CONTENTS

PREFACE XV

1 THE MANAGEMENT OF COMPLEXITY 1
1.1 The Control of Complexity 2
1.2 Abstraction, Information Hiding, and Layering 3
1.3 Division into Parts 6
  1.3.1 Encapsulation and Interchangeability 6
  1.3.2 Interface and Implementation 7
  1.3.3 The Service View 8
  1.3.4 Repetition and Recursion 9
1.4 Composition 11
1.5 Layers of Specialization 14
1.6 Multiple Views 16
1.7 Patterns 16
1.8 Chapter Summary 18
  Further Information 19
  Study Questions 20
  Exercises 21
  Programming Projects 21

2 ABSTRACT DATA TYPES 23
2.1 What Is a Type? 24
  2.1.1 Classes 25
  2.1.2 Interfaces and Polymorphism 27
2.2 Abstract Data Types 30
2.3 The Fundamental ADTs 34
  2.3.1 Collection 34
5 Increasing Confidence in Correctness 97

5.1 Program Proofs 97
  5.1.1 Invariants 98
  5.1.2 Analyzing Loops 100
  5.1.3 Asserting That the Outcome Is Correct 103
  5.1.4 Progress toward an Objective 104
  5.1.5 Manipulating Unnamed Quantities 105
  5.1.6 Function Calls 106
  5.1.7 Recursive Algorithms 107

5.2 Program Testing 109

5.3 Chapter Summary 111
  Further Information 111
  Study Questions 111
  Exercises 112
  Programming Projects 114

6 Vectors 117

6.1 The Vector Data Structure 117
6.2 Enumeration 127
6.3 Application–Silly Sentences 128
6.4 Application–Memory Game 131
6.5 Application–Shell Sort 136
6.6 A Visual Vector 140
6.7 Chapter Summary 144
  Further Information 144
  Study Questions 144
  Exercises 145
  Programming Projects 149
7 Sorting Vectors 153
7.1 Divide and Conquer 153
  7.1.1 Binary Search 155
7.2 Sorted Vectors 158
7.3 Merge Sort 161
7.4 Partitioning 165
  7.4.1 The Pivot Algorithm 166
  7.4.2 Finding the nth Element 168
  7.4.3 Quick Sort 171
7.5 Chapter Summary 175
Further Information 175
Study Questions 176
Exercises 176
Programming Projects 179

8 Linked Lists 181
8.1 Varieties of Linked Lists 185
8.2 LISP-Style Lists 187
8.3 The LinkedList Abstraction 189
8.4 Application–Asteroids Game 197
8.5 Application–Infinite-Precision Integers 207
8.6 Chapter Summary 211
Further Information 212
Study Questions 212
Exercises 212
Programming Projects 214

9 List Variations 217
9.1 Sorted Lists 217
  9.1.1 Fast Merge 219
  9.1.2 Execution Timings for Merge Operations 220
9.2 Self-Organizing Lists 221
9.3 Skip Lists 223
9.4 Chapter Summary 232
Further Information 232
Study Questions 233
10 STACKS 237
10.1 The Stack ADT 239
10.2 Checking for Balanced Parentheses 240
10.3 Towers of Hanoi, Revisited 242
10.4 A Four-Function Calculator 244
10.5 A Solitaire Program 250
10.6 Implementation of the Stack Abstraction 257
10.7 Chapter Summary 260
Further Information 260
Study Questions 260
Exercises 261
Programming Projects 264

11 DEQUES 267
11.1 A Fractal Snowflake 269
11.2 Depth- and Breadth-First Search 273
11.3 An Implementation: The IndexedDeque 281
11.4 Chapter Summary 287
Further Information 287
Study Questions 287
Exercises 288
Programming Projects 291

12 QUEUES 293
12.1 The Queue ADT 294
12.2 The Caterpillar Game 295
12.3 A Pastry Factory Simulation 302
12.4 Implementation of the Queue Abstraction 308
12.4.1 A Vector-Based Queue 312
12.4.2 The Ring Buffer Queue 313
12.4.3 Piped Input/Output Streams 317
12.5 Chapter Summary 318
Further Information 318
16 **HASH TABLES** 417

