

INHALTSVERZEICHNIS

EINLEITUNG	5
1. MASCHINENMODELL UND KOMPLEXITÄT	10
1.1. Aufbau und Wirkungsweise einer Registermaschine	11
1.2. Komplexitätsmaße für Registermaschinen	18
1.3. Asymptotische Komplexitätsanalyse von Algorithmen	26
1.4. PIDGIN-PASCAL - eine höhere Programmiersprache	31
1.5. Literaturhinweise	40
2. ÜBER DIE PRINZIPIELLE UND PRAKTISCHE LÖSBARKEIT VON PROBLEMEN	41
2.1. Berechenbarkeit	43
2.1.1. RM-berechenbare Funktionen	44
2.1.2. Partiell rekursive Funktionen	45
2.1.3. Churchsches These	53
2.2. Diagonalisierung und Reduktionstechniken	54
2.2.1. Diagonalisierung	55
2.2.2. Reduktion	58
2.3. Praktisch realisierbare und praktisch unrealisierbare Berechnungen	60
2.4. Literaturhinweise	65
3. GRUNDLEGENDE DATENSTRUKTUREN	67
3.1. Felder	67
3.2. Listen	69
3.2.1. Stapel	77
3.2.2. Schlangen	79
3.3. Bäume	81
3.4. Literaturhinweise	87
4. GRUNDLEGENDE ENTWURFSTECHNIKEN FÜR EFFIZIENTE ALGORITHMEN	88
4.1. Die Methode des "Teile-und-Herrsche"	89

4.2.	Rekursive Algorithmen	91
4.2.1.	Analyse rekursiver Algorithmen	92
4.2.1.	Implementierung rekursiver Algorithmen auf Registermaschinen	99
4.3.	Dynamisches Programmieren	103
4.4.	Die Greedy-Methode	107
4.5.	Such- und Durchlauftechniken	110
4.6.	Literaturhinweise	114
5.	EFFIZIENTES SORTIEREN	115
5.1.	Sortieren durch Vergleich	117
5.1.1.	Eine untere Schranke für die Anzahl der Vergleiche	117
5.1.2.	Sortieren durch Mischen: Mergesort	121
5.1.3.	Sortieren durch Auswahl: Heapsort	125
5.1.4.	Sortieren durch Teilen: Quicksort	134
5.1.5.	Kurzer Vergleich der vorgestellten Verfahren	143
5.2.	Sortieren strukturierter Schlüssel	145
5.2.1.	Fachverteilung von Wörtern: Bucketsort	146
5.2.2.	Sortieren reeller Zahlen durch Verteilen: Hybridsort	156
5.2.3.	Ein linearer Sortieralgorithmus große Zahlen	159
5.3.	Literaturhinweise	167
6.	EFFIZIENTES VERWALTEN VON (DATEN-) MENGEN	169
6.1.	Elementare rechnerinterne Darstellung von Mengen	171
6.2.	Verwalten durch Hashing	174
6.3.	Verwalten von Mengen mit binären Suchbäumen	181
6.4.	Balancieren von Suchbäumen	187
6.5.	Optimale Suchbäume	198
6.6.	Union-Find-Algorithmen	207
6.7.	Literaturhinweise	216
7.	EFFIZIENTE ALGORITHMEN AUF GRAPHEN	217
7.1.	Graphen und ihre rechnerinterne Darstellung	218
7.2.	Aufspannende Bäume mit minimalen Kosten	223
7.3.	Durchsuchen und Durchlaufen von Graphen	234
7.3.1.	Methode des "Zuerst in die Tiefe gehen"	236
7.3.2.	Anwendungsbeispiele	243
7.3.3.	Methode des "Zuerst in die Breite gehen"	250

7.4. Ebene Graphen	253
7.4.1. Ausblick: Planaritätstest	255
7.4.2. Separatortheorem für ebene Graphen	255
7.4.2. Maximale Paarungen in ebenen Graphen	260
7.5. Wegprobleme in Graphen	265
7.5.1. Allgemeines Wegproblem	269
7.5.2. Transitiver Abschluß	273
7.5.3. Kurzeste und kostengünstigste Wege	274
7.5.4. Wegprobleme und Matrizenmultiplikation	276
7.6. Literaturhinweise	281
8. EFFIZIENTE ALGEBRAISCHE BERECHNUNGEN	282
8.1. Schnelle Matrizenmultiplikation und ihre Anwendung	283
8.1.1. Schnelle Matrizenmultiplikation nach Strassen	284
8.1.2. Weitere asymptotische Verbesserungen	288
8.1.3. Schnelle Invertierung von Matrizen	289
8.1.4. Schnelle Determinantenberechnung	292
8.1.5. Schnelle Boolesche Matrizenmultiplikation	294
8.2. Auswertung und Interpolation von Polynomen	295
8.2.1. Rechnerinterne Darstellung von Polynomen	296
8.2.2. Auswertung von Polynomen	298
8.2.3. Interpolation	302
8.3. Schnelle Fourier-Transformation und ihre Anwendung	306
8.3.1. (Diskrete) Fourier-Transformation	307
8.3.2. Schnelle Fourier-Transformation	312
8.3.3. Nichtrekursive schnelle Fourier-Transformation	318
8.3.4. Schnelle Auswertung und Interpolation	320
8.3.5. Schnelle Berechnung der Konvolution	321
8.3.6. Schnelle Multiplikation von Polynomen	324
8.3.7. Ausblick: Schnelle Multiplikation von n -Bit Zahlen	325
8.4. Modulare Arithmetik	327
8.4.1. Grundlagen der modularen Arithmetik	328
8.4.2. Effiziente Berechnung der modularen Darstellung	334
8.4.3. Implementation des Chinesischen Restsatzes	339
8.5. Literaturhinweise	344
LITERATURVERZEICHNIS	346
STICHWORTVERZEICHNIS	350