

Contents

Preface *xiii*

1	Introduction	<i>1</i>
	1.1 Algorithms	<i>1</i>
	1.2 Analyzing algorithms	<i>6</i>
	1.3 Designing algorithms	<i>11</i>
	1.4 Summary	<i>16</i>

I Mathematical Foundations

	Introduction	<i>21</i>
2	Growth of Functions	<i>23</i>
	2.1 Asymptotic notation	<i>23</i>
	2.2 Standard notations and common functions	<i>32</i>
3	Summations	<i>42</i>
	3.1 Summation formulas and properties	<i>42</i>
	3.2 Bounding summations	<i>46</i>
4	Recurrences	<i>53</i>
	4.1 The substitution method	<i>54</i>
	4.2 The iteration method	<i>58</i>
	4.3 The master method	<i>61</i>
*	4.4 Proof of the master theorem	<i>64</i>
5	Sets, Etc.	<i>77</i>
	5.1 Sets	<i>77</i>
	5.2 Relations	<i>81</i>
	5.3 Functions	<i>84</i>
	5.4 Graphs	<i>86</i>
	5.5 Trees	<i>91</i>

- 6 Counting and Probability 99**
 - 6.1 Counting 99
 - 6.2 Probability 104
 - 6.3 Discrete random variables 111
 - 6.4 The geometric and binomial distributions 115
 - * 6.5 The tails of the binomial distribution 121
 - 6.6 Probabilistic analysis 126

Sorting and Order Statistics

- Introduction 137**
- 7 Heapsort 140**
 - 7.1 Heaps 140
 - 7.2 Maintaining the heap property 142
 - 7.3 Building a heap 145
 - 7.4 The heapsort algorithm 147
 - 7.5 Priority queues 149
- 8 Quicksort 153**
 - 8.1 Description of quicksort 153
 - 8.2 Performance of quicksort 156
 - 8.3 Randomized versions of quicksort 161
 - 8.4 Analysis of quicksort 163
- 9 Sorting in Linear Time 172**
 - 9.1 Lower bounds for sorting 172
 - 9.2 Counting sort 175
 - 9.3 Radix sort 178
 - 9.4 Bucket sort 180
- 10 Medians and Order Statistics 185**
 - 10.1 Minimum and maximum 185
 - 10.2 Selection in expected linear time 187
 - 10.3 Selection in worst-case linear time 189

III Data Structures

- Introduction 197**
- 11 Elementary Data Structures 200**
 - 11.1 Stacks and queues 200
 - 11.2 Linked lists 204
 - 11.3 Implementing pointers and objects 209
 - 11.4 Representing rooted trees 213
- 12 Hash Tables 219**
 - 12.1 Direct-address tables 219
 - 12.2 Hash tables 221
 - 12.3 Hash functions 226
 - 12.4 Open addressing 232
- 13 Binary Search Trees 244**
 - 13.1 What is a binary search tree? 244
 - 13.2 Querying a binary search tree 246
 - 13.3 Insertion and deletion 250
 - * 13.4 Randomly built binary search trees 254
- 14 Red-Black Trees 263**
 - 14.1 Properties of red-black trees 263
 - 14.2 Rotations 265
 - 14.3 Insertion 268
 - 14.4 Deletion 272
- 15 Augmenting Data Structures 281**
 - 15.1 Dynamic order statistics 281
 - 15.2 How to augment a data structure 287
 - 15.3 Interval trees 290

IV Advanced Design and Analysis Techniques

- Introduction 299**
- 16 Dynamic Programming 301**
 - 16.1 Matrix-chain multiplication 302
 - 16.2 Elements of dynamic programming 309
 - 16.3 Longest common subsequence 314
 - 16.4 Optimal polygon triangulation 320

- 17 Greedy Algorithms 329**
 - 17.1 An activity-selection problem 329
 - 17.2 Elements of the greedy strategy 333
 - 17.3 Huffman codes 337
 - ★ 17.4 Theoretical foundations for greedy methods 345
 - ★ 17.5 A task-scheduling problem 350
- 18 Amortized Analysis 356**
 - 18.1 The aggregate method 357
 - 18.2 The accounting method 360
 - 18.3 The potential method 363
 - 18.4 Dynamic tables 367

