

Inhalt

1 Einführung	1
1.1 Was ist „Informatik“?	1
1.1.1 Technische Informatik	1
1.1.2 Praktische Informatik	2
1.1.3 Theoretische Informatik	2
1.1.4 Angewandte Informatik	3
1.2 Information und Daten	4
1.2.1 Bits	5
1.2.2 Bitfolgen	6
1.2.3 Hexziffern	7
1.2.4 Bytes und Worte	8
1.2.5 Dateien	8
1.2.6 Datei- und Speichergrößen	9
1.2.7 Längen- und Zeiteinheiten	10
1.3 Informationsdarstellung	11
1.3.1 Text	11
1.3.2 ASCII-Code	11
1.3.3 ASCII-Erweiterungen	12
1.3.4 Unicode, UCS und UTF-8	13
1.3.5 Zeichenketten	15
1.3.6 Logische Werte und logische Verknüpfungen	16
1.3.7 Programme	16
1.3.8 Bilder und Musikstücke	17
1.4 Zahlendarstellungen	17
1.4.1 Binärdarstellung	18
1.4.2 Das Oktalsystem und das Hexadezimalsystem	18
1.4.3 Umwandlung in das Dezimalsystem	20
1.4.4 Umwandlung in das Binär-, Oktal- oder Hexadezimalsystem	20
1.4.5 Arithmetische Operationen	22
1.4.6 Darstellung ganzer Zahlen	23
1.4.7 Die Zweierkomplementdarstellung	24
1.4.8 Standardformate ganzer Zahlen	27

1.4.9	Gleitpunktzahlen: Reelle Zahlen	28
1.4.10	Binäre Gleitkommazahlen	30
1.4.11	Real-Zahlenbereiche in Programmiersprachen	33
1.4.12	Daten – Informationen	33
1.4.13	Informationsverarbeitung – Datenverarbeitung	34
1.5	Hardware	35
1.5.1	PCs, Workstations, Mainframes, Super-Computer	35
1.5.2	Aufbau von Computersystemen	37
1.5.3	Der Rechner von außen	38
1.5.4	Das Innenleben	38
1.5.5	Motherboard, Controller und Busse	40
1.5.6	Ein konkretes Motherboard	42
1.5.7	Die Aufgabe der CPU	44
1.5.8	Programmausführung	45
1.5.9	Die Organisation des Hauptspeichers	46
1.5.10	Speichermedien	49
1.5.11	Magnetplatten	50
1.5.12	Festplattenlaufwerke	51
1.5.13	Optische Laufwerke	53
1.5.14	Flash-Speicher	55
1.5.15	Vergleich von Speichermedien	56
1.5.16	Bildschirme	56
1.5.17	Text- und Grafikmodus	57
1.6	Von der Hardware zum Betriebssystem	58
1.6.1	Schnittstellen und Treiber	60
1.6.2	BIOS	61
1.6.3	Die Aufgaben des Betriebssystems	62
1.6.4	Prozess- und Speicherverwaltung	63
1.6.5	Dateiverwaltung	63
1.6.6	DOS, Windows und Linux	66
1.6.7	Bediensysteme	67
1.7	Anwendungsprogramme	70
1.7.1	Textverarbeitung	70
1.7.2	Zeichen und Schriftarten	70
1.7.3	Formatierung	71
1.7.4	Desktop Publishing	73
1.7.5	Textbeschreibungssprachen	73
1.7.6	Tabellenkalkulation: spread sheets	77
1.7.7	Vom Fenster zur Welt zur virtuellen Welt	79
2	Grundlagen der Programmierung	81
2.1	Programmiersprachen	82
2.1.1	Vom Programm zur Maschine	82
2.1.2	Virtuelle Maschinen	83

