

# Contents

<b>1</b>	<b>Java Programming</b>	<b>1</b>
1.1	Classes, Types, and Objects . . . . .	3
1.2	Methods . . . . .	11
1.3	Expressions . . . . .	17
1.4	Control Flow . . . . .	25
1.5	Arrays . . . . .	32
1.6	Simple Input and Output . . . . .	34
1.7	An Example Program . . . . .	37
1.8	Packages . . . . .	40
1.9	Writing a Java Program . . . . .	42
1.10	Utilities in the java.lang Package . . . . .	49
1.11	Exercises . . . . .	51
<b>2</b>	<b>Object-Oriented Design</b>	<b>55</b>
2.1	Goals and Principles . . . . .	57
2.2	Inheritance and Polymorphism . . . . .	62
2.3	Exceptions . . . . .	76
2.4	Interfaces, Abstract Classes, and Casting . . . . .	80
2.5	Recursion and Other Design Patterns . . . . .	88
2.6	Exercises . . . . .	95
<b>3</b>	<b>Analysis Tools</b>	<b>99</b>
3.1	What Is Running Time Anyway? . . . . .	101
3.2	Pseudo-Code . . . . .	103
3.3	The Seven Functions Used in This Book . . . . .	105
3.4	Analysis of Algorithms . . . . .	114
3.5	Simple Justification Techniques . . . . .	128
3.6	Exercises . . . . .	132

<b>4</b>	<b>Stacks, Queues, and Recursion</b>	<b>139</b>
4.1	Using Recursion . . . . .	141
4.2	Stacks . . . . .	152
4.3	Queues . . . . .	170
4.4	Linked Lists . . . . .	182
4.5	Double-Ended Queues . . . . .	189
4.6	Exercises . . . . .	194
<b>5</b>	<b>Vectors, Lists, and Sequences</b>	<b>199</b>
5.1	Vectors and Array Lists . . . . .	201
5.2	Lists . . . . .	210
5.3	Sequences . . . . .	221
5.4	Favorite Lists and the Move-to-Front Heuristic . . . . .	225
5.5	Iterators . . . . .	232
5.6	Exercises . . . . .	237
<b>6</b>	<b>Trees</b>	<b>243</b>
6.1	The Tree Abstract Data Type . . . . .	245
6.2	Basic Algorithms on Trees . . . . .	252
6.3	Binary Trees . . . . .	263
6.4	Data Structures for Representing Trees . . . . .	278
6.5	Exercises . . . . .	293
<b>7</b>	<b>Priority Queues</b>	<b>303</b>
7.1	The Priority Queue Abstract Data Type . . . . .	305
7.2	Implementing a Priority Queue with a List . . . . .	313
7.3	Heaps . . . . .	320
7.4	Adaptable Priority Queues . . . . .	342
7.5	Exercises . . . . .	347
<b>8</b>	<b>Maps and Dictionaries</b>	<b>353</b>
8.1	The Map Abstract Data Type . . . . .	355
8.2	Hash Tables . . . . .	360
8.3	The Dictionary Abstract Data Type . . . . .	376
8.4	Skip Lists . . . . .	386
8.5	Extensions and Applications of Dictionaries . . . . .	395
8.6	Exercises . . . . .	401

<b>9</b>	<b>Search Trees</b>	<b>407</b>
9.1	Binary Search Trees . . . . .	410
9.2	AVL Trees . . . . .	421
9.3	Splay Trees . . . . .	432
9.4	(2,4) Trees . . . . .	443
9.5	Red-Black Trees . . . . .	455
9.6	External Searching in B-Trees . . . . .	473
9.7	Exercises . . . . .	478
<b>10</b>	<b>Sorting, Sets, and Selection</b>	<b>485</b>
10.1	Merge-Sort . . . . .	487
10.2	Quick-Sort . . . . .	501
10.3	A Lower Bound on Comparison-Based Sorting . . . . .	513
10.4	Bucket-Sort and Radix-Sort . . . . .	515
10.5	Comparison of Sorting Algorithms . . . . .	518
10.6	The Set ADT and Union/Find Structures . . . . .	520
10.7	Selection . . . . .	529
10.8	Exercises . . . . .	532
<b>11</b>	<b>Text Processing</b>	<b>539</b>
11.1	String Operations . . . . .	541
11.2	Pattern Matching Algorithms . . . . .	544
11.3	Tries . . . . .	556
11.4	Text Compression . . . . .	567
11.5	Text Similarity Testing . . . . .	570
11.6	Exercises . . . . .	575
<b>12</b>	<b>Graphs</b>	<b>581</b>
12.1	The Graph Abstract Data Type . . . . .	583
12.2	Data Structures for Graphs . . . . .	589
12.3	Graph Traversals . . . . .	597
12.4	Directed Graphs . . . . .	612
12.5	Weighted Graphs . . . . .	626
12.6	Shortest Paths . . . . .	627
12.7	Minimum Spanning Trees . . . . .	638
12.8	Exercises . . . . .	648
<b>A</b>	<b>Useful Mathematical Facts</b>	<b>659</b>
	<b>Bibliography</b>	<b>667</b>
	<b>Index</b>	<b>672</b>