A guide to algorithm design
Paradigms, methods and complexity analysis
## Contents

Preface xv

I Polynomial-time algorithms: Exercises 1

1 Introduction to complexity 3

1.1 On the complexity to compute $x^n$ 3

1.1.1 Naive method 4

1.1.2 Binary method 4

1.1.3 Factorization method 4

1.1.4 Knuth’s tree method 5

1.1.5 Complexity results 6

1.2 Asymptotic notations: $O$, $o$, $\Theta$, and $\Omega$ 8

1.3 Exercises 8

Exercise 1.1: Longest balanced section 8

Exercise 1.2: Find the star 9

Exercise 1.3: Breaking boxes 9

Exercise 1.4: Maximum of $n$ integers 10

Exercise 1.5: Maximum and minimum of $n$ integers 10

Exercise 1.6: Maximum and second maximum of $n$ integers 11

Exercise 1.7: Merging two sorted sets 11

Exercise 1.8: The toolbox 12

Exercise 1.9: Sorting a small number of objects 12

1.4 Solutions to exercises 14

Solution to Exercise 1.1: Longest balanced section 14

Solution to Exercise 1.2: Find the star 15

Solution to Exercise 1.3: Breaking boxes 16

Solution to Exercise 1.4: Maximum of $n$ integers 17

Solution to Exercise 1.5: Maximum and minimum of $n$ integers 20

Solution to Exercise 1.6: Maximum and second maximum of $n$ integers 23

Solution to Exercise 1.7: Merging two sorted sets 25

Solution to Exercise 1.8: The toolbox 26

Solution to Exercise 1.9: Sorting a small number of objects 29

1.5 Bibliographical notes 31
Solution to Exercise 3.5: Scheduling independent tasks with deadlines ........................................ 73
Solution to Exercise 3.6: Edge matroids .................................. 74
Solution to Exercise 3.7: Huffman code ................................ 75
3.7 Bibliographical notes .................................................. 79

4 Dynamic programming .................................................. 81
4.1 The coin changing problem ......................................... 81
4.2 The knapsack problem ................................................. 84
4.3 Designing dynamic-programming algorithms .................... 86
4.4 Exercises ............................................................... 87
   Exercise 4.1: Matrix chains ........................................... 87
   Exercise 4.2: The library .............................................. 88
   Exercise 4.3: Polygon triangulation ................................ 88
   Exercise 4.4: Square of ones ........................................ 89
   Exercise 4.5: The wind band ........................................... 89
   Exercise 4.6: Ski rental ............................................... 89
   Exercise 4.7: Building set ............................................ 90
4.5 Solutions to exercises .................................................. 90
   Solution to Exercise 4.1: Matrix chains ............................. 90
   Solution to Exercise 4.2: The library ................................ 91
   Solution to Exercise 4.3: Polygon triangulation .................... 93
   Solution to Exercise 4.4: Square of ones ........................... 96
   Solution to Exercise 4.5: The wind band ............................ 98
   Solution to Exercise 4.6: Ski rental .................................. 98
   Solution to Exercise 4.7: Building set ............................. 102
4.6 Bibliographical notes ................................................ 103

5 Amortized analysis ...................................................... 105
5.1 Methods for amortized analysis .................................. 105
   5.1.1 Running examples ............................................ 105
   5.1.2 Aggregate analysis ............................................. 106
   5.1.3 Accounting method ............................................ 106
   5.1.4 Potential method .............................................. 107
5.2 Exercises ............................................................... 108
   Exercise 5.1: Binary counter ......................................... 108
   Exercise 5.2: Inserting and deleting ................................ 108
   Exercise 5.3: Stack ..................................................... 109
   Exercise 5.4: Deleting half the elements ............................ 109
   Exercise 5.5: Searching and inserting ............................... 109
   Exercise 5.6: Splay trees .............................................. 110
   Exercise 5.7: Half perimeter of a polygon ........................... 112
5.3 Solutions to exercises .................................................. 112
   Solution to Exercise 5.1: Binary counter ........................... 112
   Solution to Exercise 5.2: Inserting and deleting .................. 113
Solution to Exercise 5.3: Stack ......................... 114
Solution to Exercise 5.4: Deleting half the elements .... 115
Solution to Exercise 5.5: Searching and inserting ........ 116
Solution to Exercise 5.6: Splay trees ..................... 117
Solution to Exercise 5.7: Half perimeter of a polygon .... 119