2.1.3	Interpreter	85
2.1.4	Programmieren und Testen	85
2.1.5	Programmierungsumgebungen	86
2.1.6	Pascal	87
2.1.7	Java	88
2.2	Spezifikationen, Algorithmen, Programme	88
2.2.1	Spezifikationen	89
2.2.2	Vorbedingung – Nachbedingung	90
2.2.3	Algorithmen	91
2.3	Flussdiagramme	91
2.3.1	Kontrollstrukturen	94
2.3.2	Algorithmen als Lösung von Spezifikationen	95
2.3.3	Terminierung	96
2.3.4	Elementare Aktionen	97
2.3.5	Zuweisungen	97
2.3.6	Vom Algorithmus zum Programm	98
2.3.7	Ressourcen	101
2.4	Daten und Datenstrukturen	102
2.4.1	Der Begriff der Datenstruktur	102
2.4.2	Boolesche Werte	103
2.4.3	Zahlen	105
2.4.4	Natürliche Zahlen	105
2.4.5	Der Datentyp Integer	107
2.4.6	Rationale Zahlen	109
2.4.7	Reelle Zahlen	109
2.4.8	Mehrsortige Datenstrukturen	110
2.4.9	Prädikate	111
2.4.10	Konversionen	112
2.4.11	Zeichen	112
2.4.12	Zusammengesetzte Datentypen – Strings	114
2.4.13	Benutzerdefinierte Datenstrukturen	116
2.4.14	Informationsverarbeitung und Datenverarbeitung	117
2.5	Speicher, Variablen und Ausdrücke	118
2.5.1	Deklarationen	119
2.5.2	Initialisierung	120
2.5.3	Kontexte	121
2.5.4	Ausdrücke, Terme	121
2.5.5	Auswertung von Ausdrücken	124
2.5.6	Funktionsdefinitionen	125
2.5.7	Typfehler	127
2.5.8	Seiteneffekte	128
2.6	Der Kern imperativer Sprachen	128
2.6.1	Zuweisungen	129

2.6.2	Kontrollstrukturen	130
2.6.3	Drei Kontrollstrukturen genügen	130
2.6.4	Die sequentielle Komposition	131
2.6.5	Die Alternativanweisung	132
2.6.6	Die while-Schleife	133
2.6.7	Unterprogramme	134
2.6.8	Lauffähige Programme	136
2.7	Formale Beschreibung von Programmiersprachen	137
2.7.1	Lexikalische Regeln	137
2.7.2	Syntaktische Regeln	138
2.7.3	Semantische Regeln	141
2.8	Erweiterung der Kernsprache	141
2.8.1	Bedingte Anweisung	142
2.8.2	Fallunterscheidung	143
2.8.3	do-Schleife	144
2.8.4	Allgemeinere Schleifenkonstrukte	146
2.8.5	Die for-Schleife	146
2.8.6	Arrays – indizierte Variablen	148
2.9	Rekursive Funktionen und Prozeduren	149
2.9.1	Rekursive Programme	151
2.9.2	Die Türme von Hanoi	152
2.9.3	Spielstrategien als rekursive Prädikate – Backtracking	153
2.9.4	Wechselseitige Rekursion	154
2.9.5	Induktion – Rekursion	155
2.9.6	Allgemeine Rekursion	156
2.9.7	Endrekursion	157
2.9.8	Lineare Rekursion	158
2.9.9	Akkumulierende Parameter	160
2.9.10	Rekursion als Ersatz für While-Schleifen	162
2.10	Typen, Module, Klassen und Objekte	162
2.10.1	Strukturiertes Programmieren	163
2.10.2	Blockstrukturierung	164
2.10.3	Strukturierung der Daten	164
2.10.4	Objektorientierte Konstruktion neuer Datentypen	169
2.10.5	Modulares Programmieren	171
2.10.6	Schnittstellen – Interfaces	173
2.10.7	Objektorientiertes Programmieren	175
2.10.8	Vererbung	177
2.10.9	Summentypen in objektorientierten Sprachen	179
2.10.10	Datenkapselung	181
2.10.11	Funktionsobjekte – Closures	183
2.11	Verifikation	187
2.11.1	Vermeidung von Fehlern	188

