

# Table of Contents

<b>LIST OF PROGRAMS .....</b>	<b>IX</b>
<b>TABLE OF FIGURES .....</b>	<b>XVII</b>
<b>PREFACE .....</b>	<b>XXI</b>
WHY THIS BOOK .....	XXI
WHOM THIS BOOK IS FOR .....	XXII
HOW THIS BOOK IS ORGANIZED .....	XXII
SOFTWARE AND HARDWARE .....	XXIII
HOW TO USE THIS BOOK .....	XXIV
TYPOGRAPHIC CONVENTIONS .....	XXIV
HOW TO REACH THE AUTHOR .....	XXIV
THE BOOK'S WEB SITE .....	XXIV
<b>ACKNOWLEDGEMENTS .....</b>	<b>XXV</b>
<b>1    MULTITHREADED PROGRAMMING IN JAVA .....</b>	<b>1</b>
1.1    PERSPECTIVES OF CONCURRENT PROGRAMMING .....	1
1.2    EVOLUTION OF JAVA CONCURRENCY SUPPORT .....	3

1.3	JAVA THREADS .....	4
1.3.1	<i>Potential Issues with Java Concurrency</i> .....	4
1.3.2	<i>All Possible States for a Java Thread</i> .....	5
1.3.3	<i>Livelock, Starvation and Deadlock</i> .....	6
1.4	CREATING A THREAD .....	6
1.4.1	<i>Implements Runnable</i> .....	13
1.4.2	<i>Extends Thread</i> .....	15
1.5	SYNCHRONIZATION .....	17
1.5.1	<i>Synchronized Methods</i> .....	17
1.5.2	<i>Synchronized Blocks</i> .....	19
1.6	INTER-THREAD COMMUNICATIONS .....	21
1.6.1	<i>Busy Waiting / Busy Spinning</i> .....	22
1.6.2	<i>A Simple Buffer Accessed by a Single Thread</i> .....	22
1.6.3	<i>The Simple Buffer Accessed by Two Threads: Busy-Wait with no Conditional Check (OOB)</i> .....	24
1.6.4	<i>The Simple Buffer Accessed by Two Threads: Busy-Wait with Conditional Check but no Synchronization (Livelock)</i> .....	25
1.6.5	<i>The Simple Buffer Accessed by Two Threads: Busy-Wait with Conditional Check and Synchronization (Starvation)</i> .....	28
1.6.6	<i>Guarded Blocks with Asynchronous Waiting</i> .....	31
1.6.7	<i>Turning the SimpleBuffer Class into a First-In-First-Out Queue-Like Data Structure</i> .....	34
1.7	DEADLOCK .....	36
1.7.1	<i>A Deadlock Example with a Parent and a Child Thread Calling the callMe Method of two Non-Threaded Objects</i> .....	36
1.7.2	<i>Diagnosing Deadlocks using the jvisualvm Tool</i> .....	39
1.8	SUSPENDING, RESUMING, AND STOPPING THREADS .....	42
1.9	THE BRIDGE EXAMPLE .....	43
1.10	SUMMARY .....	43
1.11	EXERCISES .....	44
2	<b>JAVA THREAD EXECUTOR SERVICE FRAMEWORK</b> .....	<b>47</b>

2.1	THE CALLABLE AND FUTURE INTERFACES.....	48
2.2	THE EXECUTOR INTERFACE.....	49
2.2.1	<i>Executor</i> .....	49
2.2.2	<i>ExecutorService</i> .....	49
2.2.3	<i>ScheduledExecutorService</i> .....	50
2.3	THE THREAD POOL CLASSES.....	51
2.3.1	<i>The RunnableFuture interface and the FutureTask class</i> .....	51
2.3.2	<i>AbstractExecutorService</i> .....	53
2.3.3	<i>ThreadPoolExecutor</i> .....	54
2.3.4	<i>ForkJoinPool</i> .....	57
2.3.5	<i>ScheduledThreadPoolExecutor</i> .....	58
2.4	THE EXECUTORS UTILITY CLASS.....	60
2.4.1	<i>The DefaultThreadFactory Inner Class</i> .....	62
2.4.2	<i>The newSingleThreadExecutor Method</i> .....	63
2.4.3	<i>The newFixedThreadPool method</i> .....	63
2.4.4	<i>The newCachedThreadPool method</i> .....	64
2.5	SOME EXECUTORSERVICE EXAMPLES .....	64
2.5.1	<i>The Method execute (Runnable) does not Return a Result</i> .....	65
2.5.2	<i>The Method submit (Runnable) Returns a Future Object (Status)</i> .....	66
2.5.3	<i>The Method submit(Callable) Returns a Future Object (Result)</i> .....	67
2.5.4	<i>The Method invokeAny (Callables) Succeeds if Any One Task Succeeds</i> .....	69
2.5.5	<i>The Method invokeAll (Callables) Succeeds if All Callables Succeed</i> .....	71
2.6	SUMMARY .....	73
2.7	EXERCISES.....	73
<b>3</b>	<b>THE JAVA COLLECTIONS FRAMEWORK .....</b>	<b>75</b>
3.1	COLLECTIONS OVERVIEW .....	75
3.2	THE COLLECTION INTERFACES.....	77
3.2.1	<i>The Iterable and Iterator Interfaces</i> .....	77