16.1 Hash Functions 418
   16.1.1 Hash Functions 421
   16.1.2 Hash Functions in the Java Standard Library 423

16.2 Collision Resolution 424

16.3 Hash Table Sorting Algorithms 427
   16.3.1 Counting Sort 428
   16.3.2 Radix Sorting 430

16.4 Open-Address Hashing 433

16.5 The Hashtable Data Type 437

16.6 Application–Ranking Poker Hands 440

16.7 Chapter Summary 442
   Further Information 443
   Study Questions 444
   Exercises 445
   Programming Projects 447

17 **MAPS** 451

17.1 Example Programs 453
   17.1.1 Silly-Sentence Generation, Revisited 453
   17.1.2 An Address Database 458
   17.1.3 A Concordance 461

17.2 An Implementation 463

17.3 Searching Problems and Maps 469

17.4 Chapter Summary 471
   Further Information 471
   Study Questions 472
   Exercises 473
   Programming Projects 474
18  SETS  479
18.1 Changing a Bag into a Set  480
18.2 Set Union, Intersection, and Differences  482
18.3 Sorted List Sets  485
18.4 Application—A Spelling Checker  490
18.5 The Union-Find Problem  491
18.6 The BitSet Abstraction  495
18.7 Application—Prime Number Sieve  498
18.8 Chapter Summary  500
   Further Information  501
   Study Questions  501
   Exercises  501
   Programming Projects  503

19  MATRICES  507
19.1 Java Matrices  507
19.2 Application—Rain Game  510
19.3 Binary Matrices  514
19.4 Application—The Game of Life  515
19.5 Sparse Vectors  519
19.6 An Application—(Almost) Infinitely Large Hash Tables  521
19.7 Sparse Matrices  523
19.8 Noninteger Keys  524
19.9 Chapter Summary  526
   Further Information  526
   Study Questions  527
   Exercises  527
   Programming Projects  528

20  GRAPHS  531
20.1 Adjacency-Matrix Representation  533
20.2 Edge-List Representation  537
20.3 Weighted-Graph Representation  541
   20.3.1 Weighted-Adjacency Matrix  542
   20.3.2 Floyd’s Algorithm  542
   20.3.3 Weighted-Edge-List Representation  543
20.3.4 Dijkstra's Algorithm 544

20.4 Other Graph Problems 549
20.4.1 Topological Sorting 549
20.4.2 Depth-First Search Spanning Tree 550
20.4.3 Problem–The Traveling Salesman 551

20.5 Chapter Summary 553
Further Information 553
Study Questions 553
Exercises 554
Programming Projects 556

APPENDIX A JAVA SYNTAX 559

A.1 Program Structure 559
A.1.1 Packages 559
A.1.2 Import Declaration 560
A.1.3 Class Declaration 560
A.1.4 Interface Declaration 561
A.1.5 Method Declaration 561
A.1.6 Constructors 563
A.1.7 Data Field Declaration 563

A.2 Statements 564
A.2.1 Declaration Statement 564
A.2.2 Assignment Statement 564
A.2.3 Procedure Calls 564
A.2.4 if Statement 565
A.2.5 switch Statement 565
A.2.6 while Statement 566
A.2.7 for Statement 566
A.2.8 return Statement 566
A.2.9 throw Statement 567
A.2.10 try Statement 567

A.3 Expressions 567
A.3.1 Literal 568
A.3.2 Variables 568
A.3.3 Data Field and Method Access 569
A.3.4 Operators 570
A.3.5 Object Creation 570
A.3.6 Arrays 571

A.4 Files 571
APPENDIX B  IMPORT LIBRARIES  573

APPENDIX C  DATA STRUCTURES IN THE JAVA STANDARD LIBRARY  577
C.1 Collection  577
C.2 Enumerators and Iterators  578
C.3 Vectors  578
C.4 Lists  579
C.5 Stack, Queue, and Deque  580
C.6 Priority Queue  580
C.7 Binary Search Tree  580
C.8 Hash Tables  581
C.9 Set  581
C.10 Map  582

BIBLIOGRAPHY 583
INDEX  589