Data Structures, Algorithms, and Applications in C++

Sartaj Sahni

University of Florida
PART I  PRELIMINARIES

CHAPTER 1  PROGRAMMING IN C++  1
1.1  Introduction 3
1.2  Functions and Parameters  3
  1.2.1  Value Parameters  3
  1.2.2  Template Functions  4
  1.2.3  Reference Parameters  5
  1.2.4  Const Reference Parameters  6
  1.2.5  Return Values  7
1.2.6  Recursive Functions  8
  Fibonacci numbers
  Factorial
  Permutations
## Contents

1.3 Dynamic Memory Allocation 14  
1.3.1 The Operator `new` 14  
1.3.2 One-Dimensional Arrays 15  
1.3.3 Exception Handling 15  
1.3.4 The Operator `delete` 16  
1.3.5 Two-Dimensional Arrays 17  

1.4 Classes 20  
1.4.1 The Class `Currency` 20  
1.4.2 Using a Different Representation 28  
1.4.3 Operator Overloading 29  
1.4.4 Throwing Exceptions 32  
1.4.5 Friends and Protected Class Members 33  
1.4.6 Addition of `#ifndef`, `#define`, and `#endif` Statements 36  

1.5 Testing and Debugging 37  
1.5.1 What Is Testing? 37  
1.5.2 Designing Test Data 40  
1.5.3 Debugging 43  

1.6 References and Selected Readings 44  

### CHAPTER 2  PROGRAM PERFORMANCE 45  
2.1 Introduction 47  
2.2 Space Complexity 48  
2.2.1 Components of Space Complexity 48  
2.2.2 Examples 54  
2.3 Time Complexity 57  
2.3.1 Components of Time Complexity 57  
2.3.2 Operation Counts 58  
2.3.3 Step Counts 68  
2.4 Asymptotic Notation (O, Ω, Θ, o) 83  
2.4.1 Big Oh Notation (O) 84  
2.4.2 Omega Notation (Ω) 88  
2.4.3 Theta Notation (Θ) 89  
2.4.4 Little Oh (o) 90
PART II DATA STRUCTURES

CHAPTER 3 DATA REPRESENTATION 111

3.1 Introduction 113
3.2 Linear Lists 114
3.3 Formula-Based Representation 116
   3.3.1 Representation 116
   3.3.2 The Exception Class NoMem 117
   3.3.3 Operations 118
   3.3.4 Evaluation 122
3.4 Linked Representation 129
   3.4.1 The Classes ChainNode and Chain 129
   3.4.2 Operations 130
   3.4.3 Extensions to the Class Chain 136
   3.4.4 A Chain Iterator Class 137
   3.4.5 Circular List Representation 138
   3.4.6 Comparison with Formula-Based Representation 139
   3.4.7 Doubly Linked List Representation 140
   3.4.8 Summary 142
3.5 Indirect Addressing 146
   3.5.1 Representation 146
   3.5.2 Operations 147
3.6 Simulating Pointers 152
   3.6.1 SimSpace Operations 153
   3.6.2 Chains Using Simulated Pointers 157
3.7 A Comparison 163
3.8 Applications 164
   3.8.1 Bin Sort 164
   3.8.2 Radix Sort 170
CHAPTER 4 ARRAYS AND MATRICES 189

4.1 Arrays 191
   4.1.1 The Abstract Data Type 191
   4.1.2 Indexing a C++ Array 192
   4.1.3 Row- and Column-Major Mappings 192
   4.1.4 The Class Array1D 194
   4.1.5 The Class Array2D 197

4.2 Matrices 204
   4.2.1 Definitions and Operations 204
   4.2.2 The Class Matrix 206

4.3 Special Matrices 210
   4.3.1 Definitions and Applications 210
   4.3.2 Diagonal Matrices 212
   4.3.3 Tridiagonal Matrix 214
   4.3.4 Triangular Matrices 216
   4.3.5 Symmetric Matrices 218

