

If we had an index file, we could look it up in the index file under "index file".

— Tegan Jovanka [Janet Fielding], "Castrovalva (Part 1)",
Doctor Who, Season 19 (January 4, 1982)

I started with the phone book. Looking up "mensa" was not going to be easy, what with having to follow the strict alphabetizing rules that are so common nowadays. I prefer a softer, more fuzzy alphabetizing scheme, one that allows the mind to float free and "happen" upon the word. There is pride in that. The dictionary is a perfect example of over-alphabetization, with its harsh rules and every little word neatly in place. It almost makes me never want to eat again.

— Steve Martin, "How I Joined Mensa", *The New Yorker*, July 21, 1997.

Index

For some topics with multiple references, bold page numbers indicate the primary reference. Humans and pseudocode are indexed separately.

1-IN-3SAT, 405
2048 (game), 407
2COLOR, 420
2PARTITION, 419
2SAT, 406, 420
3CNF formula, 388
3COLOR, 395
 reduction from 3SAT, 395
3PARTITION, 405
3SAT, 388
 reduction from CIRCUITSAT,
 388, 393, 395
 reduction to 3COLOR, 395
 reduction to
 DIRECTEDHAMCYCLE, 400
 reduction to MAXINDSET, 390,
 393, 395
 rule of three, 395, 405, 408
4:20, 161
academic job market, 170
active vertex (depth-first search),
 228
acyclic graph (= forest), 191
acyclic maximum flow, 339, 345

- ad-hoc networks, 375
- addition chains, 94
 - increment and double only, 181
- additional recurrence parameter,
 - 295, 313, 318
- adjacency matrix, 360
- airline scheduling, 361
- alternating path, 356
- amortized analysis, 264, 267
- The Announcer's Test, 17
- antanairesis*, *see* Euclid's algorithm
- Antarctica, 123, 143, 181
- APSP, *see* shortest paths, all-pairs
- arbitrage, 321
- arithmetic takes time, 104, 325
- arpdonaptai*, 7, 317
- arrow notation ($a \uparrow^b c$), 414
- articulation point, *see* cut vertex
- artificial source vertex, 227, 312, 355,
 - 366
- Āryabhaṭa's pulverizer, *see* Euclid's algorithm
- assignment, *see* matching, tuple selection
- augmenting path, 332
- "average case" analysis, 31

- B*-tree, 147
- back edge (depth-first search), 229
- backtracking, 71
 - recursive brute force, 80
 - sequence of decisions, 79
 - summary of past decisions, 79
- backward induction, *see* dynamic programming
- Baguenaudier, 45
- balanced brackets, 148, 184
- "The Barley Mow", 18
- base case, 23
- baseball elimination, 363
- BEAMILLIONAIREANDNEVERPAYTAXES
 - reduction from
 - BEAMILLIONAIRE, 10
- Bellman-Ford, 289
 - as dynamic programming, 294
 - Moore's variant, 292
- Bellman-Kalaba, *see* Bellman-Ford
- Bellman-Shimbel, *see* Bellman-Ford
- Bellman's equation, *see* recurrence
- "best case" analysis, 31
- best-first search, 202
 - Dijkstra's algorithm, 288
 - Jarník's algorithm, 264
 - widest-path algorithm (Edmonds-Karp), 340, 348
- BFS, *see* breadth-first search
- binary search trees, 67
 - AA trees, 96, 145
 - AVL trees, 96, 144, 341
 - optimal, 91
 - reconfiguration, 68
 - red-black trees, 96, 144
 - left-leaning, *see* AA trees
- binary to decimal conversion, 56
- bipartite graph, 207
- bipartite maximum matching, 355
- bitonic, 60
- black box, 12, 21, 22, 416
 - see also* none of your business
- Bob's mama sees a ukulele, 129
- bond (minimal edge cut), 347
- boolean circuits, 190, 379
- boolean formula, 386
- boolean matrix multiplication, 323
- Borůvka's algorithm, 261
 - advantages, 262
- Boston Pool algorithm, 172
- bottleneck, *see also* minimum cut
- bottleneck distance, 270, 347
- bottleneck spanning tree, 348
- breadth-first search, 202, 278, 341
- Bridges of Königsburg, 190
 - see also* Euler Tour

BST, *see* binary search trees
 Bubba sees a banana, 128
 bus scheduling, 361

 c_f (residual capacity), 331
 Camelot, 185
 Candy Crush Saga, 407
 capacity scaling, 351
 careful graph coloring, 421
 central vertex of a tree, 64
 checkerboard, 59, 222, 370
 checkers, *see* draughts
 children's songs, 16
 choosing the right problem to
 reduce from, 407
 circuit satisfiability, *see* CIRCUITSAT
 CIRCUITSAT, 380
 reduction to 3SAT, 388, 393,
 395
 reduction to SAT, 386
 circulation, 337
 clause, 388
 clause gadget, 390, 395, 396, 401
 clique, 393
 closed walk in a graph, 191
 CNF, *see* conjunctive normal form
 CNF-SAT, *see* SAT
 co-NP, 381
 compass and straightedge, 7
 component, 191
computationes canonica et legalis,
 188
 condensation, *see* strong component
 graph
 configuration graph, 194, 220, 221
 conjunctive normal form, 388
 connected component, *see*
 component
 connected graph, 191
 conservation constraint, 328
 convenience, 32, 79, 227, 237, 279,
 395

Cook reduction, 384
 Cookie Clicker, 407
 counting graph components, 204
 cover gadget, 399
 cross edge (depth-first search), 229
 crossing gadget, 411
cursus publicus, 188
 cut (vertex partition), 329
 cut capacity $\|S, T\|$, 329
 cut vertex, 245
 cycle cover, 370
 cycle flow, 337
 cycle in a graph, 191

 dag, *see* directed acyclic graph
 Dance Dance Revolution, 133
 data structures for graphs, *see*
 graphs
 decision problem, 381
 decision tree, 53
 decision versus optimization, 85
 degree of a vertex, 191
 DeNile, 146
 dependency graph, 107, 109, 114,
 119, 193, 234, 366
 depth-first order, *see* preorder,
 postorder
 depth-first search, 76, 201, **225**, 282,
 380
 DFA, *see* finite-state automaton
 DFS, *see* depth-first search
 Dijkstra's algorithm, 203, 284
 with negative edges, 288
 exponential running time,
 289, 300
 with no negative edges, 286
 Dinic's [Dinitz's] algorithm, 341
 directed acyclic graph, 192, 231
 directed cycle, 192
 directed graph, 190
 DIRECTEDHAMCYCLE, 397
 reduction from 3SAT, 400

