

Inhaltsverzeichnis

1. Einführung und Überblick	1
1.1 Bedeutung und Grundprobleme der Informatik	1
1.1.1 Die Bedeutung des Berechnens von Funktionen	5
1.1.2 Das Problem der Komplexität	7
1.2 Konzeption des Buches	8
1.2.1 Aufbau des Buches	9
1.2.2 Hinweise für Dozenten	11

Teil I. Grundkonzepte von Hardware und Software

2. Aufbau und Funktionsweise eines Computers	15
2.1 Einleitung und Überblick	15
2.2 Der Kern des Rechners: von Neumann-Architektur	17
2.2.1 Speicher	18
2.2.2 Prozessor und Programmausführung	20
2.3 System-Architektur der Hardware	22
2.4 System-Architektur der Software	26
2.4.1 Schichtenaufbau	26
2.4.2 Das Betriebssystem	30
2.4.3 Java und die Virtuelle Java-Maschine JVM	31
2.5 Binärcodierung elementarer Datentypen	32
2.5.1 Ganze Zahlen (Dualzahlen)	33
2.5.2 Hexadezimalzahlen und Oktalzahlen	39
2.5.3 Zeichen (ASCII und Unicode)	39
2.5.4 Gleitkommazahlen (IEEE 754)	41
2.6 Übungen	45
3. Abstrakte Algorithmen und Sprachkonzepte	47
3.1 Einleitung und Begriffsdefinition	47
3.2 Aufbau und Beschreibung von Algorithmen	51
3.2.1 Textuelle Beschreibung in Schritten	51

3.2.2	Graphische Beschreibung mit UML (Flußdiagramme)	53
3.2.3	Grundschema des Algorithmenaufbaus	54
3.2.4	Strukturiert-iterative Beschreibungen	57
3.2.5	Rekursive Beschreibung in mathematischer Notation	59
3.2.6	Beschreibung mit Pseudo-Code	59
3.3	Programmiersprachliche Grundkonzepte	59
3.3.1	Das Sprung-Konzept <code>goto</code>	61
3.3.2	Verzweigung mit der bedingten Anweisung <code>if-then-else</code>	61
3.3.3	Rekursion	63
3.3.4	Die <code>while</code> -Schleife	64
3.3.5	Die <code>repeat-until</code> -Schleife	65
3.3.6	Die <code>for</code> -Schleife	66
3.4	Konstruktion und Verifikation rekursiver Algorithmen	66
3.4.1	Der rekursive Ansatz zur Problemlösung	66
3.4.2	Ein rekursives Verfahren in mathematischer Notation	67
3.4.3	Ein rekursives Verfahren in Java	68
3.5	Konstruktion und Verifikation iterativer Algorithmen	70
3.5.1	Der iterative Ansatz zur Problemlösung	70
3.5.2	Die Verifikation nach Floyd	73
3.5.3	Ein strukturiert-iteratives Verfahren in Java	76
3.6	Übungen	77
4.	Konzepte benutzerdefinierter Datenstrukturen	79
4.1	Einleitung	79
4.2	Reihungen (<i>arrays</i>)	80
4.3	Verbunde (<i>records, structs</i>)	81
4.4	Typ-Kombinationen von Reihung und Verbund	83
4.5	Modellierung des Enthaltenseins – Referenzen	83
4.6	Klassen, Objekte, abstrakte Datentypen	86
5.	Objektorientierte Software-Konzepte und UML	89
5.1	Objektorientierte Software-Entwicklung	89
5.2	Objekte, Klassen, abstrakte Datentypen	93
5.3	Objektbeziehungen	97
5.3.1	Informationsfluß- und Client/Server-Beziehungen	99
5.3.2	Einschlußbeziehungen (<i>has-a</i>)	102
5.3.3	Subtyp- bzw. Vererbungsbeziehungen (<i>is-a</i>)	104
5.4	Objektorientierte Analyse und Entwurf	106
5.4.1	Analyse einer Werkstück-Vereinzelungseinheit	106
5.5	Entwurfsmuster	110
5.5.1	Beispiel: Architekturmuster einer Gerätefernsteuerung	111
5.6	Übungen	115