2.11.2	Zwischenbehauptungen	188
2.11.3	Partielle Korrektheit	189
2.11.4	Zerlegung durch Zwischenbehauptungen	190
2.11.5	Zuweisungsregel	191
2.11.6	Rückwärtsbeweise	193
2.11.7	if-else-Regel	194
2.11.8	Abschwächungsregel und einarmige Alternative	195
2.11.9	Invarianten und while-Regel	195
2.11.10	Starke und schwache Invarianten	198
2.11.11	Programm-Verifizierer	200
2.11.12	do-Schleife	201
2.11.13	Terminierung	202
2.11.14	Beweis eines Programmschemas	203
2.12	Deklarative Sprachen	204
2.12.1	Prolog	204
2.12.2	Erlang	208
2.13	Zusammenfassung	212
3	Die Programmiersprache Java	213
3.1	Die lexikalischen Elemente von Java	215
3.1.1	Kommentare	215
3.1.2	Bezeichner	216
3.1.3	Schlüsselwörter	216
3.1.4	Literale	217
3.2	Datentypen und Methoden	218
3.2.1	Variablen	219
3.2.2	Referenz-Datentypen	220
3.2.3	Arrays	220
3.2.4	Methoden	221
3.2.5	Klassen und Instanzen	223
3.2.6	Objekte und Referenzen	225
3.2.7	Objekt- und Klassenkomponenten	226
3.2.8	Attribute	227
3.2.9	Überladung	229
3.2.10	Konstruktoren	229
3.2.11	Aufzählungstypen	231
3.3	Ausführbare Java-Programme	232
3.3.1	Java-Dateien – Übersetzungseinheiten	234
3.3.2	Programme	234
3.3.3	Packages	235
3.3.4	Standard-Packages	237
3.4	Ausdrücke und Anweisungen	238
3.4.1	Arithmetische Operationen	238

3.4.2	Vergleichsoperationen	239
3.4.3	Boolesche Operationen	239
3.4.4	Bitweise Operationen.	240
3.4.5	Zuweisungsausdrücke	240
3.4.6	Anweisungsausdrücke	241
3.4.7	Sonstige Operationen.	242
3.4.8	Präzedenz der Operatoren	243
3.4.9	Einfache Anweisungen	244
3.4.10	Blöcke	244
3.4.11	Alternativ-Anweisungen	245
3.4.12	switch-Anweisung	246
3.4.13	Schleifen	247
3.4.14	Die for-Anweisung	248
3.4.15	break- und continue-Anweisungen	250
3.5	Klassen und Objekte	250
3.5.1	Vererbung	252
3.5.2	Dynamische (späte) Bindung von Methoden.	257
3.5.3	Statische Bindung von Feldern	258
3.5.4	Zugriffsrechte von Feldern und Methoden	258
3.5.5	Attribute von Klassen	259
3.5.6	Abstrakte Klassen	259
3.5.7	Rekursiv definierte Klassen.	261
3.5.8	Schnittstellen (Interfaces)	263
3.5.9	Wrapper-Klassen	266
3.5.10	Generische Klassen	267
3.5.11	Vererbung generischer Typen	268
3.5.12	Typschränken.	269
3.6	Fehler und Ausnahmen	270
3.6.1	Exceptions in Java	270
3.6.2	Zusicherungen – Assertions.	273
3.7	Dateien: Ein- und Ausgabe.	277
3.7.1	Öffnen und Schließen von Dateien	277
3.7.2	Dateidialog.	278
3.7.3	Textdateien schreiben und lesen	279
3.7.4	Objekte in Dateien schreiben und lesen.	280
3.8	Threads	281
3.8.1	Thread-Erzeugung	281
3.8.2	Kontrolle der Threads	283
3.8.3	Thread-Synchronisation.	284
3.8.4	Deadlock	286
3.9	Grafische Benutzeroberflächen mit Java (AWT)	287
3.9.1	Ein erstes Fenster	288
3.9.2	Ereignisse.	289
3.9.3	Adapterklassen und anonyme Klassen	290

