Table of Contents

Preface.

Chapter 1 Object-Oriented Programming and Class Hierarchies.

1.1 ADTs, Interfaces, and the Java API.
1.2 Introduction to Object-Oriented Programming.
1.3 Method Overriding, Method Overloading, and Polymorphism.
1.4 Abstract Classes.
1.5 Class Object and Casting.
1.6 A Java Inheritance Example—The Exception Class Hierarchy.
1.7 Packages and Visibility.
1.8 A Shape Class Hierarchy.

Chapter Review, Exercises, and Programming Projects.

Chapter 2 Lists and the Collections Framework.

2.1 The List Interface and ArrayList Class.
2.2 Applications of ArrayList.
2.3 Implementation of an ArrayList Class.
2.4 Algorithm Efficiency and Big-O.
2.5 Single-Linked Lists.
2.6 Double-Linked Lists and Circular Lists.
2.7 The LinkedList Class and the Iterator, ListIterator, and Iterable Interfaces.
2.8 Implementation of a Double-Linked List Class.
2.9 The Collections Framework Design.
2.10 Application of the LinkedList Class.

Chapter Review, Exercises, and Programming Projects.

Chapter 3 Stacks.

3.1 Stack Abstract Data Type.
3.2 Stack Applications.
3.3 Implementing a Stack.
3.4 Additional Stack Applications.

Chapter Review, Exercises, and Programming Projects.

Chapter 4 Queues.

4.1 Queue Abstract Data Type.
4.2 Maintaining a Queue of Customers.
4.3 Implementing the Queue Interface.
4.4 The Deque Interface.
4.5 Simulating Waiting Lines Using Queues.
Chapter Review, Exercises, and Programming Projects.

Chapter 5 Recursion.
5.1 Recursive Thinking.
5.2 Recursive Definitions of Mathematical Formulas.
5.3 Recursive Array Search.
5.4 Recursive Data Structures.
5.5 Problem Solving with Recursion.
5.6 Backtracking.
Chapter Review, Exercises, and Programming Projects.

Chapter 6 Trees.
6.1 Tree Terminology and Applications.
6.2 Tree Traversals.
6.3 Implementing a BinaryTree Class.
6.4 Binary Search Trees.
6.5 Heaps and Priority Queues.
6.6 Huffman Trees.
Chapter Review, Exercises, and Programming Projects.

Chapter 7 Sets and Maps.
7.1 Sets and the Set Interface.
7.2 Maps and the Map Interface.
7.3 Hash Tables.
7.4 Implementing the Hash Table.
7.5 Implementation Considerations for Maps and Sets.
7.6 Additional Applications of Maps.
7.7 Navigable Sets and Maps.
Chapter Review, Exercises, and Programming Projects.

Chapter 8 Sorting.
8.1 Using Java Sorting Methods.
8.2 Selection Sort.
8.3 Bubble Sort.
8.4 Insertion Sort.
8.5 Comparison of Quadratic Sorts.
8.6 Shell Sort: A Better Insertion Sort.
8.7 Merge Sort.
8.8 Heapsort.
8.9 Quicksort.
8.10 Testing the Sort Algorithms.
8.11 The Dutch National Flag Problem (Optional Topic).

Chapter Review, Exercises, and Programming Projects.

Chapter 9 Self-Balancing Search Trees.
9.1 Tree Balance and Rotation.
9.2 AVL Trees.
9.3 Red-Black Trees.
9.4 2-3 Trees.
9.5 B-Trees and 2-3-4 Trees.
9.6 Skip-Lists.

Chapter Review, Exercises, and Programming Projects.

Chapter 10 Graphs.
10.1 Graph Terminology.
10.2 The Graph ADT and Edge Class.
10.3 Implementing the Graph ADT.
10.4 Traversals of Graphs.
10.5 Applications of Graph Traversals.
10.6 Algorithms Using Weighted Graphs.

Chapter Review, Exercises, and Programming Projects.

Appendix A Introduction to Java.
A.1 The Java Environment and Classes.
A.2 Primitive Data Types and Reference Variables.
A.3 Java Control Statements.
A.4 Methods and Class Math.
A.5 The String, StringBuilder, and StringBuffer Classes.
A.6 Wrapper Classes for Primitive Types.
A.7 Defining Your Own Classes.
A.8 Arrays.
A.9 Input/Output Using Class JOptionPane.
A.10 Input/Output Using Streams and the Scanner Class.
A.11 Catching Exceptions.
A.12 Throwing Exceptions.


Appendix B Overview of UML.
B.1 The Class Diagram.
B.2 Sequence Diagrams.

Appendix C Event-Oriented Programming.
C.1 Elements of an Event-Oriented Application.
C.2 Overview of the AWT and Swing Hierarchy.
C.3 Layout Managers.
C.4 Components for Data Entry.
C.5 Using Data Entry Components in a GUI.
C.6 Menus and Toolbars.
C.7 Processing Mouse Events.

**Appendix D Testing and Debugging.**

D.1 Testing Using the JUnit Framework.
D.2 Debugging a Program.
D.3 Visualizing Data Structures.

Glossary.

Index.