Teil II. Sprachkonzepte und ihre Verwirklichung in Java

6. Elementare Konzepte von Programmiersprachen	119
6.1 Einleitung und Überblick	119
6.2 Programmentwicklung in Java	121
6.2.1 Entwicklungsumgebungen für Java	122
6.2.2 Ein Rahmenprogramm für Java-Anweisungen	123
6.2.3 Ein Rahmenprogramm für Java-Funktionen	124
6.2.4 Übersetzung und Ausführung von Java-Programmen	125
6.3 Schlüsselwörter, Literale und Namen	127
6.4 Elementare Datentypen	129
6.5 Variablen, Referenzen, Zuweisungen	131
6.5.1 Grundkonzepte	131
6.5.2 Referenzvariablen	134
6.5.3 Reihungsvariablen	135
6.6 Java-Arithmetik	137
6.6.1 Elementare Zahltypen	137
6.6.2 Ganzzahl-Arithmetik	138
6.6.3 Gleitkomma-Arithmetik	139
6.7 Operatoren und Ausdrücke	141
6.7.1 Zuweisungsoperatoren	141
6.7.2 Arithmetische Operatoren	141
6.7.3 Boolesche Operatoren	142
6.7.4 Bitmuster	143
6.7.5 Ausdrücke	144
6.7.6 Syntax von Ausdrücken	145
6.7.7 Präzedenz von Operatoren	145
6.7.8 Semantik von Ausdrücken	147
6.7.9 Bedingte Ausdrücke	150
6.7.10 Typkonversionen	151
6.8 Anweisungen	154
6.8.1 Blöcke, Gültigkeitsbereich und Lebensdauer	155
6.8.2 Bedingte Anweisungen (if und switch)	159
6.8.3 Schleifenkonstrukte (while, do-while, for)	161
6.8.4 Marken, break und continue	165
6.9 Unterprogramme – Prozeduren und Funktionen	168
6.9.1 Konzepte und Terminologie	168
6.9.2 Unterprogramme in Java	173
6.9.3 Parameterübergabe und Laufzeitstapel	174
6.9.4 Spezifikation von Unterprogrammen	183

6.9.5	Rekursion	187
6.9.6	Allgemeine Rekursion und Speicherverwaltung	190
6.10	Übungen	193
7.	Klassen und höhere Datentypen	199
7.1	Einleitung und Überblick	199
7.2	Objekte, Felder und Methoden	201
7.2.1	Überladen von Methoden	204
7.2.2	Klassenvariablen und Klassenmethoden	204
7.2.3	Pakete (<i>packages</i>)	205
7.2.4	Kapselung und Zugriffskontrolle	206
7.2.5	Kontrakt und Aufrufchnittstelle	207
7.2.6	Verwaltung von Objekten im Speicher	208
7.2.7	Initialisierung und Konstruktoren	212
7.2.8	Selektoren	215
7.2.9	Beispiel eines Datentyps: komplexe Zahlen	216
7.3	Objekte für Ausnahmen (<i>exceptions</i>)	218
7.3.1	Einleitung und Überblick	218
7.3.2	Ausnahmeklassen	220
7.3.3	Die <code>throw</code> -Anweisung	222
7.3.4	Der Rahmen <code>try-catch-finally</code>	223
7.3.5	Deklaration von Ausnahmen mit <code>throws</code>	224
7.4	Wahrheitsbehauptungen und Zusicherungen (<i>assertions</i>)	226
7.5	Reihungen (<i>arrays</i>)	230
7.5.1	Allgemeine Konzepte, Terminologie und Realisierung	230
7.5.2	Eindimensionale Reihungen	232
7.5.3	Skalar- und Vektor-Operationen	234
7.5.4	Mehrdimensionale Reihungen und Matrizen	238
7.6	Zeichenketten (<i>strings</i>)	241
7.6.1	Veränderliche Zeichenketten	243
7.7	Listen (<i>linked lists</i>)	243
7.7.1	Konzepte, Terminologie und Entwurf	244
7.7.2	Die Implementierung von Listen	246
7.7.3	Einfügen eines Elementes	248
7.7.4	Sortiertes Einfügen eines Elements	249
7.7.5	Invertieren einer Liste	252
7.7.6	Doppelt verkettete Listen (<i>doubly linked lists</i>)	253
7.8	Stapel (<i>stacks</i>)	257
7.8.1	Konzept und Terminologie	257
7.8.2	Implementierung von Stacks	258
7.9	Warteschlangen (<i>queues</i>)	259

