
Inhaltsverzeichnis

Vorwort	IX
Vorwort zur 2. Auflage	XII
I Die Berechenbarkeit von Funktionen	1
1 Theoretische Berechenbarkeit	3
1.1 Der Begriff des Algorithmus	3
1.2 Die μ -rekursiven Funktionen	4
1.3 Turingmaschinen	10
1.4 Entscheidbarkeit	23
1.5 Ergänzende Übungsaufgaben	26
1.6 Literaturhinweise	27
2 Praktische Berechenbarkeit	28
2.1 Die Random Access Maschine	28
2.2 Die Sprachklassen P und NP	34
2.3 NP-vollständige Probleme	39
2.4 Kryptographie	55
2.4.1 Public-Key Kryptosysteme	55
2.4.2 Zero-Knowledge Beweise	59
2.5 Ergänzende Übungsaufgaben	65
2.6 Literaturhinweise	66
II Grundlegendes zum Lösen von Problemen	69
3 Datenstrukturen zur Lösung von Mengenverwaltungsproblemen	71
3.1 Einfache Datenstrukturen	72
3.1.1 Keller und Schlangen	73
3.1.2 Listen	74
3.2 Bäume	76
3.2.1 Beliebige Suchbäume	79

3.2.2	AVL-Bäume	82
3.2.3	B-Bäume	87
3.2.4	Tries	93
3.3	Hashing	95
3.3.1	Kollisionsbehandlung mittels verketteter Listen	95
3.3.2	Kollisionsbehandlung mittels offener Adressierung	100
3.3.3	Universelles Hashing	101
3.4	Datenstrukturen für disjunkte Mengen	104
3.5	Priority Queues	105
3.5.1	<i>d</i> -Heaps	106
3.5.2	Linksheaps	109
3.6	Ergänzende Übungsaufgaben	110
3.7	Literaturhinweise	112
4	Durchmusterung von Graphen	113
4.1	Graphentheoretische Grundlagen	113
4.2	Graphdurchmusterungsmethoden	116
4.3	Ergänzende Übungsaufgaben	119
4.4	Literaturhinweise	120
5	Über den Entwurf von Algorithmen	121
5.1	Divide-and-conquer	122
5.2	Dynamische Programmierung	124
5.3	Aufzählungsmethoden	125
5.4	Greedyalgorithmen	132
5.5	Approximationsalgorithmen	136
5.6	Probabilistische Algorithmen	137
5.6.1	Ein Las Vegas-Algorithmus	138
5.6.2	Ein Monte-Carlo-Algorithmus	141
5.7	Ergänzende Übungsaufgaben	145
5.8	Literaturhinweise	146
III	Die Lösung elementarer Probleme	147
6	Automatentheorie und Formale Sprachen	149
6.1	Die lexikalische Analyse	149
6.1.1	Reguläre Mengen, reguläre Ausdrücke und endliche Automaten	150
6.1.2	Minimierung endlicher Automaten	159
6.1.3	Zur Realisierung der lexikalischen Analyse	167
6.2	Die Syntaxanalyse	168
6.2.1	Kontextfreie Grammatiken	169
6.2.2	Kellerautomaten	175

6.2.3	Normalformen für kontextfreie Grammatiken	184
6.3	Eigenschaften von kontextfreien Sprachen	192
6.3.1	Das Pumping-Lemma	192
6.3.2	Abschlußeigenschaften	195
6.4	Ergänzende Übungsaufgaben	196
6.5	Literaturhinweise	198
7	Algorithmen auf Graphen	200
7.1	Bipartites Matching	200
7.1.1	Der ungewichtete Fall	201
7.1.2	Der gewichtete Fall	205
7.2	Starke Zusammenhangskomponenten	212
7.3	Kürzeste-Weg-Probleme	215
7.3.1	Dijkstras Algorithmus	220
7.3.2	Der Bellman-Ford Algorithmus	222
7.3.3	Das alle-Paare-kürzeste-Weg-Problem	224
7.4	Minimale überspannende Bäume	228
7.4.1	Der Algorithmus von Kruskal	230
7.4.2	Der Algorithmus von Prim	231
7.5	Netzwerkflußprobleme	232
7.6	Das Graphenisomorphieproblem	241
7.7	Ergänzende Übungsaufgaben	248
7.8	Literaturhinweise	250
8	Algorithmen auf Strings	252
8.1	Stringmatching	252
8.2	Approximatives Stringmatching	255
8.3	Ergänzende Übungsaufgaben	261
8.4	Literaturhinweise	262
9	Online-Algorithmen	263
9.1	Das Listenverwaltungsproblem	263
9.2	Paging	269
9.3	Ergänzende Übungsaufgaben	275
9.4	Literaturhinweise	275
10	Schnelle Fouriertransformationen	276
10.1	Wahl der Stützstellen	277
10.2	Ergänzende Übungsaufgaben	281
10.3	Literaturhinweise	281
11	Lineare Programmierung	282
11.1	Grundlagen	284

11.2	Die Simplexmethode	291
11.3	Dualität	302
11.4	Ganzzahlige lineare Programmierung	312
11.5	Ergänzende Übungsaufgaben	320
11.6	Literaturhinweise	322
	 Literaturverzeichnis	 325
	 Index	 333