

Andreas Solymosi • Ulrich Grude

# **Grundkurs Algorithmen und Datenstrukturen in JAVA**

Eine Einführung in die  
praktische Informatik

**5., aktualisierte Auflage**

**Springer Vieweg**

# Inhaltsverzeichnis

Vorwort zur 4. und 5. Auflage	V
Einleitung	XI
Danksagungen	XIII
<b>1. Begriffsbildung</b>	<b>1</b>
1.1. Algorithmus	1
1.2. Komplexität	4
1.3. Verbrauch und Komplexität	5
<b>2. Gleichwertige Lösungen</b>	<b>8</b>
2.1. Maximale Teilsumme	8
2.1.1. Summen und Teilsummen	8
2.1.2. Aufgabenstellung	9
2.1.3. Intuitive Lösung	9
2.1.4. Zeitkomplexität der Lösung	10
2.1.5. Zeit für Raum	12
2.1.6. Teile und herrsche	13
2.1.7. Die optimale Lösung	16
2.1.8. Messergebnisse	18
2.1.9. Gleichwertigkeit von Algorithmen	19
2.2. Komplexitätsformel	20
2.3. Datenstrukturen	21
2.3.1. Reihungen	22
2.3.2. Verkettete Listen	23
2.3.3. Gleichwertigkeit von Datenstrukturen	26
2.3.4. Berechnung von Ausdrücken	29
<b>3. Rekursion und Wiederholung</b>	<b>30</b>
3.1. Rekursive Algorithmen	30
3.1.1. Fakultät	30
3.1.2. Die Fibonacci-Zahlen	31
3.1.3. Die Ackermann-Funktion	33
3.1.4. Die mathematische Induktion	34
3.1.5. Permutationen	37

3.2. Abarbeitung von Datenstrukturen	37
3.2.1. Iterative Abarbeitung von rekursiven Datenstrukturen	38
3.2.2. Rekursive Abarbeitung von rekursiven Datenstrukturen	39
3.2.3. Rekursive Abarbeitung von Reihungen	40
3.2.4. Iteratoren	41
3.3. Rekursive Kurven	42
3.3.1. Schneeflockenkurve	43
3.3.2. Die Pfeilspitzenkurve	45
3.3.3. Die Hilbert-Kurve	47
3.3.4. Ersetzen der Rekursion durch Wiederholung	50
3.4. Zurückverfolgung	52
3.4.1. Labyrinth	52
3.4.2. Der Weg des Springers	53
3.4.3. Die acht Damen	55
<b>4. Suchen</b>	<b>59</b>
4.1. Textsuche	59
4.2. Suchen in Sammlungen	63
4.3. Suchen in einer Reihung	64
4.3.1. Suchen in einer unsortierten Reihung	64
4.3.2. Lineares Suchen in einer sortierten Reihung	66
4.3.3. Binäres Suchen in einer sortierten Reihung	67
4.4. Suchen in einer verketteten Liste	68
4.4.1. Lineares Suchen in einer unsortierten Liste	69
4.4.2. Lineares Suchen in einer sortierten Liste	70
4.5. Hash-Tabellen	70
4.5.1. Funktionalität	71
4.5.2. Datenorganisation	71
4.5.3. Hash-Funktionen	75
4.5.4. Weitere Aspekte	79
4.6. Zeitkomplexitäten beim Suchen	79
<b>5. Sortierverfahren</b>	<b>82</b>
5.1. Die Problemstellung	82
5.1.1. Präzisierung des Problems und Grundbegriffe	83
5.1.2. Zeitbedarf und Zeitkomplexität	84
5.1.3. Sortieralgorithmen in Java-Standardbibliotheken	85
5.1.4. Entwurfsmuster Strategie	86

Inhaltsverzeichnis	IX
5.2. Quadratische Sortierverfahren	87
5.2.1. Sortieren durch Vertauschen benachbarter Elemente	88
5.2.2. Sortieren durch Einfügen	90
5.2.3. Sortieren durch Auswählen	92
5.3. Unterquadratische Verfahren	93
5.4. Rekursive Verfahren	95
5.4.1. Quicksort	95
5.4.2. Sortieren mit Mischen	98
5.5. Logarithmische Verfahren	98
5.5.1. Halde	99
5.5.2. Die Haldenbedingung	100
5.5.3. Senken	100
5.5.4. Zwei Phasen des Heap Sorts	101
5.5.5. Sortieren auf der Halde	102
5.6. Externe Sortierverfahren	104
5.6.1. Mischen	104
5.6.2. Sortierkanal	106
5.6.3. Mischkanal	108
5.6.4. Fibonacci-Mischen	108
<b>6. Baumstrukturen</b>	<b>111</b>
6.1. Binärbaum	111
6.1.1. Definition	111
6.1.2. Suchen im sortierten Binärbaum	114
6.1.3. Darstellung von Binärbäumen	115
6.2. Sortieren mit Binärbäumen	116
6.2.1. Binärbaum als Halde	117
6.2.2. Senken im Binärbaum	118
6.2.3. Baumsort	120
6.2.4. Durchwandern eines Binärbaums	121
6.3. Operationen für Binärbäume	123
6.3.1. Binärbaum aus Knoten	123
6.3.2. Eintragen in einen sortierten Binärbaum	124
6.3.3. Löschen in Binärbäumen	125
6.4. Ausgeglichene Bäume	128
6.4.1. Eintragen in ausgeglichene Bäume	129
6.4.2. Löschen in ausgeglichenen Bäumen	133
6.5. 2-3-4-Bäume	135

6.5.1. Definition	135
6.5.2. Spalten	136
6.5.3. Einfügen	138
6.6. Rot-Schwarz-Bäume	139
6.7. B-Bäume	146
<b>7. Klassen von Algorithmen</b>	<b>149</b>
7.1. Was ist ein algorithmisches Problem?	149
7.2. Theoretische Lösbarkeit von Problemen	154
7.2.1. Definitionen	154
7.2.2. Beispiele	154
7.2.3. Das Halteproblem	157
7.2.4. Das Kachelproblem	159
7.2.5. Das Paligrammproblem	161
7.2.6. Gleichwertigkeit von Grammatiken	162
7.3. Praktische Lösbarkeit von Problemen	163
7.3.1. Das zweite Kachelproblem	164
7.3.2. Das Rucksackproblem	164
7.3.3. Das Aufteilungsproblem	165
7.3.4. Das Problem des Handelsreisenden	165
7.3.5. Hamiltonsche Wege durch einen Graphen	166
7.3.6. Das Erfüllbarkeitsproblem	167
7.4. Die Klassen <b>P</b> und <b>NP</b>	168
7.5. Ist <b>P = NP</b> ?	169
7.6. Übersicht über Problemklassen	171
Literaturverzeichnis	172
Empfehlungen	173
Programmverzeichnis	175
Abbildungs- und Tabellenverzeichnis	177
Sachwortverzeichnis	180