

Inhaltsverzeichnis

I	Grundlegende Konzepte	1
1	Vorbemerkungen und Überblick	3
1.1	Informatik, Algorithmen und Datenstrukturen	3
1.2	Historischer Überblick: Algorithmen	5
1.3	Historie von Programmiersprachen und Java	6
1.4	Grundkonzepte der Programmierung in Java	9
2	Algorithmische Grundkonzepte	17
2.1	Intuitiver Algorithmusbegriff	17
2.1.1	Beispiele für Algorithmen	17
2.1.2	Bausteine für Algorithmen	21
2.1.3	Pseudocode-Notation für Algorithmen	23
2.1.4	Struktogramme	28
2.1.5	Rekursion	29
2.2	Sprachen und Grammatiken	32
2.2.1	Begriffsbildung	33
2.2.2	Reguläre Ausdrücke	34
2.2.3	Backus-Naur-Form (BNF)	35
2.3	Elementare Datentypen	36
2.3.1	Datentypen als Algebren	37
2.3.2	Signaturen von Datentypen	37
2.3.3	Der Datentyp <code>bool</code>	39
2.3.4	Der Datentyp <code>integer</code>	40
2.3.5	Felder und Zeichenketten	41
2.4	Terme	43
2.4.1	Bildung von Termen	43
2.4.2	Algorithmus zur Termauswertung	45
2.5	Datentypen in Java	46
2.5.1	Primitive Datentypen	46
2.5.2	Referenzdatentypen	48
2.5.3	Operatoren	53

3	Algorithmenparadigmen	57
3.1	Überblick über Algorithmenparadigmen	57
3.2	Applikative Algorithmen	58
3.2.1	Terme mit Unbestimmten	58
3.2.2	Funktionsdefinitionen	59
3.2.3	Auswertung von Funktionen	60
3.2.4	Erweiterung der Funktionsdefinition	61
3.2.5	Applikative Algorithmen	62
3.2.6	Beispiele für applikative Algorithmen	63
3.3	Imperative Algorithmen	71
3.3.1	Grundlagen imperativer Algorithmen	71
3.3.2	Komplexe Anweisungen	74
3.3.3	Beispiele für imperative Algorithmen	77
3.4	Das logische Paradigma	82
3.4.1	Logik der Fakten und Regeln	83
3.4.2	Deduktive Algorithmen	84
3.5	Weitere Paradigmen	89
3.5.1	Genetische Algorithmen	90
3.5.2	Neuronale Netze	93
3.6	Umsetzung in Java	97
3.6.1	Ausdrücke und Anweisungen	97
3.6.2	Methoden	106
3.6.3	Applikative Algorithmen und Rekursion	112
4	Literaturhinweise zum Teil I	119

II Algorithmen **121**

5	Ausgewählte Algorithmen	123
5.1	Suchen in sortierten Folgen	123
5.1.1	Sequenzielle Suche	124
5.1.2	Binäre Suche	126
5.2	Sortieren	130
5.2.1	Sortieren: Grundbegriffe	131
5.2.2	Sortieren durch Einfügen	131
5.2.3	Sortieren durch Selektion	134
5.2.4	Sortieren durch Vertauschen: BubbleSort	135
5.2.5	Sortieren durch Mischen: MergeSort	138
5.2.6	QuickSort	141
5.2.7	Sortieren durch Verteilen: RadixSort	145
5.2.8	Sortierverfahren im Vergleich	149