- reduction from VERTEXCOVER, 398
- DIRECTEDHAMPATH, 402
 - in a directed acyclic graph, 250
 - in a tournament, 208
 - reduction to shortest simple path, 275
- directed path, 192
- directed walk, 192
- disconnected graph traversal, 203
- disjoint paths
 - edge-disjoint, 353
 - vertex-disjoint, 354
- disjoint-path cover, 253
 - in directed acyclic graphs, 360
 - NP-hard in general graphs, 360
- disjoint-set data structure, 267, 271
- disjunctive normal form, 415
- $\text{dist}_{\leq i}(v)$ (length of shortest walk to v with at most i edges), 291
- $\text{dist}(u, v)$, 309
- $\text{dist}(v)$ (tentative distance), 276
- distance multiplication, *see* min-plus matrix multiplication
- distance tables, 310
- divide and conquer, 26, 29, 31, 37, 91, 315, 323
- domain transformation, 34, 56
- DOMINATINGSET, 407, 422
 - in interval graphs, 178
- dominos, 370, 420
- Don't try to be clever, 80, 82, 87, 89
- DP, *see* Deadpool, dynamic programming
- draughts, 100, 408
 - English ("checkers"), 222
 - international, 408
 - reduction from UNDIRECTEDHAMCYCLE, 409
- drinking songs, 1, 16, 18
- duplation and mediation, 6, 21, 23, 43
- dynamic programming, 97, 101, 178, 183, 185, 222, 282
 - as postorder traversal, 234, 255
 - before Bellman, 101
 - boilerplate, 106
 - in directed acyclic graphs, 235, 250
 - in trees, 120
 - not *always* better than memoization, 117
 - sequential, 105
 - space optimization, 102
 - tree-shaped, 117
- edge (pair of vertices), 187
- edge capacity, 329
- edge contraction, 269
- edge demands, 347
- edge gadget, 398, 402, 410
- edge reweighting, 311
- edge-complement \overline{G} , 393
- edge-disjoint paths, 353
- edit distance, 111, 139, 149, 193, 235, 237
- Edmonds-Karp algorithms
 - fattest augmenting paths, 340
 - shortest augmenting paths, 341
- EDVAC, 27
- Egyptian multiplication, *see* duplation and mediation
- ELEMENTARY, 414
- elves, *see* Recursion Fairy
- empty edge (flows), 329
- endpoints of an edge, 191
- epiphany, 166, 242
- errors, vii
- escape problem, 369

Ethiopian peasant multiplication,
 see duplation and
 mediation
 Euclid's algorithm, 57
 Euler tour, 190, 208, 223, 397
 evaluation order, 107, 109, 114, 119
 as postorder, 235
 single and double arrows, 110,
 120
 EXACT3DIMENSIONALMATCHING,
 405
 EXP (exponential time), 413
 EXP-hard, 413
 exponential decay, 336, 340, 349,
 351
 exponentiation, 43

 $|f|$ (flow value), 328
 factorial, 57
 fake-sugar-packet game, 74
Fantastic Mr. Fox, 133
 fast Fourier transform, 42
 feasible flow, 329
 FFT, *see* fast Fourier transform
 Fibonacci heaps, 264, 288
 Fibonacci numbers, 98, 193, 249
Fight Club, 180
 finished vertex (depth-first search),
 228
 finite-state automaton, 194, 253, 255
 non-deterministic, 195
 PSPACE-hard problems, 413
 First make it work, then make it
 fast, 22, 26, 40, 80, 84, 95,
 96, 99, 106, 108, 109, 123,
 289
 First what, then how, 106
 Fizzbuzz (standard interview
 question), 17
 flood fill, 206
 flow, 328
 flow decomposition, 336, 358
 algorithm, 339
 flow value $|f|$, 328
 flow vector space, 346
 flying kings, 223, 408
 Ford-Fulkerson, 334
 can run forever, 335, 340, 349
 exponential running time, 335
 fattest augmenting paths, 340
 shortest augmenting paths, 341
 Ford's relaxation algorithm, 276
 exponential running time, 300
 forest (= acyclic graph), 191
 formula satisfiability, *see* SAT
 forward edge (depth-first search),
 229
 French flag walk, 303
 French invasion of Indochina, 24
 funny matrix multiplication, *see*
 min-plus matrix
 multiplication

 G_f (residual graph), 332
 gadgets, 395
 Gale-Shapley algorithm, 173
 game state, 75, 194
 game trees, 74, 75, 76, 79, 80, 100
 garbage collection, 210
 gate gadgets, 395
 general patterns
 backtracking, 79
 divide and conquer, 31
 dynamic programming, 105
 graph traversal, 199
 greedy exchange arguments,
 164
 minimum-spanning-tree
 algorithms, 259
 NP-hardness proofs, 392
 shortest-path algorithms, 276
 generic graph traversal, *see*
 whatever-first search
George of the Jungle, 65