Advanced Data Structures

- Introduction 379**
- 19 B-Trees 381**
 - 19.1 Definition of B-trees 384
 - 19.2 Basic operations on B-trees 387
 - 19.3 Deleting a key from a B-tree 395
- 20 Binomial Heaps 400**
 - 20.1 Binomial trees and binomial heaps 401
 - 20.2 Operations on binomial heaps 406
- 21 Fibonacci Heaps 420**
 - 21.1 Structure of Fibonacci heaps 421
 - 21.2 Mergeable-heap operations 423
 - 21.3 Decreasing a key and deleting a node 431
 - 21.4 Bounding the maximum degree 435
- 22 Data Structures for Disjoint Sets 440**
 - 22.1 Disjoint-set operations 440
 - 22.2 Linked-list representation of disjoint sets 443
 - 22.3 Disjoint-set forests 446
 - ★ 22.4 Analysis of union by rank with path compression 450

VI Graph Algorithms

- Introduction 463**
- 23 Elementary Graph Algorithms 465**
 - 23.1 Representations of graphs 465
 - 23.2 Breadth-first search 469
 - 23.3 Depth-first search 477
 - 23.4 Topological sort 485
 - 23.5 Strongly connected components 488
- 24 Minimum Spanning Trees 498**
 - 24.1 Growing a minimum spanning tree 499
 - 24.2 The algorithms of Kruskal and Prim 504
- 25 Single-Source Shortest Paths 514**
 - 25.1 Shortest paths and relaxation 518
 - 25.2 Dijkstra's algorithm 527
 - 25.3 The Bellman-Ford algorithm 532
 - 25.4 Single-source shortest paths in directed acyclic graphs 536
 - 25.5 Difference constraints and shortest paths 539
- 26 All-Pairs Shortest Paths 550**
 - 26.1 Shortest paths and matrix multiplication 552
 - 26.2 The Floyd-Warshall algorithm 558
 - 26.3 Johnson's algorithm for sparse graphs 565
 - ★ 26.4 A general framework for solving path problems in directed graphs 570
- 27 Maximum Flow 579**
 - 27.1 Flow networks 580
 - 27.2 The Ford-Fulkerson method 587
 - 27.3 Maximum bipartite matching 600
 - ★ 27.4 Preflow-push algorithms 605
 - ★ 27.5 The lift-to-front algorithm 615

Selected Topics

	Introduction	631
28	Sorting Networks	634
	28.1 Comparison networks	634
	28.2 The zero-one principle	639
	28.3 A bitonic sorting network	642
	28.4 A merging network	646
	28.5 A sorting network	648
29	Arithmetic Circuits	654
	29.1 Combinational circuits	655
	29.2 Addition circuits	660
	29.3 Multiplication circuits	671
	29.4 Clocked circuits	678
30	Algorithms for Parallel Computers	688
	30.1 Pointer jumping	692
	30.2 CRCW algorithms versus EREW algorithms	701
	30.3 Brent's theorem and work efficiency	709
★	30.4 Work-efficient parallel prefix computation	714
	30.5 Deterministic symmetry breaking	720
31	Matrix Operations	730
	31.1 Properties of matrices	730
	31.2 Strassen's algorithm for matrix multiplication	739
★	31.3 Algebraic number systems and boolean matrix multiplication	745
	31.4 Solving systems of linear equations	749
	31.5 Inverting matrices	762
	31.6 Symmetric positive-definite matrices and least-squares approximation	766
32	Polynomials and the FFT	776
	32.1 Representation of polynomials	778
	32.2 The DFT and FFT	783
	32.3 Efficient FFT implementations	791
33	Number-Theoretic Algorithms	801
	33.1 Elementary number-theoretic notions	802
	33.2 Greatest common divisor	808
	33.3 Modular arithmetic	814

	33.4 Solving modular linear equations	820
	33.5 The Chinese remainder theorem	823
	33.6 Powers of an element	827
	33.7 The RSA public-key cryptosystem	831
★	33.8 Primality testing	837
★	33.9 Integer factorization	844
34	String Matching	853
	34.1 The naive string-matching algorithm	855
	34.2 The Rabin-Karp algorithm	857
	34.3 String matching with finite automata	862
	34.4 The Knuth-Morris-Pratt algorithm	869
★	34.5 The Boyer-Moore algorithm	876
35	Computational Geometry	886
	35.1 Line-segment properties	887
	35.2 Determining whether any pair of segments intersects	892
	35.3 Finding the convex hull	898
	35.4 Finding the closest pair of points	908
36	NP-Completeness	916
	36.1 Polynomial time	917
	36.2 Polynomial-time verification	924
	36.3 NP-completeness and reducibility	929
	36.4 NP-completeness proofs	939
	36.5 NP-complete problems	946
37	Approximation Algorithms	964
	37.1 The vertex-cover problem	966
	37.2 The traveling-salesman problem	969
	37.3 The set-covering problem	974
	37.4 The subset-sum problem	978

Bibliography 987
Index 997