3.9.4	Beispiel für eine Ereignisbehandlung	291
3.9.5	Buttons	292
3.9.6	Grafikausgabe in Fenstern	294
3.9.7	Maus-Ereignisse	295
3.9.8	Paint	299
3.9.9	Weitere Bedienelemente von Programmen und Fenstern	299
3.10	Ausblick: Java 8 und dann ...	300
3.10.1	Ausblick	302
4	Algorithmen und Datenstrukturen	305
4.1	Suchalgorithmen	307
4.1.1	Lineare Suche	307
4.1.2	Exkurs: Runden, Logarithmen und Stellenzahl	309
4.1.3	Binäre Suche	310
4.1.4	Lineare Suche vs. binäre Suche	311
4.1.5	Komplexität von Algorithmen	312
4.2	Einfache Sortierverfahren	315
4.2.1	Datensätze und Schlüssel	315
4.2.2	Invarianten und Assertions	318
4.2.3	BubbleSort	320
4.2.4	SelectionSort	322
4.2.5	InsertionSort	324
4.2.6	Laufzeitvergleiche der einfachen Sortieralgorithmen	326
4.2.7	ShellSort und CombSort	327
4.3	Schnelle Sortieralgorithmen	328
4.3.1	Divide and Conquer – teile und herrsche	328
4.3.2	QuickSort	329
4.3.3	Die Partitionierung	330
4.3.4	Korrektheit von QuickSort	332
4.3.5	Komplexität von QuickSort	332
4.3.6	MergeSort	333
4.3.7	Stabilität und RadixSort	335
4.3.8	Optimalität von Sortieralgorithmen	336
4.3.9	Distribution Sort	337
4.3.10	Laufzeit der schnellen Sortieralgorithmen	340
4.3.11	Paralleles Sortieren	342
4.3.12	Externes Sortieren	345
4.4	Abstrakte Datenstrukturen	346
4.4.1	Datenstruktur = Menge + Operationen	346
4.4.2	Die axiomatische Methode	346
4.5	Stacks	347
4.5.1	Stackoperationen	348
4.5.2	Implementierung durch ein Array	350

4.5.3	Implementierung durch eine Liste	351
4.5.4	Auswertung von Postfix-Ausdrücken	352
4.5.5	Entrekursivierung.	353
4.5.6	Stackpaare	354
4.6	Queues, Puffer, Warteschlangen	355
4.6.1	Implementierung durch ein „zirkuläres“ Array	355
4.6.2	Implementierung durch eine zirkuläre Liste	357
4.6.3	DeQueues: Queues mit zwei gleichberechtigten Enden	358
4.6.4	Anwendung von Puffern	358
4.7	Container Datentypen	359
4.7.1	Listen	361
4.7.2	Einfach verkettete Listen	363
4.7.3	Listen als Verallgemeinerung von Stacks und Queues	368
4.7.4	Array-Listen.	369
4.7.5	Doppelt verkettete Listen.	370
4.7.6	Geordnete Listen und Skip-Listen	370
4.7.7	Adaptive Listen	371
4.8	Bäume	372
4.8.1	Beispiele von Bäumen	373
4.8.2	Binärbäume	373
4.8.3	Implementierung von Binärbäumen	375
4.8.4	Traversierungen	376
4.8.5	Kenngößen von Binärbäumen	380
4.8.6	Binäre Suchbäume	380
4.8.7	Implementierung von binären Suchbäumen	381
4.8.8	Balancierte Bäume	388
4.8.9	AVL-Bäume	389
4.8.10	2-3-4-Bäume	391
4.8.11	B-Bäume	392
4.8.12	Vollständige Bäume	393
4.8.13	Heaps	394
4.8.14	HeapSort	397
4.8.15	Priority-Queues	398
4.8.16	Bäume mit variabler Anzahl von Teilbäumen	398
4.9	Graphen	399
4.9.1	Wege und Zusammenhang	401
4.9.2	Repräsentationen von Graphen	401
4.9.3	Traversierungen	404
4.9.4	Tiefensuche und Backtracking	404
4.9.5	Breitensuche	405
4.9.6	Transitive Hülle	406
4.9.7	Kürzeste Wege	407
4.9.8	Schwere Probleme für Handlungsreisende	409
4.9.9	Eine Implementierung des TSP	411