- Giggle, 151, 301
- Gilbert and Sullivan
 - HMS Pinafore*, 108
 - The Mikado*, 404
 - The Pirates of Penzance*, 374
- golden ratio, 99, 335, 349
- good pivot, 33, 36
 - median of medians, 37
- graph coloring, 395
 - in interval graphs, 178
- graph embedding, 192
- graph reduction, 205, 212
- graph traversal, 199, 209, 247
 - disconnected graphs, 203, 205, 227
- graphical statics, 189
- graphs
 - data structures
 - adjacency list, 195, 198
 - adjacency matrix, 196, 198
 - comparison, 198
 - implicit, 198, 207, 235
 - historical examples, 187
 - modern examples, 192
 - terminology, 190
- greatest common divisor, 57
- greedy algorithms, **159**
 - are usually wrong, 107
 - that don't work, 172, 177, 183, 185, 223, 349, 372
 - try dynamic programming first, 108
- greedy exchange arguments, 160, 161, 163, **164**, 164, 168, 258, 260
- guillotine subdivision, 145
 - see also* kd-tree
- Gulliver's Travels*, 2, 3, 96, 98, 145
- Hamiltonian cycle, *see*
 - DIRECTEDHAMCYCLE,
 - UNDIRECTEDHAMCYCLE
 - definition, 208, 397
- Hamiltonian path, *see*
 - DIRECTEDHAMPATH,
 - UNDIRECTEDHAMPATH
 - definition, 402
- Handshake Lemma, 218
- hashtags, 80
- head of an edge, 191
- Hellenistic snobbery, 57
- helpful drawings
 - evaluation order arrows, 110, 120
 - NP-hardness reduction, 387
 - recursion trees, 32
- heuristic, 30, 76
 - see also* algorithm that doesn't work
- HITTINGSET, 405
- How do I . . .
 - choose the right problem to reduce from?, 407
 - derive a dynamic programming algorithm?, 106
 - design a backtracking algorithm?, 79
 - prove that a greedy algorithm is correct?, 164
 - prove that a problem is NP-hard?, 386, 392
- Huffman codes, 165, 179
- Huntington-Hill algorithm, 22
- Hyperbole and a Half*, 261, 291
- hypercube, 208
- IBM, 26
- implicit graph representation, 195
- in-degree of a vertex, 191
- incorrect proofs that $P=NP$, 404, 415, 416
- independent set, *see* MAXINDSET, 390
- index formulation, 77, 83, 88

indice, *see* index (dammit)
 induction, ii, 23, 28, 29, 40, 75, 77,
 100, 103, 163–165, 175, 200,
 201, 230, 242, 260, 277, 282,
 284, 287, 291, 318, 334,
 336–338, 341
 backward, *see* dynamic
 programming
 see also recursion
 induction hypothesis, *see* Recursion
 Fairy
 infinite loop, 16, 172, 277, 294, 313,
 335, 349
 input size, 404
 integer maximum flow, 334, 348,
 358
 integer multiplication
 divide-and-conquer, 40
 duplation and mediation, 5
 Karatsuba’s algorithm, 41
 Toom-Cook algorithm, 42
 via fast Fourier transform, 42
 integer multiplication lattice
 algorithm, 3
 Integrality Theorem (maximum
 flows), 334
 international draughts, *see* draughts
 interpuncts (word-spacing), 80
 intersection graph, 192
 interval graph, 192
 interview questions, 124, 143, 156
 inverse Ackerman function $\alpha(n)$,
 267
 inversion counting, 51
 iterated logarithm $\log^* n$, 42

 Jarník’s algorithm, 203, 263
 Jeff actually did this, 249, 349
 Johnson’s algorithm, 312
 jump in the middle, 80, 81, 86, 89

 Kaniel the Dane, 218

 Karp reduction, 384
 kd-tree, 65
 see also guillotine subdivision
 Klondike, 406
 knights and knaves, 59
 Kosaraju-Sharir algorithm, 240
 Kruskal’s algorithm, 265
 kuttaka, *see* Euclid’s algorithm

 label of a path, 251–253
 labeling graph components, 204
 language (set of strings), 384
 largest common subtree, 377
 Latin, 2, 80, 427
 lattice multiplication, 3
 laws of physics, irrelevance of, 336
 Let that which does not matter
 truly slide, 12, 14, 80, 82
 level of a vertex, 341
 Levenshtein distance, *see* edit
 distance
 line breaking, 130
 linear-time selection, 37, 53, 55
 list of NP-hard problems, 404
 literal, 388
 local maximum, 63
 local minimum, 61
 $\log^* n$ (iterated logarithm), 42
 logarithmic-space reduction, 385
 logic gates, 379
 longest common increasing
 subsequence, 126
 longest common subsequence, 94,
 115, 125
 longest increasing digital
 subsequence, 140
 longest increasing subsequence,
 86–90, 109, 237
 LONGESTPATH, 406
 in directed acyclic graphs, 86,
 88, 90, **236**

- reduction from
 - TRAVELINGSALESMAN, 236
- loop invariant, *see* induction
- hypothesis
- $low(v)$, 242
- lower bound via adversary
 - argument, 380
- magnetic tape, 159
- MAJORITY3SAT, 418
- majority gate, 396
- many-one reduction, 384
- marketing buzzwords, 102
- Master Theorem, *see* recursion trees
- matching, 355
 - non-crossing, 376
 - other special cases, 377
- mātrāvṛtta, 97
- matrice, *see* matrix (dammit)
- matrix multiplication
 - boolean, 323
 - in sub-cubic time, 324
 - min-plus, **316**, 323, 325
 - standard, 323, 325
- matrix rounding, 182, 374
- MAX2SAT, 406
- MAXCLIQUE, 393
 - reduction from MAXINDSET, 393
- MAXCUT, 406
- MAXINDSET, 390
 - in circular arc graphs, 185
 - in interval graphs, 162
 - in trees, 120
 - reduction from 3SAT, 390, 393, 395
 - reduction to MAXCLIQUE, 393
 - reduction to MINVERTEXCOVER, 394
- maximum flows, **329**
 - acyclic, 339, 345
 - integer, 334
 - multiple sources and targets, 368
 - with vertex capacities, 354
- maximum independent set, *see* MAXINDSET
- maximum matching in bipartite graphs, 355
- maximum subarray problem, 124
 - two-dimensional, 138
- Maxwell-Cremona diagrams, 189
- mazes, 190, 247
 - acute-angle, 216, 304
 - number, 212
- median, *see* selection
- median-of-medians selection, 37, 53
- median-of-medians-of-medians selection, 55
- median-of-three heuristic, 30, 50
- memoization, 99, 194, 234, 271, 278
 - see also* dynamic programming
- memoized recursion is depth-first search, 234
- mergesort, 26
- mergesort recurrence, 28, 30, 33
- metagraph, *see* strong component graph
- methodisches Tatonnieren*, 72
- MINVERTEXCOVER, 393
 - reduction from MAXINDSET, 394
 - reduction to
 - DIRECTEDHAMCYCLE, 398
 - reduction to SUBSETSUM, 402
- min-plus matrix multiplication, **316**, 323, 325
- Minesweeper, 406
- minimum clique cover
 - in circular arc graphs, 186
 - in interval graphs, 178
- minimum cuts, 330
- minimum spanning trees, 203, **257**
 - uniqueness, 258, 268

