Contents

Preface  xiii

1 Introduction  1

1.1 Some Computing Puzzles  1
1.1.1 A Restricted Version of the C Language  2
1.1.2 Problem 1—Detecting Strings in a Language  3
1.1.3 Problem 2—Execution Configuration  4
1.1.4 Problem 3—Deducing Behavior from Structure  6
1.1.5 Problem 4—The Halting Problem  10

1.2 What Is Computability?  13

1.3 Related Works  15

1.4 Overview of This Book  16

1.5 Exercises  17

2 Languages and Problems  19

2.1 Introduction  19

2.2 Symbols, Alphabets, and Strings  23
2.2.1 Strings  24
2.2.2 Lexicographical and Enumeration Ordering  25
2.2.3 String Concatenation  26
2.2.4 Parts of Strings  27
2.2.5 String Reversal

2.3 Languages

2.4 Operations on Languages
2.4.1 Union
2.4.2 Complementation
2.4.3 Intersection
2.4.4 Symmetric Difference
2.4.5 DeMorgan's Laws
2.4.6 Length Subsetting
2.4.7 Concatenation
2.4.8 Reversal
2.4.9 Kleene Star

2.5 Alphabet Encodings

2.6 Some Test Languages

2.7 How Many Languages Are There?
2.7.1 Most Languages Do Not Have Compilers

2.8 Problem Representations

2.9 Types of Problems
2.9.1 Representing Graphs
2.9.2 Spanning Trees
2.9.3 Decision Problems
2.9.4 Function Problems
2.9.5 Search Problems

2.10 Casting Problems into Languages

2.11 Decision, Search, and Enumeration

2.12 Language Operations and Enumeration

2.13 Exercises

3 Regular Expressions and Languages

3.1 Introduction
3.2 Regular Languages and Constructions 67
3.3 Regular Expressions 69
  3.3.1 The Grammar of Standard Regular Expressions 70
  3.3.2 Examples of Standard Regular Expressions 71
  3.3.3 Consistency and Completeness of Standard Regular Expressions 72
3.4 Not All Languages Are Regular 75
3.5 Applications of Regular Expressions 74
  3.5.1 Validation 74
  3.5.2 Search and Selection 76
  3.5.3 Tokenization 78
3.6 Exercises 79

4 Fundamental Machines Part I: Finite-State Control Machines 83

4.1 Introduction 83
  4.1.1 The Church-Turing Thesis 84
4.2 Basic Machine Notions 85
  4.2.1 Tapes 86
  4.2.2 Finite Controls 89
4.3 Deterministic Finite Automata 90
  4.3.1 DFA Design 100
4.4 Nondeterministic Finite Automata 103
4.5 Equivalence of DFAs and NFAs 107
4.6 Equivalence of DFAs and C-- Programs 115
4.7 Exercises 116

5 Properties of Finite-State Languages 121

5.1 Introduction 121
5.2 Machines for Five Language Operations 122
  5.2.1 Closure under Complement 123
5.2.2 Closure under Union 125
5.2.3 Closure under Intersection 127
5.2.4 Closure under Concatenation 128
5.2.5 Closure under Kleene Star 130

5.3 Equivalence of Regular Expressions and Finite Automata 132
5.4 Pumping Lemma for Regular Languages 142
5.5 Applications of the Pumping Lemma 145
5.6 Using Closure Properties to Deduce That a Language Is Nonregular 147
5.7 Exercises 148

6 Fundamental Machines Part II: Stack and Tape Machines 155

6.1 Pushdown Automata 155
  6.1.1 Deterministic Pushdown Automata 156
  6.1.2 Nondeterministic Pushdown Automata 160
  6.1.3 Inequivalence of DPDAs and PDAs 161

6.2 Turing Machines 163
  6.2.1 Deterministic Turing Machines 165
  6.2.2 Multiple-Work-Tape Turing Machines 175
  6.2.3 Nondeterministic Turing Machines 180
  6.2.4 Equivalence of DTMs and NTMs 184