4.4 Sparse Matrices 221
   4.4.1 Motivation 221
   4.4.2 Array Representation 222
   4.4.3 Linked Representation 229

CHAPTER 5 STACKS 239

5.1 The Abstract Data Type 241
5.2 Derived Classes and Inheritance 241
5.3 Formula-Based Representation 243
5.4 Linked Representation 248
5.5 Applications 252
   5.5.1 Parenthesis Matching 252
   5.5.2 Towers of Hanoi 254
   5.5.3 Rearranging Railroad Cars 256
   5.5.4 Switch Box Routing 263
   5.5.5 Offline Equivalence Problem 264
   5.5.6 Rat in a Maze 268

5.6 References and Selected Readings 281
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.9</td>
<td>ADT and Class Extensions</td>
<td>392</td>
</tr>
<tr>
<td>8.9.1</td>
<td>Output</td>
<td>397</td>
</tr>
<tr>
<td>8.9.2</td>
<td>Delete</td>
<td>397</td>
</tr>
<tr>
<td>8.9.3</td>
<td>Height</td>
<td>397</td>
</tr>
<tr>
<td>8.9.4</td>
<td>Size</td>
<td>398</td>
</tr>
<tr>
<td>8.10</td>
<td>Applications</td>
<td>400</td>
</tr>
<tr>
<td>8.10.1</td>
<td>Placement of Signal Boosters</td>
<td>400</td>
</tr>
<tr>
<td>8.10.2</td>
<td>Online Equivalence Classes</td>
<td>405</td>
</tr>
<tr>
<td>8.11</td>
<td>References and Selected Readings</td>
<td>416</td>
</tr>
</tbody>
</table>

### CHAPTER 9 PRIORITY QUEUES 417

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>Introduction</td>
<td>419</td>
</tr>
<tr>
<td>9.2</td>
<td>Linear Lists</td>
<td>421</td>
</tr>
<tr>
<td>9.3</td>
<td>Heaps</td>
<td>421</td>
</tr>
<tr>
<td>9.3.1</td>
<td>Definitions</td>
<td>421</td>
</tr>
<tr>
<td>9.3.2</td>
<td>Insertion into a Max Heap</td>
<td>423</td>
</tr>
<tr>
<td>9.3.3</td>
<td>Deletion from a Max Heap</td>
<td>424</td>
</tr>
<tr>
<td>9.3.4</td>
<td>Max Heap Initialization</td>
<td>424</td>
</tr>
<tr>
<td>9.3.5</td>
<td>The Class MaxHeap</td>
<td>425</td>
</tr>
<tr>
<td>9.4</td>
<td>Leftist Trees</td>
<td>432</td>
</tr>
<tr>
<td>9.4.1</td>
<td>Height- and Weight-Biased Min and Max Leftist Trees</td>
<td>432</td>
</tr>
<tr>
<td>9.4.2</td>
<td>Insertion into a Max HBLT</td>
<td>435</td>
</tr>
<tr>
<td>9.4.3</td>
<td>Deletion from a Max HBLT</td>
<td>435</td>
</tr>
<tr>
<td>9.4.4</td>
<td>Melding Two Max HBLTs</td>
<td>435</td>
</tr>
<tr>
<td>9.4.5</td>
<td>Initialization</td>
<td>437</td>
</tr>
<tr>
<td>9.4.6</td>
<td>The Class MaxHBLT</td>
<td>438</td>
</tr>
<tr>
<td>9.5</td>
<td>Applications</td>
<td>444</td>
</tr>
<tr>
<td>9.5.1</td>
<td>Heap Sort</td>
<td>444</td>
</tr>
<tr>
<td>9.5.2</td>
<td>Machine Scheduling</td>
<td>444</td>
</tr>
<tr>
<td>9.5.3</td>
<td>Huffman Codes</td>
<td>450</td>
</tr>
<tr>
<td>9.6</td>
<td>References and Selected Readings</td>
<td>457</td>
</tr>
</tbody>
</table>