Minty's algorithm, *see* Dijkstra's algorithm
 mom, *see* median of medians
 mondegreen, 18
 Monopoly, actual rules of, 385
 Moore's algorithm, 292
 Morse code, 97, 165
 mountain climbing problem, 217
 MST, *see* minimum spanning trees
 multigraph, 191

n queens, 71, 79, 80
 Nadirian Dream-Dollars, 123, 181
Napoleon Dynamite, 131
 National Resident Matching Program, 171
 "Needleman-Wunch" algorithm, 115
 negative cycle detection, 292, 294, 299
 negative cycles, 275, 277
 negative edges, 274
 neighbor of a vertex, 191
 Neitherlands (*The Magicians*), 375
 nesting boxes, 372
 new vertex (depth-first search), 228
 NFA, *see* finite-state automaton
 Nobel Prize in Algorithms
 Economics, 173
 node, *see* vertex
 none of your business, 21, 23, 25, 80, 206
 see also black box
 NOTALLEQUAL3SAT, 405
 NP (nondeterministic polynomial time), 381
 NP versus co-NP, 382
 NP versus EXP, 413
 NP versus PSPACE, 412
 NP-complete, 383
 NP-hard, 121, 138, 140, 235, 275, 357, 360, 381, **382**
 formal definition, 384
 weakly, 117, 404
 obvious, 14, 76, 80, 317, 382, 383
 Oh yeah, we already did this, 30, 78
 one-armed quicksort, *see* quickselect
 open problems
 all-pairs shortest paths, 317
 integer multiplication, 42
 matrix multiplication, 317
 optimal addition chains, 44
 optimal pancake flipping, 49
 P versus NP, 381
 sorting binary trees by swaps and rotations, 69
 winning international draughts in one turn, 411
 open-pit mining
 see project selection, 365
 optimal binary trees
 binary search trees, 91, 117
 variants, 96, 144
 expression trees, 141, 142
 prefix-free binary codes, 165
 optimal substructure, *see also* correct recurrence, 112
 ordered subtree, 156
 Orlin's algorithm, 344
 out-degree of a vertex, 191

 P (polynomial time), 381
 P versus NP, 382
 P versus PSPACE, 412
 $P \neq NP$ as a law of nature, 382
 Pac-Man, 407
 palindrome, 81, 128, 218, 222, 252, 377
 pancake sorting, 49, 151
 parallel assignment, 249
 parent, 200, 276
 see also mom
 PARTITION

- NP-hard problem, 405
- subroutine in quicksort and
 - quickselect, 29
- party planning, 182
- path compression, 271
- path flow, 337
- path in a graph, 191
- peasant multiplication, *see*
 - duplation and mediation
- pebbling, 421
- pecking order, 208
- pixels, 206
- PLANAR3SAT, 405
- PLANARCIRCUITSAT, 404
- planar graph, 192
- PLANARNOTALLEQUAL3SAT, 405
- Plankalkül*, 205, 278
- plumbus, 322
- politics
 - academic, 59, 373, 375, 383
 - Illinois, 373
 - Renaissance Italian, 47, 59
 - Soviet, 383
- postorder, 227
 - tree traversal, 65, 66, 122
- power, *see* exponentiation
- $\text{pred}(u, v)$, 309
- $\text{pred}(v)$ (tentative predecessor), 276
- prefix, 77
- prefix-free binary code, 165
- preorder, 227
 - tree traversal, 65, 66
- prerequisites, i
- references, ii
- Prim's algorithm, *see* Jarník's
 - algorithm
- project selection, 365
- proper k -coloring, 395
- proper subgraph, 191
- Propositiones ad Acuendos Juvenes*,
 - 427
- prosody, 97
 - see also* Fibonacci numbers
 - see also* Morse code
- pseudo-polynomial time, 117, 404
- PSPACE (polynomial space), 411
- PSPACE versus EXP, 413
- PSPACE-hard, 412
- punched cards, 26
- QBF (quantified boolean formula),
 - 412
- quickselect, 35
- quicksort, 29
- quicksort recurrence, 33, 34
- Racetrack, 220
- rainbow, 387
- RAND Corporation, 101
- random-access machine, 384
- $\text{reach}(v)$, 226
- $\text{reach}^{-1}(v)$, 238
- reachability, 191, 199
 - directed, 226
- reciprocal diagrams, 189
- recommended course policies, 108,
 - 123
- reconfiguration problems, 24, 45,
 - 49, 68, 213, 215, 216, 219,
 - 221, 427
- recurrences
 - full history, 85, 92
 - removing floors and ceilings,
 - 34
 - scary, 84, 92
 - solving with recursion trees, 31
- recursion, 22, 261, 291, 353
 - backtracking, 71
 - depth-first search, 225
 - divide and conquer, 26
 - smart, *see* dynamic
 - programming
 - see also* induction
- Recursion Fairy, 22, 25, 26, 28, 37,
 - 337

