Table of Contents

0. Organization of Chapters ---13

0.1 What Is This Book About? -- 13

0.2 Should I Buy This Book? --- 13

0.3 Organization of Chapters -- 14

0.4 Some Prerequisites --- 17

1. Introduction --18

1.1 Variables --- 18

1.2 Data Types --- 18

1.3 Data Structures --- 19

1.4 Abstract Data Types (ADTs) -- 19

1.5 What is an Algorithm? --- 19

1.6 Why the Analysis of Algorithms? --- 20

1.7 Goal of the Analysis of Algorithms -- 20

1.8 What is Running Time Analysis? -- 20

1.9 How to Compare Algorithms -- 20

1.10 What is Rate of Growth? --- 20

1.11 Commonly Used Rates of Growth -- 21

1.12 Types of Analysis -- 22

1.13 Asymptotic Notation -- 22

1.14 Big-O Notation -- 22

1.15 Omega-Ω Notation --- 24

1.16 Theta-Θ Notation -- 24

1.17 Why is it called Asymptotic Analysis? --- 25

1.18 Guidelines for Asymptotic Analysis -- 25

1.19 Properties of Notations -- 27

1.20 Commonly used Logarithms and Summations ------------------------------------- 27

1.21 Master Theorem for Divide and Conquer --- 27

1.22 Divide and Conquer Master Theorem: Problems & Solutions ------------------ 28

1.23 Master Theorem for Subtract and Conquer Recurrences ----------------------- 29

1.24 Variant of Subtraction and Conquer Master Theorem ---------------------------- 29

1.25 Method of Guessing and Confirming -- 29

1.26 Amortized Analysis -- 30

1.27 Algorithms Analysis: Problems & Solutions -- 31

2. Recursion and Backtracking --42

2.1 Introduction -- 42

2.2 What is Recursion? -- 42

2.3 Why Recursion? -- 42

2.4 Format of a Recursive Function --- 42

2.5 Recursion and Memory (Visualization) -- 43

2.6 Recursion versus Iteration ---------- 43
2.7 Notes on Recursion ---------- 44
2.8 Example Algorithms of Recursion ---------- 44
2.9 Recursion: Problems & Solutions ---------- 44
2.10 What is Backtracking?---------- 45
2.11 Example Algorithms of Backtracking ---------- 45
2.12 Backtracking: Problems & Solutions ---------- 45
3. Linked Lists ---------- 48
3.1 What is a Linked List?---------- 48
3.2 Linked Lists ADT ---------- 48
3.3 Why Linked Lists? ---------- 48
3.4 Arrays Overview ---------- 48
3.5 Comparison of Linked Lists with Arrays and Dynamic Arrays---------- 50
3.6 Singly Linked Lists---------- 50
3.7 Doubly Linked Lists ---------- 56
3.8 Circular Linked Lists ---------- 61
3.9 A Memory-efficient Doubly Linked List ---------- 66
3.10 Unrolled Linked Lists ---------- 68
3.11 Skip Lists ---------- 72
3.12 Linked Lists: Problems & Solutions ---------- 75
4. Stacks ---------- 96
4.1 What is a Stack? ---------- 96
4.2 How Stacks are Used ---------- 96
4.3 Stack ADT ---------- 97
4.4 Applications---------- 97
4.5 Implementation---------- 97
4.6 Comparison of Implementations----------101
4.7 Stacks: Problems & Solutions----------102
5. Queues ----------119
5.1 What is a Queue? ----------119
5.2 How are Queues Used ----------119
5.3 Queue ADT----------119
5.4 Exceptions ----------120
5.5 Applications----------120
5.6 Implementation----------120
5.7 Queues: Problems & Solutions ----------125
6. Trees ----------135
6.1 What is a Tree?----------135
6.2 Glossary ----------135
6.3 Binary Trees ----------136
6.4 Types of Binary Trees ----------137
6.5 Properties of Binary Trees ----------137

6.6 Binary Tree Traversals ---------- 139
6.7 Generic Trees (N-ary Trees) ---------- 159
6.8 Threaded Binary Tree Traversals (Stack or Queue-less Traversals) ---------- 166
6.9 Expression Trees ---------- 171
6.10 XOR Trees ---------- 174
6.11 Binary Search Trees (BSTs) ---------- 174
6.12 Balanced Binary Search Trees ---------- 189
6.13 AVL (Adelson-Velskii and Landis) Trees ---------- 189
6.14 Other Variations on Trees ---------- 206

7. Priority Queues and Heaps ---------- 211
7.1 What is a Priority Queue? ---------- 211
7.2 Priority Queue ADT ---------- 211
7.3 Priority Queue Applications ---------- 212
7.4 Priority Queue Implementations ---------- 212
7.5 Heaps and Binary Heaps ---------- 213
7.6 Binary Heaps ---------- 214
7.7 Heapsort ---------- 218
7.8 Priority Queues [Heaps]: Problems & Solutions ---------- 219