7.9.1	Konzept und Terminologie	259
7.9.2	Implementierung von Queues	260
7.10	Übungen	261
8.	Höhere objektorientierte Konzepte	263
8.1	Einleitung	263
8.2	Vererbung und abgeleitete Klassen	265
8.2.1	Der Zugriffsschutz <code>protected</code> in Klassenhierarchien	267
8.2.2	Konstruktoren in Klassen-Hierarchien	268
8.3	Virtuelle Funktionen und dynamisches Binden	270
8.3.1	Konzepte und Terminologie	270
8.3.2	Realisierung des dynamischen Bindens	273
8.3.3	Klassenkontrakte und virtuelle Funktionen	274
8.3.4	Typanpassungen in Klassenhierarchien	274
8.3.5	Zugriffsregeln und Auswahlregeln in Klassenhierarchien – Überschreiben und Verdecken	275
8.4	Abstrakte Klassen und Interfaces	279
8.4.1	Abstrakte Klassen	279
8.4.2	Schnittstellen (<i>interfaces</i>)	280
8.5	Mehrfachvererbung	281
8.6	Generisches Programmieren	282
8.6.1	Generische Datentypen	282
8.6.2	Generische Methoden	284
8.6.3	Explizite Typkonversion	285
8.6.4	Klassen-Muster (<i>template classes</i>) und generisches Java	286
8.6.5	Generische Funktionsparameter	289
8.7	Übungen	294
9.	Das „Abstract Window Toolkit“ (AWT)	299
9.1	Graphische Komponenten	300
9.1.1	Klassenhierarchie der graphischen Komponenten	300
9.1.2	Funktionalität von <code>Component</code>	300
9.1.3	Die Klasse <code>Graphics</code>	301
9.1.4	Frames	302
9.1.5	Applets	304
9.1.6	Container	305
9.2	Ereignisse (<i>events</i>)	308
9.2.1	AWT-Events	308
9.2.2	Ereignisquellen und Ereignisempfänger	308
9.2.3	Adapter-Klassen	310
9.3	Ein Beispiel: Ein Rahmen zum Zeichnen reeller Funktionen	310
9.4	Ein größeres Beispiel: Darstellung einer Winterlandschaft	314