5.4 Bibliographical notes ................................ 122

II NP-completeness and beyond 123

6 NP-completeness 125
  6.1 A practical approach to complexity theory .......... 125
  6.2 Problem classes .................................... 126
      6.2.1 Problems in P ............................... 127
      6.2.2 Problems in NP .............................. 129
  6.3 NP-complete problems and reduction theory ......... 132
      6.3.1 Polynomial reduction ......................... 132
      6.3.2 Cook's theorem ................................ 133
      6.3.3 Growing the class NPC of NP-complete problems 134
      6.3.4 Optimization problems versus decision problems 135
  6.4 Examples of NP-complete problems and reductions ... 136
      6.4.1 3-SAT ......................................... 136
      6.4.2 CLIQUE ....................................... 138
      6.4.3 VERTEX-COVER ............................... 139
      6.4.4 Scheduling problems ......................... 140
      6.4.5 Other famous NP-complete problems ............ 142
  6.5 Importance of problem definition .................... 143
  6.6 Strong NP-completeness ............................. 145
  6.7 Why does it matter? ................................ 146
  6.8 Bibliographical notes ................................ 146

7 Exercises on NP-completeness 149
  7.1 Easy reductions .................................... 149
    Exercise 7.1: Wheel ................................. 149
    Exercise 7.2: Knights of the round table .......... 149
    Exercise 7.3: Variants of CLIQUE ................... 149
    Exercise 7.4: Path with vertex pairs .............. 150
    Exercise 7.5: VERTEX-COVER with even degrees .... 150
    Exercise 7.6: Around 2-PARTITION .................. 150
  7.2 About graph coloring ................................ 151
    Exercise 7.7: COLOR ................................ 151
    Exercise 7.8: 3-COLOR ............................. 151
    Exercise 7.9: 3-COLOR-PLAN ....................... 152
  7.3 Scheduling problems ............................... 152
    Exercise 7.10: Scheduling independent tasks with p processors 152
Exercise 7.11: Scheduling with two processors.

7.4 More involved reductions

Exercise 7.12: Transitive subchain

Exercise 7.13: INDEPENDENT SET

Exercise 7.14: DOMINATING SET

Exercise 7.15: Carpenter

Exercise 7.16: \( k \)-center

Exercise 7.17: Variants of 3-SAT

Exercise 7.18: Variants of SAT

7.5 2-PARTITION is NP-complete

Exercise 7.19: SUBSET-SUM

Exercise 7.20: NP-completeness of 2-PARTITION

7.6 Solutions to exercises

Solution to Exercise 7.1: Wheel

Solution to Exercise 7.2: Knights of the round table

Solution to Exercise 7.3: Variants of CLIQUE

Solution to Exercise 7.4: Path with vertex pairs

Solution to Exercise 7.5: VERTEX-COVER with even degrees

Solution to Exercise 7.6: Around 2-PARTITION

Solution to Exercise 7.7: COLOR

Solution to Exercise 7.8: 3-COLOR

Solution to Exercise 7.9: 3-COLOR-PLAN

Solution to Exercise 7.10: Scheduling independent tasks with \( p \) processors

Solution to Exercise 7.11: Scheduling with two processors

Solution to Exercise 7.12: Transitive subchain

Solution to Exercise 7.13: INDEPENDENT SET

Solution to Exercise 7.14: DOMINATING SET

Solution to Exercise 7.15: Carpenter

Solution to Exercise 7.16: \( k \)-center

Solution to Exercise 7.17: Variants of 3-SAT

Solution to Exercise 7.18: Variants of SAT

Solution to Exercise 7.19: SUBSET-SUM

Solution to Exercise 7.20: NP-completeness of 2-PARTITION

7.7 Bibliographical notes

8 Beyond NP-completeness

8.1 Approximation results

8.1.1 Approximation algorithms

8.1.2 Vertex cover

8.1.3 Traveling salesman problem (TSP)

8.1.4 Bin packing

8.1.5 2-PARTITION

8.2 Polynomial problem instances

8.2.1 Partitioning problems
8.2.2 Assessing problem complexity ........................................... 194
8.3 Linear programming ......................................................... 195
8.3.1 Formal definition ......................................................... 195
8.3.2 Relaxation and rounding ................................................. 197
8.4 Randomized algorithms ...................................................... 200
8.4.1 The algorithm ............................................................. 201
8.4.2 Results ................................................................. 201
8.5 Branch-and-bound and backtracking .................................... 202
8.5.1 Backtracking: The $n$ queens ......................................... 203
8.5.2 Branch-and-bound: The knapsack ................................... 204
8.5.3 Graph algorithms ....................................................... 206
8.6 Bibliographical notes ........................................................ 209