### CHAPTER 10 TOURNAMENT TREES 459

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>Introduction</td>
<td>461</td>
</tr>
<tr>
<td>10.2</td>
<td>The ADT WinnerTree</td>
<td>466</td>
</tr>
<tr>
<td>10.3</td>
<td>The Class WinnerTree</td>
<td>466</td>
</tr>
<tr>
<td>10.3.1</td>
<td>Representation</td>
<td>466</td>
</tr>
<tr>
<td>10.3.2</td>
<td>Class Specification</td>
<td>467</td>
</tr>
<tr>
<td>10.3.3</td>
<td>Constructor, Destructor, and Winner</td>
<td>467</td>
</tr>
<tr>
<td>10.3.4</td>
<td>Initializing a Winner Tree</td>
<td>468</td>
</tr>
<tr>
<td>10.3.5</td>
<td>Replaying Matches</td>
<td>471</td>
</tr>
<tr>
<td>10.4</td>
<td>Loser Trees</td>
<td>472</td>
</tr>
</tbody>
</table>
10.5 Applications 474
10.5.1 Bin Packing Using First Fit 472
10.5.2 Bin Packing Using Next Fit 480

CHAPTER 11 SEARCH TREES 485
11.1 Binary Search Trees 488
11.1.1 Definition 488
11.1.2 The ADTs BSTree and IndexedBSTree 490
11.1.3 The Class BSTree 491
11.1.4 Searching 492
11.1.5 Inserting an Element 493
11.1.6 Deleting an Element 493
11.1.7 The Class DBSTree 496
11.1.8 Height of a Binary Search Tree 496
11.2 AVL Trees 500
11.2.1 Definition 500
11.2.2 Height of an AVL Tree 501
11.2.3 Representation of an AVL Tree 501
11.2.4 Searching an AVL Search Tree 502
11.2.5 Inserting into an AVL Search Tree 502
11.2.6 Deletion from an AVL Search Tree 506
11.3 Red-Black Trees 510
11.3.1 Definition 510
11.3.2 Representation of a Red-Black Tree 512
11.3.3 Searching a Red-Black Tree 512
11.3.4 Inserting into a Red-Black Tree 513
11.3.5 Deletion from a Red-Black Tree 518
11.3.6 Implementation Considerations and Complexity 521
11.4 B-Trees 524
11.4.1 Indexed Sequential Access Method 524
11.4.2 m-way Search Trees 525
11.4.3 B-Trees of Order m 528
11.4.4 Height of a B-tree 529
11.4.5 Searching a B-tree 530
11.4.6 Inserting into a B-tree 530
11.4.7 Deletion from a B-tree 533
11.4.8 Node Structure 537
11.5 Applications 539
11.5.1 Histogramming 539
11.5.2 Best-Fit Bin Packing 543
11.5.3 Crossing Distribution 546
11.6 References and Selected Readings 553
CHAPTER 12  GRAPHS  555
12.1 Definitions  557
12.2 Applications  558
12.3 Properties  562
12.4 The ADTs Graph and Digraph  565
12.5 Representation of Graphs and Digraphs  567
   12.5.1 Adjacency Matrix  567
   12.5.2 Packed-Adjacency Lists  569
   12.5.3 Linked-Adjacency Lists  570
12.6 Representation of Networks  573
12.7 Class Definitions  575
   12.7.1 The Different Classes  575
   12.7.2 Adjacency-Matrix Classes  576
   12.7.3 An Extension to the Class Chain  580
   12.7.4 The Class LinkedBase  580
   12.7.5 Linked Classes  581
12.8 Graph Iterators  588
   12.8.1 Specification  588
   12.8.2 Iterator Functions for Adjacency-Matrix Representations  589
   12.8.3 Iterator Functions for Linked-Adjacency Lists  589
12.9 Language Features  592
   12.9.1 Virtual Functions and Polymorphism  592
   12.9.2 Pure Virtual Functions and Abstract Classes  595
   12.9.3 Virtual Base Classes  596
   12.9.4 Abstract Classes and Abstract Data Types  598
12.10 Graph Search Methods  600
   12.10.1 Breadth-First Search  600
   12.10.2 The Class Network  602
   12.10.3 Implementation of Network::BFS  602
   12.10.4 Complexity Analysis of Network::BFS  602
   12.10.5 Depth-First Search  605
12.11 Applications Revisited  607
   12.11.1 Finding a Path  607
   12.11.2 Connected Graphs and Components  609
   12.11.3 Spanning Trees  611

PART III  ALGORITHM-DESIGN METHODS

CHAPTER 13  THE GREEDY METHOD  615
13.1 Optimization Problems  617
13.2 The Greedy Method  618