9.4.1	Anforderungsanalyse	315
9.4.2	Objektorientierte Analyse und Design	315
9.4.3	Implementierung der Klassen	316

Teil III. Algorithmen und weiterführende Datenstrukturen

10.	Theorie der Algorithmenkonstruktion	327
10.1	Einleitung und Überblick	327
10.1.1	Motivation und Begriffsdefinition	327
10.1.2	Notation	329
10.2	Problemspezifikation und Korrektheitsbeweise	330
10.2.1	Spezifikation	330
10.2.2	Partielle Korrektheit	332
10.2.3	Terminierung	332
10.2.4	Beispiel: Berechnung der Quadratwurzel	334
10.3	Schemata für den Algorithmenentwurf	336
10.4	Aufwand und asymptotische Komplexität	339
10.4.1	Exakte Bestimmung der Komplexität	341
10.4.2	Asymptotische Notation	343
11.	Such-Algorithmen	349
11.1	Einleitung und Problemstellung	349
11.2	Lineare Suche	349
11.2.1	Suche mit Wächter	350
11.2.2	Komplexität der linearen Suche	351
11.3	Divide-and-Conquer-Suche	352
11.3.1	Komplexität der binären Suche	353
11.4	Kombinationsverfahren	354
11.4.1	Analyse und Design von Kombinationsverfahren	355
12.	Sortier-Algorithmen	357
12.1	Einleitung und Problemstellung	357
12.2	Greedy-Sortieren	358
12.2.1	Sortieren durch Auswahl	358
12.2.2	Sortieren durch Einfügen	360
12.2.3	Sortieren durch Austauschen	361
12.3	Divide-and-Conquer-Sortieren	362
12.3.1	Quicksort	363
12.3.2	Sortieren durch Mischen	366
12.4	Übungen	369

13. Bäume	371
13.1 Einleitung und Terminologie	371
13.2 Graphen und Bäume	372
13.2.1 Gerichtete Graphen	372
13.2.2 Ungerichtete Graphen	373
13.2.3 Bäume als ungerichtete Graphen	374
13.3 Eigenschaften von Bäumen	374
13.4 Implementierung von Bäumen	376
13.5 Baumdurchläufe	377
13.5.1 Aktionsobjekte für generische Baumdurchläufe	377
13.5.2 Präorder-Sequenz	379
13.5.3 Postorder-Sequenz	381
13.5.4 Inorder-Sequenz	382
13.5.5 Levelorder-Sequenz	384
13.5.6 Optimierung der Baumdurchläufe	385
13.6 Übungen	387
14. Hashing	389
14.1 Einleitung	389
14.2 Hash-Funktionen	390
14.3 Kollisionsbehandlung	392
14.3.1 Separates Ketten	392
14.3.2 Offenes Adressieren	393
14.4 Hash-Tabellen in Java	393
14.5 Übungen	397

Teil IV. Theoretische Grundlagen

15. Mathematische Grundlagen	401
15.1 Einleitung	401
15.2 Mengen	402
15.3 Relationen	404
15.3.1 Binäre Relationen	404
15.3.2 Äquivalenzrelationen	405
15.4 Funktionen	406
15.4.1 Partielle Funktionen	406
15.4.2 Totale Funktionen	406
15.4.3 Definitions- und Bildbereich von Funktionen	406
15.4.4 Eigenschaften von Funktionen	407
15.4.5 Charakteristische Funktionen	408
15.5 Ordnungen	409

15.5.1	Partielle und totale Ordnungen	409
15.5.2	Lexikographische Ordnung	410
15.5.3	Multiset-Ordnungen	410
15.6	Das Prinzip der vollständigen Induktion	411
15.7	Übungen	412
16.	Einführung in die Logik	415
16.1	Einleitung	415
16.2	Die Algebra der Booleschen Wahrheitswerte	416
16.3	Aussagenlogik (<i>PROP</i>)	417
16.3.1	Die Syntax der Aussagenlogik	417
16.3.2	Semantik der Aussagenlogik	418
16.4	Prädikatenlogik erster Stufe (<i>FOPL</i>)	421
16.4.1	Syntax von <i>FOPL</i>	421
16.4.2	Semantik von <i>FOPL</i>	422
16.5	Beweise	424
16.5.1	Logische Äquivalenzen	424
16.5.2	Ableitungen und Logik-Kalküle	425
16.5.3	Beweisbäume	426
16.6	Übungen	427
17.	Korrektheit von Unterprogrammen	429
17.1	Terminologie und Problemstellung	429
17.2	Der Hoare-Kalkül	431
17.2.1	Regeln des Hoare-Kalküls	432
17.2.2	Konsequenzregeln	434
17.2.3	Zuweisungsaxiom	434
17.2.4	Sequenzregel	437
17.2.5	Alternativregeln	439
17.2.6	Iterationsregel	441
17.3	Übungen	444
	Literaturverzeichnis	445
	Index	449