8. Disjoint Sets ADT ---------- 233
8.1 Introduction ---------- 233
8.2 Equivalence Relations and Equivalence Classes ---------- 233
8.3 Disjoint Sets ADT ---------- 234
8.4 Applications ---------- 234
8.5 Tradeoffs in Implementing Disjoint Sets ADT ---------- 234
8.8 Fast UNION Implementation (Slow FIND) ---------- 235
8.9 Fast UNION Implementations (Quick FIND) ---------- 237
8.10 Summary ---------- 240
8.11 Disjoint Sets: Problems & Solutions ---------- 240

9. Graph Algorithms ---------- 241
9.1 Introduction ---------- 241
9.2 Glossary ---------- 241
9.3 Applications of Graphs ---------- 244
9.4 Graph Representation ---------- 244
9.5 Graph Traversals ---------- 249
9.6 Topological Sort ---------- 255
9.7 Shortest Path Algorithms ---------- 257
9.8 Minimal Spanning Tree ---------- 262
9.9 Graph Algorithms: Problems & Solutions ---------- 266

10. Sorting ---------- 286
10.1 What is Sorting? ---------- 286
10.2 Why is Sorting Necessary? ---------- 286
10.3 Classification of Sorting Algorithms ---------- 286

6.6 Binary Tree Traversals ---------- 139
6.7 Generic Trees (N-ary Trees) ---------- 159
6.8 Threaded Binary Tree Traversals (Stack or Queue-less Traversals) ---------- 166
6.9 Expression Trees ---------- 171
6.10 XOR Trees ---------- 174
6.11 Binary Search Trees (BSTs) ---------- 174
6.12 Balanced Binary Search Trees ---------- 189
6.13 AVL (Adelson-Velskii and Landis) Trees ---------- 189
6.14 Other Variations on Trees ---------- 206

7. Priority Queues and Heaps ---------- 211
7.1 What is a Priority Queue? ---------- 211
7.2 Priority Queue ADT ---------- 211
7.3 Priority Queue Applications ---------- 212
7.4 Priority Queue Implementations ---------- 212
7.5 Heaps and Binary Heaps ---------- 213
7.6 Binary Heaps ---------- 214
7.7 Heapsort ---------- 218
7.8 Priority Queues [Heaps]: Problems & Solutions ---------- 219

8. Disjoint Sets ADT ---------- 233
8.1 Introduction ---------- 233
8.2 Equivalence Relations and Equivalence Classes ---------- 233
8.3 Disjoint Sets ADT ---------- 234
8.4 Applications ---------- 234
8.5 Tradeoffs in Implementing Disjoint Sets ADT ---------- 234
8.8 Fast UNION Implementation (Slow FIND) ---------- 235
8.9 Fast UNION Implementations (Quick FIND) ---------- 237
8.10 Summary ---------- 240
8.11 Disjoint Sets: Problems & Solutions ---------- 240

9. Graph Algorithms ---------- 241
9.1 Introduction ---------- 241
9.2 Glossary ---------- 241
9.3 Applications of Graphs ---------- 244
9.4 Graph Representation ---------- 244
9.5 Graph Traversals ---------- 249
9.6 Topological Sort ---------- 255
9.7 Shortest Path Algorithms ---------- 257
9.8 Minimal Spanning Tree ---------- 262
9.9 Graph Algorithms: Problems & Solutions ---------- 266

10. Sorting ---------- 286
10.1 What is Sorting? ---------- 286
10.2 Why is Sorting Necessary? ---------- 286
10.3 Classification of Sorting Algorithms ---------- 286