9 Exercises going beyond NP-completeness 211
9.1 Approximation results ........................................................ 211
Exercise 9.1: Single machine scheduling .................................... 211
Exercise 9.2: SUBSET-SUM ................................................... 212
Exercise 9.3: SET-COVER .................................................... 213
Exercise 9.4: VERTEX-COVER ............................................ 213
Exercise 9.5: Scheduling independent tasks in parallel ................ 215
Exercise 9.6: Point clustering ................................................... 215
Exercise 9.7: $k$-center ........................................................ 216
Exercise 9.8: Knapsack .......................................................... 217
9.2 Dealing with NP-complete problems ...................................... 218
Exercise 9.9: Mixed integer linear program for replica placement .......... 218
Exercise 9.10: A randomized algorithm for independent set ........ 218
Exercise 9.11: Branch-and-bound applied to MAX-SAT ............... 219
9.3 Solutions to exercises ........................................................ 219
Solution to Exercise 9.1: Single machine scheduling .................... 219
Solution to Exercise 9.2: SUBSET-SUM ................................... 221
Solution to Exercise 9.3: SET-COVER .................................... 223
Solution to Exercise 9.4: VERTEX-COVER ................................ 224
Solution to Exercise 9.5: Scheduling independent tasks in parallel ..... 226
Solution to Exercise 9.6: Point clustering ................................ 228
Solution to Exercise 9.7: $k$-center ........................................ 229
Solution to Exercise 9.8: Knapsack ........................................ 231
Solution to Exercise 9.9: Mixed integer linear program for replica placement ................................................. 234
Solution to Exercise 9.10: A randomized algorithm for independent set .......................................................... 237
Solution to Exercise 9.11: Branch-and-bound applied to MAX-SAT .... 237
9.4 Bibliographical notes ........................................................ 238
# III Reasoning on problem complexity

## 10 Reasoning to assess a problem complexity

10.1 Basic reasoning .......................... 241
   10.1.1 Polynomial instances ............... 241
   10.1.2 NP-complete instances .............. 242
10.2 Set of problems with polynomial-time algorithms .......... 243
10.3 Set of NP-complete problems .................. 244
   10.3.1 Numbers ............................ 245
   10.3.2 Graphs ............................. 246

## 11 Chains-on-chains partitioning

11.1 Optimal algorithms for homogeneous resources .......... 249
   11.1.1 Dynamic-programming algorithm .......... 250
   11.1.2 Binary search algorithm .............. 250
   11.1.3 Improved algorithms .................... 250
11.2 Variants of the problem ........................ 252
   11.2.1 Communication costs .................... 252
   11.2.2 Chain of heterogeneous resources .......... 253
11.3 Extension to a clique of heterogeneous resources ........ 254
   11.3.1 NP-completeness ...................... 254
   11.3.2 Practical solutions .................... 257
   11.3.3 Integer linear program .................. 257
11.4 Conclusion ................................ 258

## 12 Replica placement in tree networks

12.1 Access policies .......................... 262
   12.1.1 Motivation ............................ 262
   12.1.2 Impact of the policies on the existence of a solution . 263
   12.1.3 Impact of the policies on the cost of a solution .... 264
12.2 Complexity results ........................ 266
   12.2.1 Definitions ............................ 266
   12.2.2 MinNb problem ....................... 267
   12.2.3 MinCOST problem ..................... 273
   12.2.4 Integer linear program .............. 275
12.3 Variants of the replica placement problem .............. 279
   12.3.1 Enforcing a quality of service .......... 280
   12.3.2 Power-aware replica placement .......... 282
12.4 Conclusion ................................ 286

## 13 Packet routing

13.1 MEDP: Maximum edge-disjoint paths ...................... 288
   13.1.1 Problem statement ..................... 288
   13.1.2 Naive greedy algorithm .................. 289
   13.1.3 Short-requests-first greedy algorithm .... 291
# List of exercises

1.1 Longest balanced section ............................................ 8
1.2 Find the star ............................................................. 9
1.3 Breaking boxes ......................................................... 9
1.4 Maximum of $n$ integers .............................................. 10
1.5 Maximum and minimum of $n$ integers ............................. 10
1.6 Maximum and second maximum of $n$ integers ................... 11
1.7 Merging two sorted sets ............................................. 11
1.8 The toolbox .............................................................. 12
1.9 Sorting a small number of objects ................................. 12

2.1 Product of two polynomials ........................................... 39
2.2 Toeplitz matrices ....................................................... 40
2.3 Maximum sum ........................................................... 40
2.4 Boolean matrices: The Four-Russians algorithm .................. 41
2.5 Matrix multiplication and inversion ................................. 42

