Steven S. Skiena Miguel A. Revilla

PROGRAMMING CHALLENGES

The Programming Contest Training Manual

With 65 Illustrations

Springer

Contents

1 Getting Started 1

1.1 Getting Started With the Judge . . . 1

1.1.1 The Programming Challenges Robot Judge . . . 2

1.1.2 The Universidad de Valladolid Robot Judge . . . 2

1.1.3 Feedback From the Judge . . . 3

1.2 Choosing Your Weapon . . . 4

1.2.1 Programming Languages . . . 5

1.2.2 Reading Our Programs . . . 6

1.2.3 Standard Input/Output . . . 7

1.3 Programming Hints . . . 9

1.4 Elementary Data Types . . . 11

1.5 About the Problems . . . 13

1.6 Problems . . . 15

1.6.1 The $3n + 1$ Problem . . . 15

1.6.2 Minesweeper . . . 16

1.6.3 The Trip . . . 17

1.6.4 LCD Display . . . 18

1.6.5 Graphical Editor . . . 19

1.6.6 Interpreter . . . 21

1.6.7 Check the Check . . . 23

1.6.8 Australian Voting . . . 25

1.7 Hints . . . 26

1.8 Notes . . . 26

2 Data Structures **27**
2.1 Elementary Data Structures 27
2.1.1 Stacks 28
2.1.2 Queues 28
2.1.3 Dictionaries 30
2.1.4 Priority Queues 31
2.1.5 Sets 32
2.2 Object Libraries 33
2.2.1 The C++ Standard Template Library 33
2.2.2 The Java `java.util` Package 33
2.3 Program Design Example: Going to War 34
2.4 Hitting the Deck 35
2.5 String Input/Output 37
2.6 Winning the War 38
2.7 Testing and Debugging 39
2.8 Problems 42
2.8.1 Jolly Jumpers 42
2.8.2 Poker Hands 43
2.8.3 Hartals 45
2.8.4 Crypt Kicker 47
2.8.5 Stack 'em Up 48
2.8.6 Erdös Numbers 50
2.8.7 Contest Scoreboard 52
2.8.8 Yahtzee 53
2.9 Hints 55
2.10 Notes 55

3 Strings **56**
3.1 Character Codes 56
3.2 Representing Strings 58
3.3 Program Design Example: Corporate Renamings 59
3.4 Searching for Patterns 61
3.5 Manipulating Strings 62
3.6 Completing the Merger 63
3.7 String Library Functions 64
3.8 Problems 66
3.8.1 WERTYU 66
3.8.2 Where's Waldorf? 67
3.8.3 Common Permutation 69
3.8.4 Crypt Kicker II 70
3.8.5 Automated Judge Script 71
3.8.6 File Fragmentation 73
3.8.7 Doublets 74
3.8.8 Fmt 75
3.9 Hints 77

3.10 Notes . . . 77

4 Sorting **78**
4.1 Sorting Applications . . . 78
4.2 Sorting Algorithms . . . 79
4.3 Program Design Example: Rating the Field . . . 82
4.4 Sorting Library Functions . . . 83
4.5 Rating the Field . . . 85
4.6 Problems . . . 88
4.6.1 Vito's Family . . . 88
4.6.2 Stacks of Flapjacks . . . 89
4.6.3 Bridge . . . 91
4.6.4 Longest Nap . . . 92
4.6.5 Shoemaker's Problem . . . 94
4.6.6 CDVII . . . 95
4.6.7 ShellSort . . . 97
4.6.8 Football (aka Soccer) . . . 99
4.7 Hints . . . 101
4.8 Notes . . . 101

5 Arithmetic and Algebra **102**
5.1 Machine Arithmetic . . . 102
5.1.1 Integer Libraries . . . 103
5.2 High-Precision Integers . . . 103
5.3 High-Precision Arithmetic . . . 105
5.4 Numerical Bases and Conversion . . . 110
5.5 Real Numbers . . . 112
5.5.1 Dealing With Real Numbers . . . 113
5.5.2 Fractions . . . 113
5.5.3 Decimals . . . 114
5.6 Algebra . . . 115
5.6.1 Manipulating Polynomials . . . 115
5.6.2 Root Finding . . . 116
5.7 Logarithms . . . 116
5.8 Real Mathematical Libraries . . . 117
5.9 Problems . . . 119
5.9.1 Primary Arithmetic . . . 119
5.9.2 Reverse and Add . . . 120
5.9.3 The Archeologist's Dilemma . . . 121
5.9.4 Ones . . . 122
5.9.5 A Multiplication Game . . . 123
5.9.6 Polynomial Coefficients . . . 124
5.9.7 The Stern-Brocot Number System . . . 125
5.9.8 Pairsumonious Numbers . . . 127
5.10 Hints . . . 128