6	Formale Algorithmenmodelle	155
6.1	Registermaschinen	155
6.2	Abstrakte Maschinen	164
6.3	Markov-Algorithmen	168
6.4	Church'sche These	174
6.5	Interpreter für formale Algorithmenmodelle in Java	176
	6.5.1 Java: Markov-Interpreter	176
	6.5.2 Registermaschine in Java	178
7	Eigenschaften von Algorithmen	185
7.1	Berechenbarkeit und Entscheidbarkeit	185
	7.1.1 Existenz nichtberechenbarer Funktionen	186
	7.1.2 Konkrete nichtberechenbare Funktionen	188
	7.1.3 Das Halteproblem	190
	7.1.4 Nichtentscheidbare Probleme	192
	7.1.5 Post'sches Korrespondenzproblem	193
7.2	Korrektheit von Algorithmen	195
	7.2.1 Relative Korrektheit	196
	7.2.2 Korrektheit von imperativen Algorithmen	196
	7.2.3 Korrektheitsbeweise für Anweisungstypen	199
	7.2.4 Korrektheit imperativer Algorithmen an Beispielen	202
	7.2.5 Korrektheit applikativer Algorithmen	207
7.3	Komplexität	208
	7.3.1 Motivierendes Beispiel	208
	7.3.2 Asymptotische Analyse	210
	7.3.3 Komplexitätsklassen	214
	7.3.4 Analyse von Algorithmen	216
8	Entwurf von Algorithmen	219
8.1	Entwurfsprinzipien	219
	8.1.1 Schrittweise Verfeinerung	219
	8.1.2 Einsatz von Algorithmenmustern	224
	8.1.3 Problemreduzierung durch Rekursion	225
8.2	Algorithmenmuster: Greedy	226
	8.2.1 Greedy-Algorithmen am Beispiel	226
	8.2.2 Greedy: Optimales Kommunikationsnetz	227
	8.2.3 Verfeinerung der Suche nach billigster Kante	228
8.3	Rekursion: Divide-and-conquer	231
	8.3.1 Das Prinzip »Teile und herrsche«	231
	8.3.2 Beispiel: Spielpläne für Turniere	232
8.4	Rekursion: Backtracking	234
	8.4.1 Prinzip des Backtracking	235
	8.4.2 Beispiel: Das Acht-Damen-Problem	237

8.4.3	Beispiel: Tic Tac Toe mit Backtracking	240
8.5	Dynamische Programmierung	242
8.5.1	Das Rucksackproblem	243
8.5.2	Rekursive Lösung des Rucksackproblems	245
8.5.3	Prinzip der dynamischen Programmierung	246
9	Parallele und verteilte Berechnungen	249
9.1	Grundlagen	249
9.2	Modell der Petri-Netze	252
9.2.1	Definition von Petri-Netzen	252
9.2.2	Formalisierung von Petri-Netzen	256
9.2.3	Das Beispiel der fünf Philosophen	258
9.3	Programmieren nebenläufiger Abläufe	260
9.3.1	Koordinierte Prozesse	261
9.3.2	Programmieren mit Semaphoren	262
9.3.3	Philosophenproblem mit Semaphoren	264
9.3.4	Verklemmungsfreie Philosophen	265
9.4	Nebenläufige Berechnungen in Java	268
9.4.1	Threads und wechselseitiger Ausschluss	268
9.4.2	Parallelisierung in Java	271
9.4.3	Das Philosophenproblem in Java	274
10	Literaturhinweise zum Teil II	281
III	Datenstrukturen	283
11	Abstrakte Datentypen	285
11.1	Signaturen und Algebren	286
11.2	Algebraische Spezifikation	288
11.2.1	Spezifikationen und Modelle	289
11.2.2	Termalgebra und Quotiententermalgebra	290
11.2.3	Probleme mit initialer Semantik	293
11.3	Beispiele für abstrakte Datentypen	294
11.3.1	Der Kellerspeicher (Stack)	295
11.3.2	Beispiel für Kellernutzung	297
11.3.3	Die Warteschlange (Queue)	301
11.4	Entwurf von Datentypen	303
12	Klassen, Schnittstellen und Objekte in Java	305
12.1	Grundzüge der Objektorientierung	305
12.2	Klassen und Objekte in Java	308
12.3	Vererbung	313
12.4	Abstrakte Klassen und Schnittstellen	320