10.4 Other Classifications ---------- 287

10.5 Bubble Sort ---------- 287

10.6 Selection Sort ---------- 288

10.7 Insertion Sort ---------- 289

10.8 Shell Sort ---------- 290

10.9 Merge Sort ---------- 291

10.10 Heap Sort ---------- 293

10.11 Quick Sort ---------- 293

10.12 Tree Sort ---------- 295

10.13 Comparison of Sorting Algorithms ---------- 295

10.14 Linear Sorting Algorithms ---------- 296

10.15 Counting Sort ---------- 296

10.16 Bucket Sort (or Bin Sort) ---------- 296

10.17 Radix Sort ---------- 297

10.18 Topological Sort ---------- 298

10.19 External Sorting ---------- 298

10.20 Sorting: Problems & Solutions ---------- 299

11. Searching ---------- 309

11.1 What is Searching? ---------- 309

11.2 Why do we need Searching? ---------- 309

11.3 Types of Searching ---------- 309

11.4 Unordered Linear Search ---------- 309

11.5 Sorted/Ordered Linear Search ---------- 310

11.6 Binary Search ---------- 310

11.7 Comparing Basic Searching Algorithms ---------- 311

11.8 Symbol Tables and Hashing ---------- 311

11.9 String Searching Algorithms ---------- 311

11.10 Searching: Problems & Solutions ---------- 311

12. Selection Algorithms [Medians] ---------- 333

12.1 What are Selection Algorithms? ---------- 333

12.2 Selection by Sorting ---------- 333

12.3 Partition-based Selection Algorithm ---------- 333

12.4 Linear Selection Algorithm - Median of Medians Algorithm ---------- 333

12.5 Finding the K Smallest Elements in Sorted Order ---------- 333

12.6 Selection Algorithms: Problems & Solutions ---------- 334

13. Symbol Tables ---------- 343

13.1 Introduction ---------- 343

13.2 What are Symbol Tables? ---------- 343

13.3 Symbol Table Implementations ---------- 343

13.4 Comparison Table of Symbols for Implementations ---------- 344

14. Hashing ---------- 345

14.1 What is Hashing? ---------- 345

14.2 Why Hashing? ---------- 345
14.3 HashTable ADT ---------- 345
14.4 Understanding Hashing ---------- 345
14.5 Components of Hashing ---------- 346
14.6 Hash Table ---------- 346
14.7 Hash Function ---------- 347
14.8 Load Factor ---------- 348
14.9 Collisions ---------- 348
14.10 Collision Resolution Techniques ---------- 348
14.11 Separate Chaining ---------- 348
14.12 Open Addressing ---------- 348
14.13 Comparison of Collision Resolution Techniques ---------- 350
14.14 How Hashing Gets O(1) Complexity ---------- 350
14.15 Hashing Techniques ---------- 351
14.16 Problems for which Hash Tables are not suitable ---------- 351
14.17 Bloom Filters ---------- 351
14.18 Hashing: Problems & Solutions ---------- 352

15. String Algorithms ---------- 360
15.1 Introduction ---------- 360
15.2 String Matching Algorithms ---------- 360
15.3 Brute Force Method ---------- 360
15.4 Robin-Karp String Matching Algorithm ---------- 361
15.5 String Matching with Finite Automata ---------- 362
15.6 KMP Algorithm ---------- 363
15.7 Boyce-Moore Algorithm ---------- 366
15.8 Data Structures for Storing Strings ---------- 367
15.9 Hash Tables for Strings ---------- 367
15.10 Binary Search Trees for Strings ---------- 367
15.11 Tries ---------- 367
15.12 Ternary Search Trees ---------- 369
15.13 Comparing BSTs, Tries and TSTs ---------- 375
15.14 Suffix Trees ---------- 375
15.15 String Algorithms: Problems & Solutions ---------- 378

16. Algorithms Design Techniques ---------- 385
16.1 Introduction ---------- 385
16.2 Classification ---------- 385
16.3 Classification by Implementation Method ---------- 385
16.4 Classification by Design Method ---------- 386
16.5 Other Classifications ---------- 387

17. Greedy Algorithms ---------- 388
17.1 Introduction ---------- 388
17.2 Greedy Strategy ---------- 388

17.3 Elements of Greedy Algorithms -- 388
17.4 Does Greedy Always Work? -- 388
17.5 Advantages and Disadvantages of Greedy Method -- 388
17.6 Greedy Applications -- 389
17.7 Understanding Greedy Technique -- 389
17.8 Greedy Algorithms: Problems & Solutions -- 391

18. Divide and Conquer Algorithms -- 397

18.1 Introduction -- 397
18.2 What is Divide and Conquer Strategy? -- 397
18.3 Does Divide and Conquer Always Work? -- 397
18.4 Divide and Conquer Visualization -- 397
18.5 Understanding Divide and Conquer -- 398
18.6 Advantages of Divide and Conquer -- 398
18.7 Disadvantages of Divide and Conquer -- 399
18.8 Master Theorem -- 399
18.9 Divide and Conquer Applications -- 399
18.10 Divide and Conquer: Problems & Solutions -- 399

19. Dynamic Programming -- 411

19.1 Introduction -- 411
19.2 What is Dynamic Programming Strategy? -- 411
19.3 Properties of Dynamic Programming Strategy -- 411
19.4 Can Dynamic Programming Solve All Problems? -- 411
19.5 Dynamic Programming Approaches -- 411
19.6 Examples of Dynamic Programming Algorithms -- 412
19.7 Understanding Dynamic Programming -- 412
19.8 Longest Common Subsequence -- 414
19.9 Dynamic Programming: Problems & Solutions -- 416

20. Complexity Classes -- 448

20.1 Introduction -- 448
20.2 Polynomial/Exponential Time -- 448
20.3 What is a Decision Problem? -- 448
20.4 Decision Procedure -- 449
20.5 What is a Complexity Class? -- 449
20.6 Types of Complexity Classes -- 449
20.7 Reductions -- 451
20.8 Complexity Classes: Problems & Solutions -- 453

21. Miscellaneous Concepts -- 455

21.1 Introduction -- 455
21.2 Hacks on Bitwise Programming -- 455
21.3 Other Programming Questions with Solutions -- 459

References -- 466