

Inhaltsverzeichnis

Vorwort zur 1. Auflage	xi
Vorwort zur 2. Auflage	xiii
Vorwort zur 3. Auflage	xiv
Vorwort zur 4. Auflage	xv
Vorwort zur 5. Auflage	xvi
1 Einführung und Übersicht	1
1.1 Ausgangspunkte für das Themengebiet	1
1.2 Anwendungen theoretischer Erkenntnisse	3
1.3 Stoffübersicht und -abgrenzung	4
1.4 Externe Lernhilfen und Web-Seiten	5
1.5 Allgemeine Bibliographische Hinweise	6
I Endliche Automaten und reguläre Sprachen	7
2 Endliche Automaten	9
2.1 Deterministische endliche Automaten	9
2.1.1 Beispiel: Der Automat A_{Eintritt}	10
2.1.2 Alphabete, Wörter, Sprachen	13
2.1.3 Zustände und Zustandsübergänge	19
2.1.4 Deterministische endliche Automaten und reguläre Sprachen .	20
2.1.5 Vollständige Automaten	25
2.1.6 Zusammenfassung	26
2.2 Nichtdeterministische endliche Automaten	26
2.2.1 Definitionen	26
2.2.2 Äquivalenz von deterministischen und nichtdeterministischen endlichen Automaten	31
2.2.3 Zusammenfassung	36
2.3 Endliche Automaten mit ε -Übergängen	36

2.3.1	Definitionen	36
2.3.2	Äquivalenz von ε -Automaten zu nichtdeterministischen endlichen Automaten	37
2.3.3	Zusammenfassung	42
2.4	Verallgemeinerte endliche Automaten	42
2.4.1	Definitionen	42
2.4.2	Äquivalenz von verallgemeinerten und endlichen Automaten	43
2.5	Minimierung endlicher Automaten	44
2.5.1	Isomorphie endlicher Automaten	45
2.5.2	Der Satz von Myhill und Nerode	46
2.5.3	Verfahren zur Minimierung endlicher Automaten	49
2.6	Anwendungen endlicher Automaten	54
2.6.1	Rechnersysteme und Systemprogrammierung	55
2.6.2	Teilworterkennung	55
2.6.3	Objektorientierte Modellierung. Interaktionsdiagramme	59
2.7	Bibliographische Hinweise und Ergänzungen	61
2.8	Übungen	62
3	Reguläre Sprachen	65
3.1	Reguläre Ausdrücke	66
3.1.1	Definitionen und Eigenschaften	66
3.1.2	Anwendung regulärer Ausdrücke	72
3.1.3	Äquivalenz von endlichen Automaten und regulären Ausdrücken	74
3.1.4	Scanner-Generatoren	79
3.1.5	Zusammenfassung	82
3.2	Typ-3-Grammatiken	82
3.2.1	Rechtslineare Grammatiken	82
3.2.2	Linkslineare Grammatiken	86
3.2.3	Äquivalenz rechtslinearer und linkslinearer Grammatiken . . .	87
3.2.4	Verallgemeinerte Typ-3-Grammatiken	88
3.2.5	Äquivalenz von endlichen Automaten und Typ-3-Grammatiken	89
3.2.6	Zusammenfassung	91
3.3	Eigenschaften regulärer Sprachen	92
3.3.1	Abschlusseigenschaften von REG_{Σ}	92
3.3.2	Das Pumping-Lemma für reguläre Sprachen	101
3.3.3	Entscheidbarkeitsprobleme	103
3.3.4	Grenzen endlicher Automaten	106
3.4	Bibliographische Hinweise und Ergänzungen	108
3.5	Übungen	109
4	Endliche Maschinen und Automatennetze	113
4.1	Endliche Maschinen	114
4.1.1	Erweiterung des endlichen Automaten $A_{Eintritt}$	114
4.1.2	Mealy-Maschinen	116
4.1.3	Ein formales Vorgehensmodell bei der Problemlösung	123

4.1.4	Moore-Maschinen	124
4.1.5	Äquivalenz von Mealy- und Moore-Berechenbarkeit	129
4.1.6	Grenzen endlicher Maschinen	133
4.2	Endliche Transducer	136
4.3	Beispiele für Automatennetze	138
4.3.1	Synchrone Automaten: Zelluläre Automaten	138
4.3.2	Asynchrone Automaten: Petri-Netze	145
4.3.3	Anwendungen und Varianten von Petri-Netzen	152
4.4	Anwendungen endlicher Maschinen	157
4.4.1	Software- und Systementwurf. Statecharts	159
4.4.2	Workflow-Management	161
4.4.3	Elektronischer Handel	164
4.4.4	Modellierung und Komposition von Web Services	166
4.5	Bibliographische Hinweise und Ergänzungen	174
4.6	Übungen	175
II	Kontextfreie Sprachen und Kellerautomaten	179
5	Kontextfreie Sprachen	181
5.1	Kontextfreie Grammatiken	181
5.1.1	Beispiele und Definitionen	182
5.1.2	Normalformen	184
5.2	Eigenschaften kontextfreier Sprachen	191
5.2.1	Mehrdeutigkeit	191
5.2.2	Das Pumping-Lemma für kontextfreie Sprachen	193
5.2.3	Abschlusseigenschaften	196
5.3	Übungen	199
6	Kellerautomaten	201
6.1	Nichtdeterministische Kellerautomaten	202
6.1.1	Grundlegende Definitionen	202
6.1.2	Akzeptieren mit leerem Keller	206
6.2	Äquivalenz von kontextfreien Grammatiken und Kellerautomaten	207
6.3	Deterministische Kellerautomaten	211
6.4	Bibliographische Hinweise	215
6.5	Übungen	215
7	Anwendungen kontextfreier Sprachen	217
7.1	Ableitungs- und Syntaxbäume	217
7.2	Compilerbau	220
7.3	Syntax von Programmiersprachen	229
7.3.1	Erweiterte Backus-Naur-Form	229
7.3.2	Syntaxdiagramme	233
7.4	Reguläre Definitionen	236