recursion trees, 31, 49, 99
 all levels equal, 28, 30, **33**, 33,
 39, 49
 backtracking, 72, 76
 exponential decay, **33**, 37–39,
 43, 49
 exponential growth, **33**, 40, 49,
 78, 85
 path, 36, 37
 weird, 30, 49
 recursive brute force, *see*
 backtracking
 reduced flow network, 331, 345
 reduction, 21, 56, 205, 212, 385
 regular expressions, 151
 generalized, 414
 PSPACE-hard problems, 413
 relaxing a tense edge, 276
 repeated squaring, 43, 44, 103, 317
 replacement paths, 298
 repricing, *see* vertex reweighting
 residual capacity c_f , 331
 residual graph G_f , 332
 results by RAND researchers, 101,
 276, 284, 331
 results by students, 35, 41, 42, 166,
 278, 324, 341, 383, 406
 Revelation 13:15–18, 161
 reversal $rev(G)$ of directed graph G ,
 234
 reverse topological order, *see*
 postorder
Rick and Morty, 322
 road maps, 187, 274, 310
 rock climbing, 153, 184, 253
 rooted subtree, 121
 Rubik's Cube, 407
 rule of three, 392, 393, 395, 405,
 407, 408, 419
 ruler function, 45, 46
 Russian peasant multiplication, *see*
 duplation and mediation
 $\|S, T\|$ (cut capacity), 329
 (s, t) -cut, 329
 (s, t) -flow, 328
 safe edge, 259
 SAT, 386
 reduction from CIRCUITSAT,
 386
 satisfiability
 circuit, *see* CIRCUITSAT
 formula, *see* SAT
 saturated edge (flows), 329
 $scc(G)$ (strong component graph),
 237
 scheduling, 161
 greedy algorithms that don't
 work, 177
 via dynamic programming, 162
 via greedy algorithm, 162, 362
 via maximum flows, 358, 361
 Scrabble, 134
scriptio continua, 80
 Seidel's algorithm, 317, 324
 selection, 35, 52
 median-of-medians, 37, 53
 median-of-medians-of-
 medians,
 55
 quickselect, 35
 self-descriptive sentence, 166
 self-reduction, 416
 semi-connected graph, 244
 sequence alignment, *see* edit
 distance
 sequence of decisions, 79, 112
 series-parallel graph, 350
 SETCOVER, 405
 Sham-Poobanana University, 142,
 161, 358, 361, 372, 373
 Shimbél's algorithm, *see*
 Bellman-Ford
 shortest common supersequence,
 95, 125

- shortest path tree, 274
- shortest paths, 203
 - all-pairs, **309**
 - analog algorithms, 278
 - in directed acyclic graphs, 282
 - in unweighted graphs, 278
 - single-source, **273**
 - versus shortest walks, 275
 - with negative edges, 274, 288
 - in undirected graphs, 275
 - with non-negative edges, 284
- shortest simple path
 - reduction from
 - DIRECTEDHAMPATH, 275
- shuffle, 126
- simple graph, 191
- sink (vertex with out-degree 0), 231
- sink component, 239, 242
- Snakes and Ladders, 212
- soapbox, v, 12, 106
- Sollin's algorithm, *see* Borůvka's algorithm
- solving a more general problem, 36, 80
- solving the right problem, 83, 87, 88, 90, 91
- sorting algorithms
 - mergesort, 26
 - quicksort, 29
- source (in a flow network), 328
- source (vertex with in-degree 0), 231
- source component, 240
- spanning forest, 192
- spanning tree, 192
- squaring and mediation, 44
- SSSP, *see* shortest paths, single-source
- stable matching, 170, 179
- starting time of a vertex (depth-first search), 228, 242
- STEINERTREE, 406
- Stigler's Law of Eponymy, 42, **98**, 114, 189, 261, 263, 284, 289, 383
- Strassen's algorithm, 317
- strong component graph, 237
- strong components, 237
 - connected in depth-first forest, 239
 - in linear time, 238
 - Kosaraju-Sharir, 240
 - Tarjan's algorithm, 242
- strong connectivity, 237
- strongly connected components, *see* strong components
- strongly connected graph, 192
- subgraph, 191
- subsequence, 86
- subset construction, 195
- SUBSETSUM, 76, 79, 80, 93, 116, 237, 402
 - dynamic programming algorithm, 404
 - in pseudo-polynomial time, 117
 - reduction from VERTEXCOVER, 402
- Sudoku, 406
- suffix, 83
- Sumerian clay tablets, 56
- summary of past decisions, 79
- Super Mario Brothers, 407
- Tabula Peutingeriana*, 187
- tail of an edge, 191
- talking dog joke, 318, 383
- tape sorting, 159
- target (in a flow network), 328
- Tarjan's algorithm, 242
- tâtonner*, 72
- tense edge, 276, 312
- Tetris, 406
- text segmentation, **80**, 86, 94, 105, 124, 130, 237

Theseus (maze-solving robot), 100, 278
 Threes (game), 407
 Tibetan Memory Trick, *see* The Announcer's Test
 token (breadth-first search), 279
 token (Moore's algorithm), 293
 topological order, *see* reverse postorder
 topological sort, 232
 implicit, 232
 Tower of Hanoi, 24, 44
 configuration graph, 194
 non-recursive solutions, 44
 recurrence, 26, 78, 85, 89
 variants, 46–48, 140
 Trainyard, 407
 transforming certificates, 392
 transitive closure, 246, 323
 transitive reduction, 246
 TRAVELINGSALESMAN, 402
 dynamic programming, 139
 Euclidean, convex position, 139
 reduction from
 DIRECTEDHAMCYCLE, 402
 reduction to LONGESTPATH, 236
 tree (connected acyclic graph), 191
 equivalent definitions, 207
 tree edge (depth-first search), 229
 tree traversal, 66, 227
 postorder, 122
 trivial but useless $O(1)$ -time algorithms, 17, 409
 truth gadget, 395
 TSP, *see* TRAVELINGSALESMAN
 tuple selection, 357
 Turing machines, 384
 Turing reduction, 384
 Twitbook, 152, 301
 typography, 80, 130
 Ulam distance, *see* edit distance
 Ultimate Paperclips, 407
 undecided edge, 260
 undirected graph, 190
 UNDIRECTEDHAMCYCLE, 402
 in a hypercube, 208
 knight's tour, 190
 reduction to international draughts, 409
 UNDIRECTEDHAMPATH, 402
 union-find, *see* disjoint-set data structure
 unordered subtree, 156
 UNSAT, 385
 useful deliberate ignorance, 22, 26, 35
 useless edge, 259
 vacuous base case, 25, 76, 87, 92
 value of a node in a recursion tree, 31
 Vankin's Kilometer, 138
 Vankin's Mile, 137
 variable gadget, 390, 395, 400
 vertex, 187
 vertex cover, 393
 vertex gadget, 398, 402, 409
 vertex-disjoint paths, 354
 vertice, *see* vertex (dammit)
 Vidrach Itky Leda, 213
 walk in a graph, 191
 wavefront, 281, 284, 286
 weakly NP-hard, 117, 404
 weighted median, 53
 WFS, *see* whatever-first search
 Whackbat, 133
 whatever-first search, 199
 best-first (priority queue), *see* also best-first search, 202
 breadth-first (queue), 202
 depth-first (stack), 201
 widest paths, 203, 270

