

Inhaltsverzeichnis

	Vorwort	3
	Inhaltsverzeichnis	5
1	Endliche Automaten mit Ausgabe	9
	1.1 Autokorrektur und Smileys	9
	1.2 Strichcodes	13
	1.3 Ein Strichcode-Erkenner	15
	1.4 Automatentafeln	19
	1.5 Die Definition endlicher Automaten mit Ausgabe	19
	1.6 Endliche Automaten und Programme	22
	1.7 Aufgaben	25
2	Erkennende Automaten	26
	2.1 Suchmaschinen	26
	2.2 Die Definition erkennender Automaten	28
	2.3 Aufgaben	29
3	SIENA: Ein Simulator für endliche Automaten	30
	3.1 Aufgaben des Simulators	30
	3.2 Die erforderlichen Komponenten	31
	3.3 Die Automatenklassen	32
	3.4 Die Implementierung der Automaten	33
	3.5 Die Bearbeitung der Automatentafel	35
	3.6 Die Simulation	38
	3.7 Aufgaben	40
4	Syntaxanalyse mit erkennenden Automaten	41
	4.1 Das Übersetzen von Programmen: Parser	41
	4.2 Die Analyse von Konstantenvereinbarungen	42
	4.3 Die Analyse von Variablenvereinbarungen	45
	4.4 Aufgaben	50
5	LOGO für Arme – Version 1	51
	5.1 Die Turtlegrafik	51
	5.2 Die Syntax von LOGO für Arme	55
	5.3 Der Turtleparser	56
	5.4 Der Turtleinterpreter	60
	5.5 Die Klassenbeziehungen	62
	5.6 Aufgaben	64

6	Gitterautomaten	65
6.1	Lineare Gitterautomaten	65
6.2	Zweidimensionale Gitterautomaten	74
6.3	Aufgaben	80
7	Formale Sprachen	81
7.1	Satzgliederungssprachen	81
7.2	Die Backus-Naur-Form BNF	85
7.3	Linksreguläre Sprachen	87
7.4	Analysierende Grammatiken	90
7.5	Die Äquivalenz von erkennenden Automaten und linksreg. Grammatiken	94
7.6	Die Grenzen regulärer Sprachen	98
7.7	Aufgaben	100
7.8	Ein Funktionsplotter	103
7.9	Aufgaben	112
8	Kellerautomaten	113
8.1	Die Definition von Kellerautomaten	113
8.2	Die Arbeitsweise von Kellerautomaten	116
8.3	Kellerautomaten und Programmiersprachen	119
8.4	LOGO für Arme – Version 2	119
8.5	Aufgaben	124
8.6	Die Compilierung arithmetischer Ausdrücke	125
8.7	Aufgaben	130
9.	Kontextfreie Sprachen	131
9.1	Klammerstrukturen erzeugen	131
9.2	Analyse durch Syntaxbäume	133
9.3	Effiziente Syntaxanalyse	134
9.4	Parsingtabellen	137
9.5	Aufgaben	140
10.	LINDA: Die Lindenmeyersystem-Anzeige	142
10.1	Lindenmeyer-Systeme	143
10.2	Das Regelsystem	145
10.3	Der Stapel	147
10.4	Pflanzen zeichnen	148
10.5	Beispiele	150

11. Turingmaschinen und Algorithmen	151
11.1 Berechenbare Funktionen	151
11.2 Aufgaben	154
11.3 Eindimensionale Turingmaschinen	156
11.4 Aufgaben	160
11.5 Gekoppelte Turingmaschinen	161
11.6 Zweidimensionale Turingmaschinen	165
11.7 Aufgaben	173
12. TuPS: eine Turing-Programmiersprache für gekoppelte Turingmaschinen .	174
12.1 Die Definition der Sprache TuPS	174
12.2 Eine Grammatik für TuPS	178
12.3 Ein Parser für TuPS	179
12.4 Ein Befehlslexer für TuPS	180
12.5 Ein Interpreter für TuPS	183
12.6 Die Oberfläche von TuPS	187
13. Turingberechenbarkeit	190
13.1 Berechenbarkeit	190
13.2 Gekoppelte Turingmaschinen für die elementaren Funktionen	192
13.3 Rekursive Turingmaschinen	194
13.4 Die universelle Turingmaschine	199
13.5 Fleißige Biber und das Halteproblem	200
13.6 Zur Unentscheidbarkeit des Halteproblems	203
13.7 Aufgaben	205
Literaturverzeichnis	206
Stichwortverzeichnis	207