3.2.2	<i>The Collection Interface</i> .....	79
3.2.3	<i>The Set Interface</i> .....	80
3.2.4	<i>The SortedSet Interface</i> .....	80
3.2.5	<i>The NavigableSet Interface</i> .....	81
3.2.6	<i>The List Interface</i> .....	82
3.2.7	<i>The ListIterator Interface</i> .....	83
3.2.8	<i>The Queue Interface</i> .....	84
3.2.9	<i>The Deque Interface</i> .....	85
3.3	THE SET COLLECTION CLASSES.....	86
3.3.1	<i>The AbstractSet Class</i> .....	87
3.3.2	<i>The HashSet Class</i> .....	88
3.3.3	<i>The LinkedHashSet Class</i> .....	91
3.3.4	<i>The TreeSet Class</i> .....	92
3.4	THE LIST COLLECTION CLASSES .....	95
3.4.1	<i>The AbstractList Class</i> .....	96
3.4.2	<i>The RandomAccess Interface</i> .....	97
3.4.3	<i>The ArrayList Class</i> .....	97
3.4.4	<i>The AbstractSequentialList Class</i> .....	101
3.4.5	<i>The LinkedList Class</i> .....	102
3.4.6	<i>ArrayList versus LinkedList</i> .....	108
3.5	THE QUEUE COLLECTION CLASSES.....	109
3.5.1	<i>The ArrayDeque Class</i> .....	110
3.5.2	<i>The AbstractQueue Class</i> .....	114
3.5.3	<i>The PriorityQueue Class</i> .....	115
3.6	THE MAP INTERFACES .....	118
3.6.1	<i>The Map Interface</i> .....	119
3.6.2	<i>The SortedMap Interface</i> .....	121
3.6.3	<i>The NavigableMap Interface</i> .....	122

3.7	THE MAP CLASSES .....	123
3.7.1	<i>The AbstractMap Class</i> .....	123
3.7.2	<i>The HashMap Class</i> .....	125
3.7.3	<i>The LinkedHashMap Class</i> .....	129
3.7.4	<i>The TreeMap Class</i> .....	133
3.7.5	<i>The IdentityHashMap Class</i> .....	136
3.7.6	<i>The WeakHashMap Class</i> .....	137
3.8	THE ALGORITHMS APPLIED TO COLLECTIONS .....	137
3.8.1	<i>The Algorithms Applicable to Collection</i> .....	139
3.8.2	<i>The Algorithms Applicable to Set</i> .....	141
3.8.3	<i>The Algorithms Applicable to List</i> .....	142
3.8.4	<i>The Algorithms Applicable to Queue</i> .....	145
3.8.5	<i>The Algorithms Applicable to Map</i> .....	147
3.8.6	<i>The emptyXxxx and singletonXxxx Algorithms</i> .....	149
3.9	THE ARRAYS CLASS.....	151
3.10	LEGACY COLLECTION CLASSES.....	152
3.11	SUMMARY .....	153
3.12	EXERCISES.....	153
<b>4</b>	<b>ATOMIC OPERATIONS .....</b>	<b>155</b>
4.1	THE NATIVE UNSAFE CLASS.....	157
4.2	ATOMICINTEGER.....	159
4.2.1	<i>Implementation</i> .....	159
4.2.2	<i>An Example</i> .....	161
4.3	OTHER ATOMIC CLASSES .....	163
4.4	ATOMICINTEGERARRAY.....	164
4.5	SUMMARY .....	168
4.6	EXERCISES.....	168
<b>5</b>	<b>LOCKS.....</b>	<b>171</b>

