

Contents

77	2.3	Search Primitives	82
81	2.4	Priority Queues	83
85	2.5	War Story: Striping (Radix Search)	85
90	2.6	Hashing and Strings	90
93	2.7	Specialized Data Structures: A Not So Simple List	93
96	2.8	War Story: String 'em Up	96
98	2.9	Exercises	98
102	3	Sorting and Searching	102
104	3.1	Applications of Sorting	104
107	3.2	Primitives of Sorting	107
110	3.3	Heapsort: Fast Sorting via Data Structure Management	110
114	3.4	War Story: Give me a List to Merge (and a Good One)	114
120	3.5	Mergesort: Sorting by Divide-and-Conquer's Standards	120
123	3.6	Quicksort: Sorting by Randomization's General Impact	123
129	3.7	Distribution Sort: Sorting by Distribution's General Impact	129
132	3.8	War Story: Slams for the Defense	132
137	3.9	Exercises	137
137	I	Practical Algorithm Design	1
137	1	Introduction to Algorithm Design	3
142	1.1	Robot Tour Optimization	5
145	1.2	Selecting the Right Jobs	9
148	1.3	Reasoning about Correctness	11
152	1.4	Modeling the Problem	19
156	1.5	About the War Stories	22
158	1.6	War Story: Psychic Modeling	23
160	1.7	Exercises	27
163	2	Algorithm Analysis	31
169	2.1	The RAM Model of Computation	31
172	2.2	The Big Oh Notation	34
175	2.3	Growth Rates and Dominance Relations	37
180	2.4	Working with the Big Oh	40
181	2.5	Reasoning About Efficiency	41
184	2.6	Logarithms and Their Applications	46
189	2.7	Properties of Logarithms	50
190	2.8	War Story: Mystery of the Pyramids	51
195	2.9	Advanced Analysis (*)	54
197	2.10	Exercises	57
197	3	Data Structures	65
202	3.1	Contiguous vs. Linked Data Structures	66

3.2	Stacks and Queues	71
3.3	Dictionaries	72
3.4	Binary Search Trees	77
3.5	Priority Queues	83
3.6	War Story: Stripping Triangulations	85
3.7	Hashing and Strings	89
3.8	Specialized Data Structures	93
3.9	War Story: String 'em Up	94
3.10	Exercises	98
4	Sorting and Searching	103
4.1	Applications of Sorting	104
4.2	Pragmatics of Sorting	107
4.3	Heapsort: Fast Sorting via Data Structures	108
4.4	War Story: Give me a Ticket on an Airplane	118
4.5	Mergesort: Sorting by Divide-and-Conquer	120
4.6	Quicksort: Sorting by Randomization	123
4.7	Distribution Sort: Sorting via Bucketing	129
4.8	War Story: Skiena for the Defense	131
4.9	Binary Search and Related Algorithms	132
4.10	Divide-and-Conquer	135
4.11	Exercises	139
5	Graph Traversal	145
5.1	Flavors of Graphs	146
5.2	Data Structures for Graphs	151
5.3	War Story: I was a Victim of Moore's Law	155
5.4	War Story: Getting the Graph	158
5.5	Traversing a Graph	161
5.6	Breadth-First Search	162
5.7	Applications of Breadth-First Search	166
5.8	Depth-First Search	169
5.9	Applications of Depth-First Search	172
5.10	Depth-First Search on Directed Graphs	178
5.11	Exercises	184
6	Weighted Graph Algorithms	191
6.1	Minimum Spanning Trees	192
6.2	War Story: Nothing but Nets	202
6.3	Shortest Paths	205
6.4	War Story: Dialing for Documents	212
6.5	Network Flows and Bipartite Matching	217
6.6	Design Graphs, Not Algorithms	222
6.7	Exercises	225

7	Combinatorial Search and Heuristic Methods	230
7.1	Backtracking	231
7.2	Search Pruning	238
7.3	Sudoku	239
7.4	War Story: Covering Chessboards	244
7.5	Heuristic Search Methods	247
7.6	War Story: Only it is Not a Radio	260
7.7	War Story: Annealing Arrays	263
7.8	Other Heuristic Search Methods	266
7.9	Parallel Algorithms	267
7.10	War Story: Going Nowhere Fast	268
7.11	Exercises	270
8	Dynamic Programming	273
8.1	Caching vs. Computation	274
8.2	Approximate String Matching	280
8.3	Longest Increasing Sequence	289
8.4	War Story: Evolution of the Lobster	291
8.5	The Partition Problem	294
8.6	Parsing Context-Free Grammars	298
8.7	Limitations of Dynamic Programming: TSP	301
8.8	War Story: What's Past is Prolog	304
8.9	War Story: Text Compression for Bar Codes	307
8.10	Exercises	310
9	Intractable Problems and Approximation Algorithms	316
9.1	Problems and Reductions	317
9.2	Reductions for Algorithms	319
9.3	Elementary Hardness Reductions	323
9.4	Satisfiability	328
9.5	Creative Reductions	330
9.6	The Art of Proving Hardness	334
9.7	War Story: Hard Against the Clock	337
9.8	War Story: And Then I Failed	339
9.9	P vs. NP	341
9.10	Dealing with NP-complete Problems	344
9.11	Exercises	350
10	How to Design Algorithms	356
II	The Hitchhiker's Guide to Algorithms	361
11	A Catalog of Algorithmic Problems	363