12.5	Ausnahmen	323
12.6	Umsetzung abstrakter Datentypen	326
12.7	Lambda-Ausdrücke	329
13	Grundlegende Datenstrukturen	335
13.1	Stack und Queue als Datentypen	335
13.1.1	Implementierung des Stacks	339
13.1.2	Implementierung der Queue	340
13.1.3	Bewertung der Implementierungen	342
13.2	Verkettete Listen	343
13.3	Doppelt verkettete Listen	350
13.4	Skip-Listen	355
13.5	Das Iterator-Konzept	358
13.6	Java Collection Framework	361
13.7	Generics in Java	365
14	Bäume	369
14.1	Bäume: Begriffe und Konzepte	369
14.2	Binärer Baum: Datentyp und Basisalgorithmen	372
14.2.1	Der Datentyp »Binärer Baum«	372
14.2.2	Algorithmen zur Traversierung	377
14.3	Suchbäume	382
14.3.1	Suchen in Suchbäumen	383
14.3.2	Einfügen und Löschen	386
14.3.3	Komplexität der Operationen	391
14.4	Ausgeglichene Bäume	392
14.4.1	Rot-Schwarz-Bäume	393
14.4.2	AVL-Bäume	402
14.4.3	B-Bäume	410
14.5	Digitale Bäume	423
14.5.1	Tries	424
14.5.2	Patricia-Bäume	429
14.6	Praktische Nutzung von Bäumen	431
14.6.1	Sortieren mit Bäumen: HeapSort	431
14.6.2	Sets mit binären Suchbäumen	438
15	Hashverfahren	443
15.1	Grundprinzip des Hashens	443
15.2	Grundlagen und Verfahren	444
15.2.1	Hashfunktionen	445
15.2.2	Behandlung von Kollisionen	446
15.2.3	Aufwand beim Hashen	450
15.2.4	Hashen in Java	452

15.2.5	Cuckoo-Hashing	456
15.3	Dynamische Hashverfahren	459
15.3.1	Grundideen für dynamische Hashverfahren	460
15.3.2	Erweiterbares Hashen	463
15.3.3	Umsetzung des erweiterbaren Hashens	465
16	Graphen	471
16.1	Arten von Graphen	471
16.1.1	Ungerichtete Graphen	472
16.1.2	Gerichtete Graphen	473
16.1.3	Gewichtete Graphen	474
16.1.4	Weitere Eigenschaften von Graphen	475
16.2	Realisierung von Graphen	475
16.2.1	Knoten- und Kantenlisten	475
16.2.2	Adjazenzmatrix	476
16.2.3	Graphen als dynamische Datenstrukturen	477
16.2.4	Transformationen zwischen Darstellungen	477
16.2.5	Vergleich der Komplexität	478
16.2.6	Eine Java-Klasse für Graphen	479
16.3	Ausgewählte Graphenalgorithmen	481
16.3.1	Breitendurchlauf	481
16.3.2	Tiefendurchlauf	486
16.3.3	Zyklenfreiheit und topologisches Sortieren	490
16.4	Algorithmen auf gewichteten Graphen	493
16.4.1	Kürzeste Wege	493
16.4.2	Dijkstras Algorithmus	494
16.4.3	A*-Algorithmus	498
16.4.4	Kürzeste Wege mit negativen Kantengewichten	504
16.4.5	Maximaler Durchfluss	508
16.4.6	Der Ford-Fulkerson-Algorithmus	510
16.5	Zentralitätsanalyse in Graphen	513
16.6	Weitere Fragestellungen für Graphen	518
17	Algorithmen auf Texten	523
17.1	Probleme der Worterkennung	523
17.2	Knuth-Morris-Pratt	525
17.3	Boyer-Moore	529
17.4	Pattern Matching	535
17.4.1	Reguläre Ausdrücke	536
17.4.2	Endliche Automaten	537
17.4.3	Java-Klassen für reguläre Ausdrücke	543
17.5	Ähnlichkeit von Zeichenketten	544
17.5.1	Levenshtein-Distanz	544

17.5.2	n-Gramme	547
17.5.3	Anwendungen der Ähnlichkeitsvergleiche	549
18	Literaturhinweise zum Teil III	551
IV	Anhang	553
A	Quelltext der Klasse IOUtils	555
	Abbildungsverzeichnis	559
	Tabellenverzeichnis	565
	Algorithmenverzeichnis	567
	Beispielverzeichnis	569
	Programmverzeichnis	571
	Literaturverzeichnis	575
	Index	580