- variable
  - static final, 43
- variable-length arguments, *see* varargs
- Vector, 161, 251
- versioning, 61
- Viroli, Mirko, xiv
- Vlissides, John, 46, 127, 251
- volatile, 50, 161
- von der Ahé, Peter, xiv
  
- Wadler, Philip, xiv
- wait-free algorithm, 203
- warnings, 76, 79, 81, 94
- weak reference, 236, 254
- WeakHashMap, 156, 232, 235, 237, 254
  - fail-fast iterators, 237
  - key-value pair, 236
  - keys, 236, 237
  - performance, 237
  - values, 237
- weakly consistent iterators, 155, 165
  - ArrayBlockingQueue, 209
  - ConcurrentHashMap, 244
  - ConcurrentLinkedQueue, 203
  - ConcurrentSkipListSet, 194
  - EnumMap, 239
  - EnumSet, 183
  - LinkedBlockingDeque, 216
  - LinkedBlockingQueue, 208
- WeakReference, 236, 237
  - get, 236
- wildcard, 1, 15, 17–19, 23, 25, 28–30, 35–37, 47, 79, 100, 108, 122
  - bounded, 28
  - capture, 27, 28
  - restrictions on, 28
  - specialization at, 121
  - type, 28, 29, 80
  - unbounded, 79, 96, 100, 103
- WildcardType, 108
- wrapper, 67, 71, 101, 116, 162, 181, 247, 251
  - class, 71
  - synchronization, 251
  - type checking, 251
  - unmodifiability, 251, 252
  - see also* delegation, 254
- Writer, 48
  - Xbootclasspath flag, 148
  - Xlint:unchecked flag, 53