12 Data Structures	366
12.1 Dictionaries	367
12.2 Priority Queues	373
12.3 Suffix Trees and Arrays	377
12.4 Graph Data Structures	381
12.5 Set Data Structures	385
12.6 Kd-Trees	389
13 Numerical Problems	393
13.1 Solving Linear Equations	395
13.2 Bandwidth Reduction	398
13.3 Matrix Multiplication	401
13.4 Determinants and Permanents	404
13.5 Constrained and Unconstrained Optimization	407
13.6 Linear Programming	411
13.7 Random Number Generation	415
13.8 Factoring and Primality Testing	420
13.9 Arbitrary-Precision Arithmetic	423
13.10 Knapsack Problem	427
13.11 Discrete Fourier Transform	431
14 Combinatorial Problems	434
14.1 Sorting	436
14.2 Searching	441
14.3 Median and Selection	445
14.4 Generating Permutations	448
14.5 Generating Subsets	452
14.6 Generating Partitions	456
14.7 Generating Graphs	460
14.8 Calendrical Calculations	465
14.9 Job Scheduling	468
14.10 Satisfiability	472
15 Graph Problems: Polynomial-Time	475
15.1 Connected Components	477
15.2 Topological Sorting	481
15.3 Minimum Spanning Tree	484
15.4 Shortest Path	489
15.5 Transitive Closure and Reduction	495
15.6 Matching	498
15.7 Eulerian Cycle/Chinese Postman	502
15.8 Edge and Vertex Connectivity	505
15.9 Network Flow	509
15.10 Drawing Graphs Nicely	513

15.11	Drawing Trees	517
15.12	Planarity Detection and Embedding	520
16	Graph Problems: Hard Problems	523
16.1	Clique	525
16.2	Independent Set	528
16.3	Vertex Cover	530
16.4	Traveling Salesman Problem	533
16.5	Hamiltonian Cycle	538
16.6	Graph Partition	541
16.7	Vertex Coloring	544
16.8	Edge Coloring	548
16.9	Graph Isomorphism	550
16.10	Steiner Tree	555
16.11	Feedback Edge/Vertex Set	559
17	Computational Geometry	562
17.1	Robust Geometric Primitives	564
17.2	Convex Hull	568
17.3	Triangulation	572
17.4	Voronoi Diagrams	576
17.5	Nearest Neighbor Search	580
17.6	Range Search	584
17.7	Point Location	587
17.8	Intersection Detection	591
17.9	Bin Packing	595
17.10	Medial-Axis Transform	598
17.11	Polygon Partitioning	601
17.12	Simplifying Polygons	604
17.13	Shape Similarity	607
17.14	Motion Planning	610
17.15	Maintaining Line Arrangements	614
17.16	Minkowski Sum	617
18	Set and String Problems	620
18.1	Set Cover	621
18.2	Set Packing	625
18.3	String Matching	628
18.4	Approximate String Matching	631
18.5	Text Compression	637
18.6	Cryptography	641
18.7	Finite State Machine Minimization	646
18.8	Longest Common Substring/Subsequence	650
18.9	Shortest Common Superstring	654

19 Algorithmic Resources	657
19.1 Software Systems	657
19.2 Data Sources	663
19.3 Online Bibliographic Resources	663
19.4 Professional Consulting Services	664
Bibliography	665
Index	709
1.1 Drawing Trees	711
1.2 Binary Trees	712
1.3 Graphs	713
1.4 Hamiltonian Cycle	715
1.5 Traveling Salesman Problem	716
1.6 Vertex Cover	717
1.7 Independent Set	718
1.8 Graph Isomorphism	719
1.9 Graph Automorphism	720
1.10 Steiner Tree	721
1.11 Feedback Edge/Vertex Set	722
1.12 Random Number Generation	723
1.13 Factoring and Primality Testing	724
1.14 Robust Geometric Primitives	725
1.15 Arbitrary-Precision Arithmetic	726
1.16 Knapsack Problem	727
1.17 Discrete Fourier Transform	728
1.18 Voronoi Diagrams	729
1.19 Nearest Neighbor Search	730
1.20 Range Search	731
1.21 Point Location	732
1.22 Intersection Detection	733
1.23 Bin Packing	734
1.24 Median-Axis Transform	735
1.25 Polygon Partitioning	736
1.26 Supplying Polygons	737
1.27 Shape Similarity	738
1.28 Motion Planning	739
1.29 Maintaining Line Arrangements	740
1.30 Minkowski Sum	741
1.31 Graph Problems: Polynomials	742
1.32 Set and String Problems	743
1.33 Connected Components	744
1.34 Topological Sorting	745
1.35 Minimum Spanning Trees	746
1.36 Shortest Path	747
1.37 String Matching	748
1.38 Approximate String Matching	749
1.39 Text Compression	750
1.40 Eulerian Cycle/Circuit	751
1.41 Finite State Machine Minimization	752
1.42 Edge and Vertex Cuts	753
1.43 Longest Common Substring/Subsequence	754
1.44 Shortest Common Superstring	755