5.1	THE JAVA LOCKS .....	172
5.1.1	<i>The Lock Interface</i> .....	173
5.1.2	<i>The ReentrantLock Class</i> .....	174
5.1.3	<i>An Example</i> .....	179
5.2	THE JAVA READWRITELOCKS .....	181
5.2.1	<i>The ReadWriteLock Interface</i> .....	181
5.2.2	<i>The ReentrantReadWriteLock Class</i> .....	181
5.2.3	<i>An Example</i> .....	186
5.3	THE CONDITION INTERFACE .....	188
5.4	ABSTRACT QUEUED SYNCHRONIZERS .....	193
5.4.1	<i>The AbstractOwnableSynchronizer</i> .....	193
5.4.2	<i>The AbstractQueuedSynchronizer</i> .....	194
5.4.3	<i>The AbstractQueuedLongSynchronizer</i> .....	201
5.5	SUMMARY.....	201
5.6	EXERCISES .....	202
<b>6</b>	<b>SYNCHRONIZERS .....</b>	<b>205</b>
6.1	SEMAPHORE .....	206
6.1.1	<i>Semaphore Implementation</i> .....	206
6.1.2	<i>An Example of Using a Binary Semaphore</i> .....	211
6.1.3	<i>A Buffer Synchronized with Semaphores</i> .....	214
6.2	CYCLICBARRIER .....	216
6.2.1	<i>CyclicBarrier Implementation</i> .....	216
6.2.2	<i>An Example of Using a CyclicBarrier</i> .....	219
6.3	COUNTDOWNLATCH .....	221
6.3.1	<i>CountDownLatch Implementation</i> .....	221
6.3.2	<i>An Example of Using a CountDownLatch</i> .....	223
6.4	EXCHANGER .....	226
6.4.1	<i>Exchanger Implementation</i> .....	226

6.4.2	<i>An Example of Using an Exchanger</i> .....	228
6.5	PHASER .....	231
6.5.1	<i>An Overview of Phaser Implementation</i> .....	231
6.5.2	<i>An Example of Using a Phaser</i> .....	232
6.6	SUMMARY .....	234
6.7	EXERCISES .....	234
<b>7</b>	<b>SYNCHRONIZED COLLECTIONS</b> .....	<b>235</b>
7.1	ARRAYBLOCKING, SYNCHRONOUS, DELAY, AND PRIORITYBLOCKING QUEUES .....	236
7.1.1	<i>The BlockingQueue Interface</i> .....	236
7.1.2	<i>ArrayBlockingQueue</i> .....	237
7.1.3	<i>SynchronousQueue</i> .....	243
7.1.4	<i>DelayQueue</i> .....	247
7.1.5	<i>PriorityBlockingQueue</i> .....	252
7.2	CONCURRENT MAPS, QUEUES AND SET .....	256
7.2.1	<i>ConcurrentHashMap</i> .....	256
7.2.2	<i>ConcurrentLinkedQueue</i> .....	263
7.2.3	<i>ConcurrentLinkedDeque</i> .....	268
7.2.4	<i>ConcurrentSkipListMap</i> .....	273
7.2.5	<i>ConcurrentSkipListSet</i> .....	283
7.3	LINKEDBLOCKING AND TRANSFER QUEUES .....	289
7.3.1	<i>LinkedBlockingQueue</i> .....	289
7.3.2	<i>LinkedBlockingDeque</i> .....	295
7.3.3	<i>LinkedTransferQueue</i> .....	302
7.4	COPYONWRITE ARRAYLIST AND ARRAYSET .....	307
7.4.1	<i>CopyOnWriteArrayList</i> .....	307
7.4.2	<i>CopyOnWriteArraySet</i> .....	313
7.5	SUMMARY .....	314
7.6	EXERCISES .....	314

<b>8</b>	<b>PARALLEL PROGRAMMING USING THE FORK-JOIN FRAMEWORK .....</b>	<b>317</b>
8.1	THE FORKJOINTASK<V> CLASS .....	318
8.1.1	<i>The ForkJoinWorkerThread Class .....</i>	<i>318</i>
8.1.2	<i>The ForkJoinTask Class .....</i>	<i>321</i>
8.2	THE FORKJOINPOOL CLASS.....	324
8.3	THE RECURSIVEACTION CLASS.....	330
8.3.1	<i>Definition of the RecursiveAction Class .....</i>	<i>330</i>
8.3.2	<i>An Example.....</i>	<i>331</i>
8.4	THE RECURSIVETASK<V> CLASS.....	334
8.4.1	<i>The Definition of the RecursiveTask Class .....</i>	<i>334</i>
8.4.2	<i>An Example.....</i>	<i>335</i>
8.5	SUMMARY.....	338
8.6	EXERCISES .....	338
	<b>APPENDIX A ALGORITHM ANALYSIS .....</b>	<b>341</b>
A.1	THE BIG-O NOTATION.....	341
A.2	GROWTH RATE COMPARISON .....	342
A.3	RUNNING TIME ESTIMATES.....	345
A.4	EXAMPLES.....	345
A.4.1	<i>Maximum Subsequence Sum Problem: <math>O(n^3)</math>.....</i>	<i>346</i>
A.4.2	<i>MAXIMUM SUBSEQUENCE SUM PROBLEM: <math>O(N^2)</math> .....</i>	<i>347</i>
A.4.3	<i>MAXIMUM SUBSEQUENCE SUM PROBLEM: <math>O(N)</math> (an Online Algorithm).....</i>	<i>348</i>
A.4.4	<i>Running Times of a Binary Search Algorithm .....</i>	<i>349</i>
	<b>APPENDIX B THE BRIDGE EXERCISE .....</b>	<b>351</b>
	<b>INDEX .....</b>	<b>359</b>