4.10	Zeichenketten	415
4.10.1	Array-Implementierung	415
4.10.2	Nullterminierte Strings	415
4.10.3	Stringoperationen	416
4.10.4	Suchen in Zeichenketten	416
4.10.5	Der Boyer-Moore-Algorithmus	417
5	Rechnerarchitektur	419
5.1	Vom Transistor zum Chip	419
5.1.1	Chips	421
5.1.2	Chipherstellung	422
5.1.3	Kleinste Chip-Strukturen	423
5.1.4	Chipfläche und Anzahl der Transistoren	423
5.1.5	Weitere Chip-Parameter	424
5.1.6	Speicherbausteine	424
5.1.7	Logikbausteine	426
5.1.8	Schaltungsentwurf	426
5.2	Boolesche Algebra	427
5.2.1	Serien-parallele Schaltungen	428
5.2.2	Verknüpfung von Schaltgliedern	429
5.2.3	Serien-parallele Schaltglieder	430
5.2.4	Terme	430
5.2.5	Schaltfunktionen und -tabellen	431
5.2.6	Gleichungen	432
5.2.7	Dualität	432
5.2.8	SP-Schaltungen sind monoton	433
5.2.9	Negation	434
5.2.10	Boolesche Terme	434
5.2.11	Dualitätsprinzip	435
5.2.12	Realisierung von Schaltfunktionen – DNF	435
5.2.13	Konjunktive Normalform – KNF	437
5.2.14	Algebraische Umwandlung in DNF oder KNF	437
5.2.15	Aussagenlogik	438
5.2.16	Mengenalgebra	439
5.3	Digitale Logik	439
5.3.1	Logikgatter	440
5.3.2	Entwurf und Vereinfachung boolescher Schaltungen	442
5.3.3	KV-Diagramme	443
5.3.4	Spezielle Schaltglieder	445
5.3.5	Gatter mit mehreren Ausgängen	446
5.3.6	Codierer und Decodierer	446
5.3.7	Addierer	447

5.3.8	Logik-Gitter	449
5.3.9	Programmierbare Gitterbausteine	450
5.4	CMOS Schaltungen und VLSI Design	451
5.4.1	Logikgatter in CMOS-Technik	452
5.4.2	CMOS-Entwurf	454
5.4.3	Entwurf von CMOS Chips	455
5.4.4	VLSI-Werkzeuge	456
5.5	Sequentielle Logik	458
5.5.1	Gatterlaufzeiten	458
5.5.2	Rückgekoppelte Schaltungen	459
5.5.3	Einfache Anwendungen von Flip-Flops	461
5.5.4	Technische Schwierigkeiten	462
5.5.5	Synchrone und asynchrone Schaltungen	463
5.5.6	Getaktete Flip-Flops	464
5.5.7	Zustandsautomaten	465
5.5.8	Entwurf sequentieller Schaltungen	466
5.5.9	Eine Fußgängerampel	467
5.5.10	Die Konstruktion der Hardwarekomponenten	469
5.5.11	Tristate Puffer	469
5.5.12	Speicherzellen	470
5.5.13	MOS-Implementierung von Speicherzellen	471
5.5.14	Register und adressierbarer Speicher	473
5.5.15	Die Arithmetisch-Logische Einheit	475
5.6	Von den Schaltgliedern zur CPU	480
5.6.1	Busse	481
5.6.2	Mikrocodegesteuerte Operationen	482
5.6.3	Der Zugang zum Hauptspeicher	484
5.6.4	Von Neumann Architektur	486
5.6.5	Der Mikrobefehlsspeicher – das ROM	488
5.6.6	Sprünge	488
5.6.7	Berechnete Sprünge	489
5.6.8	Der Adressrechner	490
5.6.9	Ein Mikroprogramm	491
5.6.10	Maschinenbefehle	492
5.6.11	Der Maschinenspracheinterpretierer	494
5.6.12	Argumente	495
5.7	Assemblerprogrammierung	496
5.7.1	Maschinensprache und Assembler	497
5.7.2	Register der x86-Familie	497
5.7.3	Assemblerbefehle	499
5.7.4	Mehrzweckregister und Spezialregister	500
5.7.5	Flag-Register	500
5.7.6	Arithmetische Flags	501
5.7.7	Größenvergleiche	503

5.7.8	Logische Operationen	504
5.7.9	Sprünge	505
5.7.10	Struktur eines vollständigen Assemblerprogrammes.	507
5.7.11	Ein Beispielprogramm	508
5.7.12	Testen von Assemblerprogrammen	509
5.7.13	Speicheradressierung	510
5.7.14	Operationen auf Speicherblöcken	512
5.7.15	Multiplikation und Division.	513
5.7.16	Shift-Operationen.	513
5.7.17	LOOP-Befehle	515
5.7.18	Der Stack	516
5.7.19	Einfache Unterprogramme.	516
5.7.20	Parameterübergabe und Stack	518
5.7.21	Prozeduren und Funktionen	519
5.7.22	Makros	520
5.8	RISC-Architekturen	521
5.8.1	CISC.	521
5.8.2	Von CISC zu RISC	522
5.8.3	RISC-Prozessoren	523
5.8.4	Pipelining	524
5.8.5	Superskalare Architekturen	525
5.8.6	Cache-Speicher	526
5.8.7	Leistungsvergleiche	526
5.8.8	Konkrete RISC-Architekturen	527
5.9	Architektur der Intel-PC-Mikroprozessorfamilie	530
5.9.1	Datenstrukturen und Befehle des Pentium.	531
5.9.2	MMX- und SSE-Befehle	531
5.9.3	Adressierung	532
5.9.4	Die Segmentierungseinheit	533
5.9.5	Adressübersetzung	534
5.9.6	Betriebsarten des Pentium	535
6	Betriebssysteme	539
6.1	Basis-Software	540
6.2	Betriebsarten.	542
6.2.1	Teilhaberbetrieb.	543
6.2.2	Client-Server-Systeme	543
6.3	Verwaltung der Ressourcen	545
6.3.1	Dateisystem	546
6.3.2	Dateioperationen	547
6.3.3	Prozesse und Threads.	547
6.3.4	Vom Programm zum Prozess.	548

