

Contents

Preface *xiii*

I Foundations

Introduction 3

1 The Role of Algorithms in Computing 5

- 1.1 Algorithms 5
- 1.2 Algorithms as a technology 12

2 Getting Started 17

- 2.1 Insertion sort 17
- 2.2 Analyzing algorithms 25
- 2.3 Designing algorithms 34

3 Characterizing Running Times 49

- 3.1 O -notation, Ω -notation, and Θ -notation 50
- 3.2 Asymptotic notation: formal definitions 53
- 3.3 Standard notations and common functions 63

4 Divide-and-Conquer 76

- 4.1 Multiplying square matrices 80
- 4.2 Strassen's algorithm for matrix multiplication 85
- 4.3 The substitution method for solving recurrences 90
- 4.4 The recursion-tree method for solving recurrences 95
- 4.5 The master method for solving recurrences 101
- ★ 4.6 Proof of the continuous master theorem 107
- ★ 4.7 Akra-Bazzi recurrences 115

5	Probabilistic Analysis and Randomized Algorithms	126
5.1	The hiring problem	126
5.2	Indicator random variables	130
5.3	Randomized algorithms	134
★ 5.4	Probabilistic analysis and further uses of indicator random variables	140

II *Sorting and Order Statistics*

	Introduction	157
6	Heapsort	161
6.1	Heaps	161
6.2	Maintaining the heap property	164
6.3	Building a heap	167
6.4	The heapsort algorithm	170
6.5	Priority queues	172
7	Quicksort	182
7.1	Description of quicksort	183
7.2	Performance of quicksort	187
7.3	A randomized version of quicksort	191
7.4	Analysis of quicksort	193
8	Sorting in Linear Time	205
8.1	Lower bounds for sorting	205
8.2	Counting sort	208
8.3	Radix sort	211
8.4	Bucket sort	215
9	Medians and Order Statistics	227
9.1	Minimum and maximum	228
9.2	Selection in expected linear time	230
9.3	Selection in worst-case linear time	236

III *Data Structures*

	Introduction	249
10	Elementary Data Structures	252
10.1	Simple array-based data structures: arrays, matrices, stacks, queues	252
10.2	Linked lists	258
10.3	Representing rooted trees	265

11	Hash Tables	272
11.1	Direct-address tables	273
11.2	Hash tables	275
11.3	Hash functions	282
11.4	Open addressing	293
11.5	Practical considerations	301
12	Binary Search Trees	312
12.1	What is a binary search tree?	312
12.2	Querying a binary search tree	316
12.3	Insertion and deletion	321
13	Red-Black Trees	331
13.1	Properties of red-black trees	331
13.2	Rotations	335
13.3	Insertion	338
13.4	Deletion	346

IV Advanced Design and Analysis Techniques

	Introduction	361
14	Dynamic Programming	362
14.1	Rod cutting	363
14.2	Matrix-chain multiplication	373
14.3	Elements of dynamic programming	382
14.4	Longest common subsequence	393
14.5	Optimal binary search trees	400
15	Greedy Algorithms	417
15.1	An activity-selection problem	418
15.2	Elements of the greedy strategy	426
15.3	Huffman codes	431
15.4	Offline caching	440
16	Amortized Analysis	448
16.1	Aggregate analysis	449
16.2	The accounting method	453
16.3	The potential method	456
16.4	Dynamic tables	460

V Advanced Data Structures

	Introduction	477
17	Augmenting Data Structures	480
	17.1 Dynamic order statistics	480
	17.2 How to augment a data structure	486
	17.3 Interval trees	489
18	B-Trees	497
	18.1 Definition of B-trees	501
	18.2 Basic operations on B-trees	504
	18.3 Deleting a key from a B-tree	513
19	Data Structures for Disjoint Sets	520
	19.1 Disjoint-set operations	520
	19.2 Linked-list representation of disjoint sets	523
	19.3 Disjoint-set forests	527
★	19.4 Analysis of union by rank with path compression	531