- word RAM model, 384
- X₃M, 405
- XCNF-SAT, 418
- xkcd, 407
- zero cycles, 321

Dicebat Bernardus Carnotensis nos esse quasi nanos gigantium humeris insidentes, ut possimus plura eis et remotiora videre, non utique proprii visus acumine, aut eminentia corporis, sed quia in altum subvehimur et extollimur magnitudine gigantea.

[Bernard of Chartres used to say that we were like dwarfs seated on the shoulders of giants. He pointed out that we see more and farther than our predecessors, not because we have keener vision or greater height, but because we are lifted up and borne aloft on their gigantic stature.]

— John of Salisbury, *Metalogicon* (1159),
translated by Daniel D. McGarry (1955)

The secret to productivity is getting dead people to do your work for you.

— Robert J. Lang (2009)

Index of People

Adelson-Velsky, Georgy, 96, 144, 341
Adler, Ilan, 363
Adversary, All-Powerful Malicious,
 31, 161, 376, 379
Alcuin of York, 428
Alice, 217
Alighieri, Dante, 2
Alon, Noga, 317
Andersson, Arne, 96, 145
Apollonius of Perga, 3
Approximate Median Fairy, 33, 37
Archimedes, 3
Atlas, Charles, 102
St. Augustine of Hippo, 81

Bayer, Rudolf, 96, 144
Bellman, Richard, 101, 289
Berge, Claude, 356
Blagojevich, Rod, 372
Blum, Manuel, 35
Bob, 217
Borůvka, Otakar, 261
Brahmagupta, 3
Brosh, Allie, 261, 291

Cayley, Arthur, 190
Cegłowski, Maciej, 182
Chaucer, Geoffrey, 2
Chazelle, Bernard, 217

- Chazelle, Damien, 217
 Choquet, Gustav, 261
 Chowdhury, Rezaul, 324
 Cicero, Marcus Tullius, 80
 Claus, N. (de Siam), *see* Lucas, Édouard
 Clifford, William, 190
 Cobham, Alan, 381
 Cook, Stephen, 42, 383
 Couper, Archibald, 190
 Cremona, Luigi, 189
 Culmann, Carl, 189

 Dantzig, George, 276, 284, 289, 328
 Demaine, Erik, 413
 Dijkstra, Edsger, 210, 263, 269, 284
 DiMaggio, Joe, 318
 Dinitz, Yefim, 341
 Dweighter, Harry (pseudonym of Jacob Goodman), 49

 Edmonds, Jack, 311, 334, 340, 381
 Elias, Peter, 331
 “Engine Charlie”, *see* Wilson, Charles Erwin
 Eppstein, David, 209
 Erera, Alan, 363
 Erickson, Hannah, 216, 252
 Erickson, Kate, 387
 Euclid, 7, 57
 Euler, Leonhard, 190, 247
 Eutocius of Ascalon, 3

 Fahlberg, Constantin, 74
 Fano, Robert, 166
 Feinstein, Amiel, 331
 Fernández-Baca, David, 364
 Fibonacci, *see* Leonardo of Pisa
 Fischer, Michael, 114, 315
 Floyd, Robert, 35, 318
 Fontana, Giovanni, 190
 Ford, Lester, 276, 331

 Frederick II, Holy Roman Emperor, 47
 Fredman, Micheal, 264
 Frisius, Gemma, 190
 Fulkerson, Delbert, 311, 331

 Gödel, Kurt, 381
 Gale, David, 172
 Galil, Zvi, 317
 Garey, Michael, 404
 Gates, Bill, 49
 Gauß, Karl Friedrich, 42, 72
 Goldstine, Herman, 26
 Goodrich, Michael, 209
 Gregory IX, Pope, 47
 Grimm, Jacob and Wilhelm, 23
 Guibas, Leonidas, 96, 144
 Gusfield, Dan, 364

 Harris, Theodore, 327
 Harvey, David, 42
 Hearn, Robert, 409, 413
 Herotodus, 2
 Hierholzer, Carl, 190, 247
 Hillier, John, 284
 Hoare, Tony, 29, 35
 Hochbaum, Dorit, 363
 van der Hoeven, Joris, 42
 Hopcroft, John, 356
 Huffman, David, 166

 Ingerman, Peter, 318

 Jacobi, Carl, 356
 Jarník, Vojtěch, 263
 Jay, Ricky, 392
 Johnson, David, 404
 Johnson, Donald, 288, 311

 Kőnig, Dénes, 356
 Kalaba, Robert, 291
 Kane, Daniel, 218
 Karatsuba, Anatolii, 41