3.1 Interval cover ............................................................ 64
3.2 Memory usage ........................................................... 64
3.3 Scheduling dependent tasks on several machines ............... 65
3.4 Scheduling independent tasks with priorities .................... 66
3.5 Scheduling independent tasks with deadlines .................... 66
3.6 Edge matroids ............................................................ 67
3.7 Huffman code ............................................................ 67

4.1 Matrix chains ........................................................... 87
4.2 The library ............................................................... 88
4.3 Polygon triangulation .................................................. 88
4.4 Square of ones .......................................................... 89
4.5 The wind band .......................................................... 89
4.6 Ski rental ............................................................... 89
4.7 Building set ............................................................ 90

5.1 Binary counter .......................................................... 108
5.2 Inserting and deleting .................................................. 108
5.3 Stack ................................................................. 109
5.4 Deleting half the elements .......................................... 109
5.5 Searching and inserting .............................................. 109
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.6</td>
<td>Splay trees</td>
<td>110</td>
</tr>
<tr>
<td>5.7</td>
<td>Half perimeter of a polygon</td>
<td>112</td>
</tr>
<tr>
<td>7.1</td>
<td>Wheel</td>
<td>149</td>
</tr>
<tr>
<td>7.2</td>
<td>Knights of the round table</td>
<td>149</td>
</tr>
<tr>
<td>7.3</td>
<td>Variants of CLIQUE</td>
<td>149</td>
</tr>
<tr>
<td>7.4</td>
<td>Path with vertex pairs</td>
<td>150</td>
</tr>
<tr>
<td>7.5</td>
<td>VERTEX-COVER with even degrees</td>
<td>150</td>
</tr>
<tr>
<td>7.6</td>
<td>Around 2-PARTITION</td>
<td>150</td>
</tr>
<tr>
<td>7.7</td>
<td>COLOR</td>
<td>151</td>
</tr>
<tr>
<td>7.8</td>
<td>3-COLOR</td>
<td>151</td>
</tr>
<tr>
<td>7.9</td>
<td>3-COLOR-PLAN</td>
<td>152</td>
</tr>
<tr>
<td>7.10</td>
<td>Scheduling independent tasks with p processors</td>
<td>152</td>
</tr>
<tr>
<td>7.11</td>
<td>Scheduling with two processors</td>
<td>152</td>
</tr>
<tr>
<td>7.12</td>
<td>Transitive subchain</td>
<td>153</td>
</tr>
<tr>
<td>7.13</td>
<td>INDEPENDENT SET</td>
<td>153</td>
</tr>
<tr>
<td>7.14</td>
<td>DOMINATING SET</td>
<td>153</td>
</tr>
<tr>
<td>7.15</td>
<td>Carpenter</td>
<td>153</td>
</tr>
<tr>
<td>7.16</td>
<td>k-center</td>
<td>153</td>
</tr>
<tr>
<td>7.17</td>
<td>Variants of 3-SAT</td>
<td>154</td>
</tr>
<tr>
<td>7.18</td>
<td>Variants of SAT</td>
<td>154</td>
</tr>
<tr>
<td>7.19</td>
<td>SUBSET-SUM</td>
<td>155</td>
</tr>
<tr>
<td>7.20</td>
<td>NP-completeness of 2-PARTITION</td>
<td>155</td>
</tr>
<tr>
<td>9.1</td>
<td>Single machine scheduling</td>
<td>211</td>
</tr>
<tr>
<td>9.2</td>
<td>SUBSET-SUM</td>
<td>212</td>
</tr>
<tr>
<td>9.3</td>
<td>SET-COVER</td>
<td>213</td>
</tr>
<tr>
<td>9.4</td>
<td>VERTEX-COVER</td>
<td>213</td>
</tr>
<tr>
<td>9.5</td>
<td>Scheduling independent tasks in parallel</td>
<td>215</td>
</tr>
<tr>
<td>9.6</td>
<td>Point clustering</td>
<td>215</td>
</tr>
<tr>
<td>9.7</td>
<td>k-center</td>
<td>216</td>
</tr>
<tr>
<td>9.8</td>
<td>Knapsack</td>
<td>217</td>
</tr>
<tr>
<td>9.9</td>
<td>Mixed integer linear program for replica placement</td>
<td>218</td>
</tr>
<tr>
<td>9.10</td>
<td>A randomized algorithm for independent set</td>
<td>218</td>
</tr>
<tr>
<td>9.11</td>
<td>Branch-and-bound applied to MAX-SAT</td>
<td>219</td>
</tr>
</tbody>
</table>