7.4.1	SADT	237
7.4.2	XML	239
7.5	Bibliographische Hinweise	243
7.6	Übungen	243
III	Berechenbarkeit und Komplexität	245
8	Typ-1- und Typ-0-Sprachen	247
8.1	Die Chomsky-Hierarchie	247
8.1.1	Typ-1-Sprachen (kontextsensitive Sprachen)	248
8.1.2	Typ-0-Sprachen (rekursiv-aufzählbare Sprachen)	250
8.1.3	Die Hierarchie	256
8.1.4	Das Wortproblem	258
8.2	Turingautomaten	260
8.2.1	Definitionen und Beispiele	260
8.2.2	Varianten von Turingautomaten	264
8.2.3	Äquivalenz von deterministischen und nichtdeterministischen Turingautomaten	265
8.2.4	Linear beschränkte Automaten	266
8.2.5	Äquivalenz zwischen Typ-1-Grammatiken und linear beschränkten Automaten	267
8.2.6	Äquivalenz zwischen Typ-0-Grammatiken und Turingautomaten	269
8.2.7	Entscheidbare Sprachen	269
8.3	Zusammenfassung	270
8.4	Bibliographische Hinweise	273
8.5	Übungen	273
9	Berechenbarkeit	277
9.1	Turing-Berechenbarkeit	277
9.1.1	Definition und Beispiele	278
9.1.2	Die Programmiersprache TURING	282
9.2	Loop-, While- und Goto-Berechenbarkeit	286
9.2.1	Die Programmiersprache LOOP	287
9.2.2	Die Programmiersprache WHILE	293
9.2.3	Die Programmiersprache GOTO	295
9.3	Primitiv rekursive und μ -rekursive Funktionen	297
9.3.1	Primitiv-rekursive Funktionen	298
9.3.2	μ -rekursive Funktionen	302
9.4	Die Churchsche These	306
9.5	Die Ackermannfunktion	311
9.6	Universelle Turingmaschinen	315
9.6.1	Codierung von Turingmaschinen	316
9.6.2	Nummerierung von Turingmaschinen	318

9.6.3	Eine Standardnummerierung für \mathcal{P}	321
9.6.4	Fundamentale Anforderungen an Programmiersprachen	323
9.6.5	Das utm-Theorem	324
9.6.6	Das smn-Theorem	325
9.6.7	Anwendungen von utm- und smn-Theorem	328
9.6.8	Bedeutung von utm- und smn-Theorem	330
9.7	Bibliographische Hinweise	332
9.8	Übungen	333
10	Entscheidbarkeit	337
10.1	Existenz unentscheidbarer Probleme	337
10.2	Entscheidbare und semi-entscheidbare Mengen	338
10.3	Reduzierbarkeit von Mengen	342
10.4	Unentscheidbare Mengen	343
10.4.1	Das Halteproblem	343
10.4.2	Das Korrektheitsproblem	347
10.4.3	Das Äquivalenzproblem	348
10.4.4	Der Satz von Rice	348
10.4.5	Das Postsche Korrespondenzproblem	349
10.4.6	Anwendungen des Postschen Korrespondenzproblems	352
10.5	Zusammenfassung	355
10.6	Bibliographische Hinweise	355
10.7	Übungen	356
11	Komplexität	359
11.1	Die O -Notation	359
11.2	Komplexität von Algorithmen	362
11.3	Wichtige Komplexitätsklassen	365
11.4	Die Klassen P und NP	366
11.4.1	Die Klasse P	366
11.4.2	Die Klasse NP	367
11.4.3	Die Klassen $EXPTIME$ und $NEXPTIME$	369
11.4.4	Das P - NP -Problem	369
11.4.5	NP -Vollständigkeit	370
11.5	Konkrete NP -vollständige Probleme	373
11.5.1	SAT – Das Erfüllbarkeitsproblem der Aussagenlogik	373
11.5.2	Weitere NP -vollständige Probleme	378
11.6	Weitere Komplexitätsklassen	385
11.6.1	Die Klasse $PSPACE$	385
11.6.2	Komplementäre Komplexitätsklassen	389
11.7	Zusammenfassung	391
11.8	Bibliographische Hinweise und Ergänzungen	392
11.9	Übungen	393

12 Approximative und probabilistische Ansätze und deren Anwendungen	395
12.1 Approximative Algorithmen für <i>NP</i> -vollständige Probleme	397
12.1.1 Approximierbarkeit	397
12.1.2 Lokale Verbesserung am Beispiel <i>TSP</i>	400
12.1.3 Untere Schranken für das Approximieren	403
12.1.4 <i>TSP</i> in der Praxis	405
12.2 Probabilistische Algorithmen und Komplexitätsklassen	406
12.2.1 Die Klasse <i>RP</i>	407
12.2.2 Die Klasse <i>ZPP</i>	410
12.2.3 Die Klasse <i>BPP</i>	411
12.2.4 Anwendung: Verschlüsselung	412
12.2.5 Anwendung: Sortieren	420
12.3 Interaktive Beweissysteme	420
12.4 Zero Knowledge Beweise. Anwendung: Authentifikation	424
12.5 Zusammenfassung	430
12.6 Bibliographische Hinweise und Ergänzungen	431
12.7 Übungen	432
Literaturverzeichnis	435
Index	443