Karp, Richard, 311, 334, 340, 356,
404

Karzanov, Alexander, 344

Kekulé, August, 190

al-Khwārizmī, Muḥammad ibn
Mūsā, 2

Kirchhoff, Gustav, 190

Kleene, Stephen, 318

Kolmogorov, Andrei, 41

Kosaraju, Rao, 240

Kruskal, Joseph, 263

Kuhn, Harald, 356

Lamport, Leslie, 210

Landis Evgenii, 96, 144

Laquière, Emmanuel, 72

Ledger, Heath, 380

Lee, Chin Yang, 278

Leonardo of Pisa, 2, 4, 47, 97, 98,
101

Levin, Leonid, 383

Leyzorek, Michael, 284, 316

Loberman, Harry, 263, 266

Lucas, Édouard, 24, 72, 103

Łukaszewicz, Józef, 261

Maḍry, Aleksander, 344

Margalit, Oded, 317

Marston, John, 18

Martel, Charles, 364

Martin, Alain J., 210

Martin, Steve, 10

Massé, Pierre, 101

Maxwell, James Clerk, 189

McKenna, Terence, 23

Meyer, Albert, 315

Michie, Donald, 100, 117

Miller, Gary, 50

Minty, George, 276, 284, 289

Mom, 38

Moore, Edward, 206, 273, 278, 289

Moreno, Jacob, 190

Morgenstern, Oskar, 101

Murena, Lucius Licinius, 80

Musk, Elon, 182

Nash, John, 381

Nauck, Franz, 72

Okasaki, Chris, 13

Olinick, Eli, 363

Orlin, James, 343

Pacioli, Luca, 45

Papadimitriou, Christos, 49

Pappus of Alexandria, 3

Park, Joon-Sang, 324

Peirce, Charles Sanders, 190

Penner, Michael, 324

Peranson, Elliott, 172

Piñgala, 43, 97, 103

Pinker, Steven, 13

Prasanna, Viktor, 324

Pratt, Vaughan, 35

Prim, Robert, 260, 263

Queyranne, Maurice, 349

Rabin, Michael, 381

Ramachandran, Vijaya, 324

Rebaudi, Ovidio, 74

Recursion Fairy, ii, 22, 77, 80, 82, 89,
164, 169

Remsen, Ira, 74

Rivest, Ronald, 35

Ross, Frank, 327

Roy, Bernard, 318

Sainte-Laguë, André, 190

Sallows, Lee, 166

Samuel, Arthur, 100

Saxel, Jindřich, 261

Scholten, Carel S., 210

Schrijver, Lex, 327

Schumacher, Heinrich, 72

Schwartz, Benjamin, 364

Sedgewick, Robert, 96, 144, 145

- Shannon, Claude, 100, 166, 190, 278, 331
- Shapley, Lloyd, 172
- Sharir, Micha, 240
- Shier, Douglas, 289
- Shimbel, Alfonso, 289, 314
- Siedel, Raimund, 317
- Skiena, Steve, vii
- Smullyan, Raymond, 59
- Snell, Willebrod, 190
- Sollin, George, 261
- Steele, Guy, 16
- Steffens, Elisabeth, 210
- Stevin, Simon, 189
- Stigler, Stephen, 98
- Strassen, Volker, 317
- Sulpicius Rufus, Servius, 81
- Sylvester, James, 190
- Tarjan, Robert, 35, 242, 264
- Tarry, Gaston, 247
- Tomizawa, Nobuaki, 311
- Toom, Andrei, 42
- Trémaux, Charles, 247
- Tseitin, Grigorii, 388
- Turing, Alan, 101
- Varignon, Pierre, 189
- Virahāṅka, 98, 101
- von Neumann, John, 26, 101, 381
- von Staudt, Karl, 190
- Wagner, Robert, 114
- Waits, Tom, 379
- Warshall, Stephen, 318
- Wayne, Kevin, 365
- Weinberger, Arnold, 263, 266
- Weiss, Mark Allen, 96, 145
- Whiting, Peter, 284
- Whittlesey, Kim, 273
- Wiener, Christian, 247
- Wilson, Charles Erwin, 102
- Witzgall, Christoph, 289
- Woodbury, Max, 289
- Yuval, Gideon, 325
- Zermelo, Ernst, 75
- Zuse, Konrad, 205, 278
- Zwick, Uri, 335

We should explain, before proceeding, that it is not our object to consider this program with reference to the actual arrangement of the data on the Variables of the engine, but simply as an abstract question of the nature and number of the operations required to be performed during its complete solution.

— Ada Augusta Byron King, Countess of Lovelace,
translator's notes for Luigi F. Menabrea,
"Sketch of the Analytical Engine invented by Charles Babbage, Esq." (1843)

*How to play the flute. [picks up a flute] Well, here we are.
You blow there and you move your fingers up and down here.*

— Alan [John Cleese], "How to Do It",
Monty Python's *Flying Circus*, episode 28 (aired October 26, 1972)

Index of Pseudocode

This index includes only algorithms with explicit pseudocode; see the main index for other named algorithms.

ADDSAFEEDGES, 262

ADDSAFEEDGES, 272

ALLPAIRSBELLMANFORD, 314

ALOUETTE, 16

APPORTIONCONGRESS, 9

BEAMILLIONAIREANDNEVERPAYTAXES,
10

BELLMANFORD, 291, 292

BELLMANFORDDP, 295

BELLMANFORDDP2, 296

BELLMANFORDDP3, 296

BELLMANFORDFINAL, 296

BFS, 279

BFSWITHTOKEN, 279

BINARYGCD, 58

BORŮVKA, 262, 272

BOTTLESOFBEER, 1

CIRCUITSAT, 387

COLLECTSTEP, 211

COMPUTEOPTCOST, 119

CONSTRUCTSUBSET, 79

COUNTANDLABEL, 204

COUNTCOMPONENTS, 204

CRUEL, 51

DAGSSSP, 283
 DFS, 225, 226, 228, 235
 DFSALL, 227, 228
 DIJKSTRA, 285
 DYNAMICPROGRAMMING, 235

 EAGERWFS, 209
 EUCLIDGCD, 58

 FACTORIAL, 57
 FALLING, 57
 FASTEUCLIDGCD, 58
 FASTLIS, 110
 FASTLIS2, 111
 FASTMULTIPLY, 41
 FASTRECFIBO, 104
 FASTSPLITTABLE, 106
 FASTSUBSETSUM, 117
 FELLMANBORED, 299
 FETCHBIT, 63
 FIBONACCI MULTIPLY, 4
 FINDLOW, 243
 FINDLOWDFS, 243
 FINDSAFEEDGES, 272
 FISCHERMEYERAPSP, 315
 FLOYDWARSHALL, 319
 FORDSSSP, 277