5.11 Notes . . . 128

6 Combinatorics **129**
6.1 Basic Counting Techniques . . . 129
6.2 Recurrence Relations . . . 131
6.3 Binomial Coefficients . . . 131
6.4 Other Counting Sequences . . . 133
6.5 Recursion and Induction . . . 135
6.6 Problems . . . 137
6.6.1 How Many Fibs? . . . 137
6.6.2 How Many Pieces of Land? . . . 138
6.6.3 Counting . . . 139
6.6.4 Expressions . . . 140
6.6.5 Complete Tree Labeling . . . 141
6.6.6 The Priest Mathematician . . . 142
6.6.7 Self-describing Sequence . . . 144
6.6.8 Steps . . . 145
6.7 Hints . . . 146
6.8 Notes . . . 146

7 Number Theory **147**
7.1 Prime Numbers . . . 147
7.1.1 Finding Primes . . . 148
7.1.2 Counting Primes . . . 149
7.2 Divisibility . . . 149
7.2.1 Greatest Common Divisor . . . 150
7.2.2 Least Common Multiple . . . 151
7.3 Modular Arithmetic . . . 152
7.4 Congruences . . . 154
7.4.1 Operations on Congruences . . . 154
7.4.2 Solving Linear Congruences . . . 155
7.4.3 Diophantine Equations . . . 155
7.5 Number Theoretic Libraries . . . 156
7.6 Problems . . . 157
7.6.1 Light, More Light . . . 157
7.6.2 Carmichael Numbers . . . 158
7.6.3 Euclid Problem . . . 159
7.6.4 Factovisors . . . 160
7.6.5 Summation of Four Primes . . . 161
7.6.6 Smith Numbers . . . 162
7.6.7 Marbles . . . 163
7.6.8 Repackaging . . . 164
7.7 Hints . . . 166
7.8 Notes . . . 166

8 Backtracking **167**
8.1 Backtracking . . . 167
8.2 Constructing All Subsets . . . 169
8.3 Constructing All Permutations . . . 170
8.4 Program Design Example: The Eight-Queens Problem . . . 172
8.5 Pruning Search . . . 173
8.6 Problems . . . 176
8.6.1 Little Bishops . . . 176
8.6.2 15-Puzzle Problem . . . 177
8.6.3 Queue . . . 179
8.6.4 Servicing Stations . . . 180
8.6.5 Tug of War . . . 181
8.6.6 Garden of Eden . . . 182
8.6.7 Color Hash . . . 184
8.6.8 Bigger Square Please... . . . 186
8.7 Hints . . . 188
8.8 Notes . . . 188

9 Graph Traversal **189**
9.1 Flavors of Graphs . . . 189
9.2 Data Structures for Graphs . . . 191
9.3 Graph Traversal: Breadth-First . . . 194
9.3.1 Breadth-First Search . . . 194
9.3.2 Exploiting Traversal . . . 195
9.3.3 Finding Paths . . . 196
9.4 Graph Traversal: Depth-First . . . 198
9.4.1 Finding Cycles . . . 198
9.4.2 Connected Components . . . 199
9.5 Topological Sorting . . . 200
9.6 Problems . . . 203
9.6.1 Bicoloring . . . 203
9.6.2 Playing With Wheels . . . 204
9.6.3 The Tourist Guide . . . 206
9.6.4 Slash Maze . . . 208
9.6.5 Edit Step Ladders . . . 210
9.6.6 Tower of Cubes . . . 211
9.6.7 From Dusk Till Dawn . . . 213
9.6.8 Hanoi Tower Troubles Again! . . . 215
9.7 Hints . . . 216

10 Graph Algorithms **217**
10.1 Graph Theory . . . 217
10.1.1 Degree Properties . . . 217
10.1.2 Connectivity . . . 218
10.1.3 Cycles in Graphs . . . 219

10.1.4 Planar Graphs . . . 220
10.2 Minimum Spanning Trees . . . 220
10.3 Shortest Paths . . . 223
10.3.1 Dijkstra's Algorithm . . . 223
10.3.2 All-Pairs Shortest Path . . . 225
10.4 Network Flows and Bipartite Matching . . . 227
10.5 Problems . . . 231
10.5.1 Freckles . . . 231
10.5.2 The Necklace . . . 232
10.5.3 Fire Station . . . 234
10.5.4 Railroads . . . 235
10.5.5 War . . . 237
10.5.6 Tourist Guide . . . 239
10.5.7 The Grand Dinner . . . 241
10.5.8 The Problem With the Problem Setter . . . 242
10.6 Hints . . . 244

