

Inhaltsverzeichnis

1.	Formale Sprachen	1
1.1	Relationen und Graphen	2
1.1.1	Zweistellige Relationen	2
1.1.2	Wege in Graphen und Hüllenbildung	6
1.2	Grammatiken	13
1.2.1	Reduktive und generative Grammatiken	18
1.2.2	Die Sprachhierarchie nach Chomsky	22
1.2.3	Strukturgraphen und Strukturbäume	23
1.2.4	Sackgassen und unendliche Ableitungspfade	29
1.3	Chomsky-3-Sprachen und endliche Automaten	34
1.3.1	Reguläre Ausdrücke	34
1.3.2	Endliche Automaten	36
1.3.3	Äquivalenz der Darstellungsformen	39
1.3.4	Reguläre Ausdrücke, endliche Automaten und Chomsky-3-Sprachen	42
1.3.5	Minimale Automaten	50
1.4	Kontextfreie Sprachen und Kellerautomaten	52
1.4.1	Die BNF-Notation	52
1.4.2	Kellerautomaten	53
1.4.3	Kellerautomaten und kontextfreie Sprachen	56
1.4.4	Greibach-Normalform	59
1.4.5	LR(k)-Sprachen	61
1.4.6	LL(k)-Grammatiken	66
1.4.7	Das Verfahren des rekursiven Abstiegs	69
1.5	Kontextsensitive Grammatiken	70
2.	Berechenbarkeit	73
2.1	Hypothetische Maschinen	75
2.1.1	Die Turing-Maschine	75
2.1.2	Registermaschinen	82
2.2	Rekursive Funktionen	84
2.2.1	Primitiv rekursive Funktionen	85
2.2.2	μ -rekursive Funktionen	91
2.2.3	Allgemein rekursive Funktionen	94
2.3	Äquivalenz der Berechenbarkeitsbegriffe	98

2.3.1	Äquivalenz von μ -Berechenbarkeit und Turing-Berechenbarkeit	98
2.3.2	Äquivalenz von RM- und Turing-Berechenbarkeit	100
2.3.3	Die Churchsche These	102
2.4	Entscheidbarkeit	103
2.4.1	Nichtberechenbare Funktion	103
2.4.2	Entscheidbare Prädikate	104
2.4.3	Rekursive und rekursiv aufzählbare Mengen	106
3.	Komplexitätstheorie	109
3.1	Komplexitätsmaße	109
3.1.1	Zeitkomplexität	109
3.1.2	Bandkomplexität	111
3.1.3	Zeit- und Bandkomplexität von Problemen	113
3.1.4	Polynomiale und nichtdeterministisch polynomiale Zeitkomplexität	117
3.1.5	Backtracking-Nichtdeterminismus in Programmiersprachen	118
3.2	NP-Vollständigkeit	122
3.2.1	Das Erfüllbarkeitsproblem für Boolesche Ausdrücke	123
3.2.2	Weitere NP-vollständige Probleme	125
3.3	Effiziente Algorithmen für NP-vollständige Probleme	126
3.3.1	Geschicktes Durchsuchen großer Baumstrukturen	128
3.3.2	Alpha/Beta-Suche	130
3.3.3	Dynamisches Programmieren	135
3.3.4	Greedy-Algorithmen	136
4.	Effiziente Algorithmen und Datenstrukturen	139
4.1	Ausgewählte Algorithmen	139
4.1.1	Komplexität von Sortieralgorithmen	140
4.1.2	Wege in Graphen	142
4.2	Bäume	144
4.2.1	Geordnete, orientierte und sortierte Bäume	145
4.2.2	Darstellung von Bäumen durch Felder	146
4.2.3	AVL-Bäume	147
4.2.4	B-Bäume	148
4.3	Effiziente Darstellung von Mengen	148
4.3.1	Die Rechenstruktur der Mengen mit Zugriff über Schlüssel	149
4.3.2	Mengendarstellung durch AVL-Bäume	150
4.3.3	Streuspeicherverfahren	158
5.	Beschreibungstechniken in der Programmierung	163
5.1	Formalismen für die Spezifikation	163
5.1.1	Abstraktion in der Spezifikation	163
5.1.2	Die Spezifikation von abstrakten Rechenstrukturen	165
5.1.3	Spezifikation von Funktionen	168

5.1.4	Spezifikation von Anweisungen	169
5.2	Datenbanken und Informationssysteme	172
5.2.1	Die Entitäts/Relationsbeziehungsmodellierung	172
5.2.2	Entitäts/Relations-Diagramme	176
5.2.3	Charakterisierung von Relationen	176
5.2.4	Zur Verwendung von Datenbanksystemen	177
5.2.5	Das DB-Managementsystem	177
5.2.6	Datenbankabfragen und Ändern der Datenbestände	177
5.3	Logikprogrammierung	178
5.3.1	Eine Problemlösung mit Logikprogramm	179
5.3.2	Die Auswertung von Logikprogrammen	181
5.3.3	Unifikation	184
5.4.	Objektorientierte Programmierung	187
6.	Abschließende Bemerkungen zur Informatik	197
6.1	Anwendungen der Informatik	198
6.2	Informatik und Recht	200
6.3	Soziale Kompetenz der Informatiker	201
6.4	Informatik und Ökonomie	202
6.5	Informatik, Wissenschaftstheorie und Philosophie	202
6.6	Zur Verantwortung des Informatikers	203
	Literaturangaben	205
	Stichwortverzeichnis	211