 GARBAGECOLLECT, 211
 GREEDYFLOW, 349
 GREEDYSCHEDULE, 163

 HANOI, 26
 HHGUESS, 19

 INITF, 118
 INITSSSP, 276
 ISACYCLIC, 231
 ISACYCLICDFS, 231
 ITERATIVEDFS, 199
 ITERFIBO, 101
 ITERFIBO2, 103

 JARNÍK, 265
 JARNÍKINIT, 265
 JARNÍKLOOP, 265
 JOHNSONAPSP, 313

 KLEENEAPSP, 319
 KOSARAJUSHARIR, 241
 KRUSKAL, 267

 LABELONE, 204
 LEYZOREKAPSP, 316
 LIS, 89, 90
 LISBIGGER, 88
 LISFIRST, 90
 LONGESTPATH, 236, 237

 MARKEVERYVERTEXDUH, 203
 MEMFIBO, 100
 MEMOIZE, 235
 MERGE, 27
 MERGESORT, 27
 MOMOMSELECT, 55
 MOMSELECT, 37
 MOM_bSELECT, 53
 MOORE, 293
 MULTIPLYORDIVIDE, 8
 MUTATE, 212

 NONNEGATIVEDIJKSTRA, 288

 OBVIOUSAPSP, 310
 OPTIMALBST, 119
 OPTIMALBST2, 120
 OPTIMALBST3, 120

 PARTITION, 29
 PEASANTMULTIPLY, 6, 23
 PEASANTPOWER, 44
 PIÑGALAPOWER, 43
 PLACEQUEENS, 73
 PLAYANYGAME, 76
 POSTPROCESS, 233
 POSTPROCESSDAG, 233
 POSTPROCESSDAGDFS, 233
 POSTPROCESSDFS, 233
 POSTVISIT, 227

PREPROCESS, 227
PREVISIT, 227
PUSHDAGSSSP, 284

QUEYRANNEFATPATHS, 350
QUICKSELECT, 36
QUICKSORT, 29

RECFIBO, 99
RECTARRY, 247
RECTARRY2, 248
RECURSIVEDFS, 199
RELAX, 277
RIGHTANGLE, 8
RULERHANOI, 45

SHIMBELAPSP, 314
SHORTEREDGE, 259
SLOWPOWER, 43
SPLITMULTIPLY, 40
SPLITTABLE, 83
SQRTSORT, 52
STOOGESORT, 50

STRONGCOMPONENTS, 239
SUBSETSUM, 77, 78

TARJAN, 244
TARJANDFS, 244
TARRY, 247
TARRY2, 248
THREECOLORQUEUESEARCH, 211
THREECOLORQUEUESTEP, 211
THREECOLORSEARCH, 210
THREECOLORSTACKSEARCH, 210
THREECOLORSTACKSTEP, 210
THREECOLORSTEP, 210
TOPOLOGICALSORT, 233, 234
TOPSORTDFS, 233
TREEMIS, 122

UNUSUAL, 51

WFSALL, 203
WHATEVERFIRSTSEARCH, 200, 205
WHOTARGETSWHOM, 62

A wisely chosen illustration is almost essential to fasten the truth upon the ordinary mind, and no teacher can afford to neglect this part of his preparation.

— Howard Crosby (c.1880)

One showing is worth a hundred sayings.

— Alan Watts (misquoting a Chinese proverb), *The Way of Zen* (1957)

Please do not think that this is a neutral matter and that the only advantage of doing without pictures is that of saving space. Pictures in textbooks actually interfere with the learning process.

— Neville Martin Gwynne, *Gwynne's Grammar* (2013)

Image Credits

All figures in this book are original works of the author, except for the following. All listed works are in the public domain unless otherwise indicated.

- Figure 0.1 (page 5) — Biblioteca nazionale Braidense (Milano)
<http://atena.beic.it/webclient/DeliveryManager?pid=2953344>
- Figure 0.2 (page 5) — Internet Archive
<https://archive.org/details/archimedisopera05eutogoo/page/n377>
- Figure 1.16 (page 46) — Internet Archive
<https://archive.org/details/p1rcrationsmoolucaoft/page/162>
- Figure 1.25 (page 61) — Derived from a crayon portrait of the author by Tina Erickson (2000); included with permission of the artist.
- Figure 5.1 (page 188) — Wikimedia Commons
https://commons.wikimedia.org/wiki/File:Tabula_Peutingeriana_-_Miller.jpg

- Figure 5.2 (page 189) — Gallery of “Legal Trees” published by the Yale Law Library under a Creative Commons Licence
<https://www.flickr.com/photos/yalelawlibrary/albums/72157621954683764>
- Figure 5.3 (page 189) — Internet Archive
<https://archive.org/details/Ao77240124/page/n261>
- Exercises 5.20 (page 216) and 8.22 (page 304) — Original puzzles by the author, inspired by Jason Batterson and Shannon Rogers, *Beast Academy Math: Practice 3A*, 2012.
<https://beastacademy.com/pdf/3A/printables/AngleMazes.pdf>
<https://www.beastacademy.com/resources/printables.php>
- Figure 10.1 (page 328) — T[homas] E. Harris and F[rank] S. Ross. Fundamentals of a method for evaluating rail net capacities. The RAND Corporation, Research Memorandum RM-1517, October 24, 1955. United States Government work in the public domain.
<http://www.dtic.mil/dtic/tr/fulltext/u2/o93458.pdf>

1. *Have something to say.*
2. *Say it.*
3. *Stop when you have said it.*
4. *Give the paper a proper title.*

— John Shaw Billings, “An Address on Our Medical Literature”,
International Medical Congress, London (1881)

You know, I could write a book.

And this book would be thick enough to stun an ox.

— Laurie Anderson, “Let X=X”, *Big Science* (1982)

Colophon

This book was edited in TeXShop and typeset using pdf \LaTeX (MacTeX-2018) using the memoir document class (with madsen chapter style, komalike head style, and Ruled page style); several standard packages including amsmath, babel, enumitem, imakeidx, mathdesign, microtype, and standalone; and an embarrassing amount of customization and \TeX -hacking. The text is typeset in Bitstream Charter, Ἀγρεμυσία, Roboto, and Inconsolata. Except as indicated in the Image Credits, all figures were drawn by the author using OmniGraffle Pro and included as PDF files, using the graphicx \LaTeX package.

Portions of our programming have been mechanically reproduced, and we now conclude our broadcast day.