6.3 Undecidable Languages 187
6.4 Relations among Language Classes 188
6.5 Exercises 189

7 Grammars 195

7.1 Introduction 195
7.2 Regular Grammars 195
7.3 Context-Free Grammars 199
## Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.4</td>
<td>Closure Properties of Context-Free Languages</td>
<td>204</td>
</tr>
<tr>
<td>7.5</td>
<td>Parsing with Nondeterministic Pushdown Automata</td>
<td>206</td>
</tr>
<tr>
<td>7.6</td>
<td>Parsing with Deterministic Pushdown Automata</td>
<td>208</td>
</tr>
<tr>
<td>7.7</td>
<td>Parse Trees and Attribution</td>
<td>209</td>
</tr>
<tr>
<td>7.8</td>
<td>Languages That Are Not Context-Free</td>
<td>212</td>
</tr>
<tr>
<td>7.9</td>
<td>Exercises</td>
<td>216</td>
</tr>
<tr>
<td>8</td>
<td>Computational Complexity</td>
<td>221</td>
</tr>
<tr>
<td>8.1</td>
<td>Introduction</td>
<td>221</td>
</tr>
<tr>
<td>8.2</td>
<td>Asymptotic Notation</td>
<td>222</td>
</tr>
<tr>
<td>8.3</td>
<td>Time and Space Complexity</td>
<td>225</td>
</tr>
<tr>
<td>8.4</td>
<td>Simulations</td>
<td>229</td>
</tr>
<tr>
<td>8.5</td>
<td>Reducibilities</td>
<td>234</td>
</tr>
<tr>
<td>8.5.1</td>
<td>Many-One Reducibility</td>
<td>235</td>
</tr>
<tr>
<td>8.6</td>
<td>Exercises</td>
<td>237</td>
</tr>
<tr>
<td>9</td>
<td>Circuit Complexity</td>
<td>241</td>
</tr>
<tr>
<td>9.1</td>
<td>Introduction</td>
<td>241</td>
</tr>
<tr>
<td>9.2</td>
<td>The Boolean Circuit Model of Computation</td>
<td>242</td>
</tr>
<tr>
<td>9.3</td>
<td>Circuit Resources</td>
<td>247</td>
</tr>
<tr>
<td>9.3.1</td>
<td>Uniform Circuit Families</td>
<td>248</td>
</tr>
<tr>
<td>9.4</td>
<td>Examples of Boolean Circuits</td>
<td>250</td>
</tr>
<tr>
<td>9.5</td>
<td>The Complexity Class NC</td>
<td>253</td>
</tr>
<tr>
<td>9.6</td>
<td>Exercises</td>
<td>254</td>
</tr>
<tr>
<td>10</td>
<td>Feasible Problems</td>
<td>259</td>
</tr>
<tr>
<td>10.1</td>
<td>Introduction</td>
<td>259</td>
</tr>
<tr>
<td>10.2</td>
<td>Polynomial Time</td>
<td>261</td>
</tr>
<tr>
<td>10.3</td>
<td>P-Completeness Theory</td>
<td>263</td>
</tr>
</tbody>
</table>
## Contents

### 10.4 Examples of P–Complete Problems
- 10.4.1 Generic Machine Simulation
- 10.4.2 The Circuit Value Problem and Some Variants

### 10.5 P–Complete Problems and Reductions
- 10.5.1 NAND Circuit Value Problem
- 10.5.2 Context–Free Grammar Nonempty
- 10.5.3 Lexicographically First Maximal Independent Set

### 10.6 Exercises

### 11 Intractable Problems

#### 11.1 Introduction

#### 11.2 Nondeterministic Polynomial Time

#### 11.3 NP–Completeness Theory

#### 11.4 Examples of NP–Complete Problems
- 11.4.1 Satisfiability and Some Variants
- 11.4.2 Hamiltonian Circuit Problem

#### 11.5 NP–Complete Problems and Reductions
- 11.5.1 Traveling Salesperson Problem
- 11.5.2 Vertex Cover
- 11.5.3 Three-Dimensional Matching

#### 11.6 Exercises

### Appendix A Notation

### Appendix B Greek Alphabet

### Bibliography

### Index