11 Dynamic Programming **245**
11.1 Don't Be Greedy . . . 245
11.2 Edit Distance . . . 246
11.3 Reconstructing the Path . . . 250
11.4 Varieties of Edit Distance . . . 251
11.5 Program Design Example: Elevator Optimization . . . 253
11.6 Problems . . . 257
11.6.1 Is Bigger Smarter? . . . 257
11.6.2 Distinct Subsequences . . . 258
11.6.3 Weights and Measures . . . 259
11.6.4 Unidirectional TSP . . . 260
11.6.5 Cutting Sticks . . . 262
11.6.6 Ferry Loading . . . 263
11.6.7 Chopsticks . . . 265
11.6.8 Adventures in Moving: Part IV . . . 266
11.7 Hints . . . 267
11.8 Notes . . . 267

12 Grids **268**
12.1 Rectilinear Grids . . . 268
12.1.1 Traversal . . . 269
12.1.2 Dual Graphs and Representations . . . 270
12.2 Triangular and Hexagonal Grids . . . 271
12.2.1 Triangular Lattices . . . 271
12.2.2 Hexagonal Lattices . . . 272
12.3 Program Design Example: Plate Weight . . . 275
12.4 Circle Packings . . . 277
12.5 Longitude and Latitude . . . 278

12.6 Problems . . . 279
12.6.1 Ant on a Chessboard . . . 279
12.6.2 The Monocycle . . . 280
12.6.3 Star . . . 282
12.6.4 Bee Maja . . . 283
12.6.5 Robbery . . . 284
12.6.6 (2/3/4)-D Sqr/Rects/Cubes/Boxes? . . . 286
12.6.7 Dermuba Triangle . . . 287
12.6.8 Airlines . . . 288
12.7 Hints . . . 290

13 Geometry **291**
13.1 Lines . . . 291
13.2 Triangles and Trigonometry . . . 294
13.2.1 Right Triangles and the Pythagorean Theorem . . . 295
13.2.2 Trigonometric Functions . . . 295
13.2.3 Solving Triangles . . . 296
13.3 Circles . . . 298
13.4 Program Design Example: Faster Than a Speeding Bullet . . . 299
13.5 Trigonometric Function Libraries . . . 302
13.6 Problems . . . 304
13.6.1 Dog and Gopher . . . 304
13.6.2 Rope Crisis in Ropeland! . . . 305
13.6.3 The Knights of the Round Table . . . 306
13.6.4 Chocolate Chip Cookies . . . 307
13.6.5 Birthday Cake . . . 308
13.6.6 The Largest/Smallest Box 309
13.6.7 Is This Integration? . . . 310
13.6.8 How Big Is It? . . . 311
13.7 Hints . . . 312

14 Computational Geometry **313**
14.1 Line Segments and Intersection . . . 313
14.2 Polygons and Angle Computations . . . 315
14.3 Convex Hulls . . . 316
14.4 Triangulation: Algorithms and Related Problems . . . 319
14.4.1 Van Gogh's Algorithm . . . 320
14.4.2 Area Computations . . . 322
14.4.3 Point Location . . . 322
14.5 Algorithms on Grids . . . 324
14.5.1 Range Queries . . . 324
14.5.2 Lattice Polygons and Pick's Theorem . . . 325
14.6 Geometry Libraries . . . 326
14.7 Problems . . . 327
14.7.1 Herding Frosh . . . 327

14.7.2 The Closest Pair Problem . . . 328
14.7.3 Chainsaw Massacre . . . 329
14.7.4 Hotter Colder . . . 330
14.7.5 Useless Tile Packers . . . 331
14.7.6 Radar Tracking . . . 333
14.7.7 Trees on My Island . . . 334
14.7.8 Nice Milk . . . 336
14.8 Hints . . . 337

A Appendix **339**
A.1 The ACM International Collegiate Programming Contest . . . 339
A.1.1 Preparation . . . 340
A.1.2 Strategies and Tactics . . . 341
A.2 International Olympiad in Informatics . . . 343
A.2.1 Participation . . . 343
A.2.2 Format . . . 344
A.2.3 Preparation . . . 345
A.3 Topcoder.com . . . 345
A.4 Go to Graduate School! . . . 346
A.5 Problem Credits . . . 348

References **350**

Index **353**