VI Graph Algorithms

	Introduction	547
20	Elementary Graph Algorithms	549
	20.1 Representations of graphs	549
	20.2 Breadth-first search	554
	20.3 Depth-first search	563
	20.4 Topological sort	573
	20.5 Strongly connected components	576
21	Minimum Spanning Trees	585
	21.1 Growing a minimum spanning tree	586
	21.2 The algorithms of Kruskal and Prim	591
22	Single-Source Shortest Paths	604
	22.1 The Bellman-Ford algorithm	612
	22.2 Single-source shortest paths in directed acyclic graphs	616
	22.3 Dijkstra's algorithm	620
	22.4 Difference constraints and shortest paths	626
	22.5 Proofs of shortest-paths properties	633

- 23 All-Pairs Shortest Paths 646**
 - 23.1 Shortest paths and matrix multiplication 648
 - 23.2 The Floyd-Warshall algorithm 655
 - 23.3 Johnson's algorithm for sparse graphs 662
- 24 Maximum Flow 670**
 - 24.1 Flow networks 671
 - 24.2 The Ford-Fulkerson method 676
 - 24.3 Maximum bipartite matching 693
- 25 Matchings in Bipartite Graphs 704**
 - 25.1 Maximum bipartite matching (revisited) 705
 - 25.2 The stable-marriage problem 716
 - 25.3 The Hungarian algorithm for the assignment problem 723

VII Selected Topics

- Introduction 745**
- 26 Parallel Algorithms 748**
 - 26.1 The basics of fork-join parallelism 750
 - 26.2 Parallel matrix multiplication 770
 - 26.3 Parallel merge sort 775
- 27 Online Algorithms 791**
 - 27.1 Waiting for an elevator 792
 - 27.2 Maintaining a search list 795
 - 27.3 Online caching 802
- 28 Matrix Operations 819**
 - 28.1 Solving systems of linear equations 819
 - 28.2 Inverting matrices 833
 - 28.3 Symmetric positive-definite matrices and least-squares approximation 838
- 29 Linear Programming 850**
 - 29.1 Linear programming formulations and algorithms 853
 - 29.2 Formulating problems as linear programs 860
 - 29.3 Duality 866
- 30 Polynomials and the FFT 877**
 - 30.1 Representing polynomials 879
 - 30.2 The DFT and FFT 885
 - 30.3 FFT circuits 894

31	Number-Theoretic Algorithms	903
31.1	Elementary number-theoretic notions	904
31.2	Greatest common divisor	911
31.3	Modular arithmetic	916
31.4	Solving modular linear equations	924
31.5	The Chinese remainder theorem	928
31.6	Powers of an element	932
31.7	The RSA public-key cryptosystem	936
★ 31.8	Primality testing	942
32	String Matching	957
32.1	The naive string-matching algorithm	960
32.2	The Rabin-Karp algorithm	962
32.3	String matching with finite automata	967
★ 32.4	The Knuth-Morris-Pratt algorithm	975
32.5	Suffix arrays	985
33	Machine-Learning Algorithms	1003
33.1	Clustering	1005
33.2	Multiplicative-weights algorithms	1015
33.3	Gradient descent	1022
34	NP-Completeness	1042
34.1	Polynomial time	1048
34.2	Polynomial-time verification	1056
34.3	NP-completeness and reducibility	1061
34.4	NP-completeness proofs	1072
34.5	NP-complete problems	1080
35	Approximation Algorithms	1104
35.1	The vertex-cover problem	1106
35.2	The traveling-salesperson problem	1109
35.3	The set-covering problem	1115
35.4	Randomization and linear programming	1119
35.5	The subset-sum problem	1124

VIII Appendix: Mathematical Background

	Introduction	1139
A	Summations	1140
A.1	Summation formulas and properties	1140
A.2	Bounding summations	1145

B	Sets, Etc.	1153
B.1	Sets	1153
B.2	Relations	1158
B.3	Functions	1161
B.4	Graphs	1164
B.5	Trees	1169
C	Counting and Probability	1178
C.1	Counting	1178
C.2	Probability	1184
C.3	Discrete random variables	1191
C.4	The geometric and binomial distributions	1196
★ C.5	The tails of the binomial distribution	1203
D	Matrices	1214
D.1	Matrices and matrix operations	1214
D.2	Basic matrix properties	1219

Bibliography	1227
---------------------	-------------

Index	1251
--------------	-------------

¹ To understand many of the ways in which algorithms influence our daily lives, see the book by Fry [162].