6.3.5	Prozessverwaltung	549
6.3.6	Prozesskommunikation	552
6.3.7	Kritische Abschnitte – wechselseitiger Ausschluss	553
6.3.8	Semaphore und Monitore	555
6.3.9	Deadlocks	556
6.3.10	Speicherverwaltung	557
6.3.11	Paging	558
6.3.12	Page faults	561
6.4	Das Betriebssystem UNIX	562
6.4.1	Linux	562
6.4.2	Das UNIX-Dateisystem	563
6.4.3	Dateinamen	564
6.4.4	Dateirechte	564
6.4.5	Namen und Pfade	566
6.4.6	Special files	567
6.4.7	Externe Dateisysteme	567
6.4.8	UNIX-Shells	567
6.4.9	UNIX-Kommandos	568
6.4.10	Optionen	569
6.4.11	Datei-Muster	570
6.4.12	Standard-Input/Standard-Output	571
6.4.13	Dateibearbeitung	571
6.4.14	Reguläre Ausdrücke	572
6.5	UNIX-Prozesse	573
6.5.1	Pipes	574
6.5.2	Sind Pipes notwendig?	575
6.5.3	Prozess-Steuerung	577
6.5.4	Multitasking	579
6.5.5	UNIX-Shell-Programmierung	581
6.5.6	Die C-Shell	581
6.5.7	Kommando-Verknüpfungen	582
6.5.8	Variablen	582
6.5.9	Shell-Scripts	584
6.5.10	Ausführung von Shell-Scripts	584
6.5.11	UNIX-Kommandos und Shell-Kommandos	584
6.5.12	UNIX als Mehrbenutzersystem	585
6.5.13	UNIX-Tools	586
6.5.14	Editoren	587
6.5.15	C und C++	588
6.5.16	Scanner- und Parsergeneratoren	589
6.5.17	Projektbearbeitung	591
6.6	X Window System	591
6.6.1	Window-Manager und Terminal Emulator	593
6.6.2	Grafische Oberflächen	593

6.7	MS-DOS und MS-Windows	594
6.7.1	Windows NT, Windows 2000	595
6.7.2	Windows XP	596
6.7.3	Windows Vista	597
6.7.4	Windows 7 und Windows 8	598
6.8	Alternative PC-Betriebssysteme	598
7	Rechnernetze	601
7.1	Rechner-Verbindungen	602
7.1.1	Signalübertragung	602
7.1.2	Physikalische Verbindung	604
7.1.3	Synchronisation	606
7.1.4	Bitcodierungen	606
7.2	Datenübertragung mit Telefonleitungen	608
7.2.1	ISDN	608
7.2.2	DSL, ADSL und T-DSL	610
7.3	Protokolle und Netze	611
7.3.1	Das OSI-Modell	612
7.3.2	Netze	614
7.3.3	Netztopologien	615
7.3.4	Netze von Netzen	617
7.3.5	Zugriffsverfahren	620
7.3.6	Wettkampfverfahren: CSMA-CD	620
7.4	Netztechnologien	621
7.4.1	Ethernet	622
7.4.2	FDDI	622
7.4.3	ATM	623
7.4.4	SONET/SDH	624
7.5	Drahtlose Netze	626
7.5.1	Bluetooth	627
7.5.2	WLAN Standards	628
7.5.3	Access Points	629
7.5.4	Datenübertragung mit mobilen Telefonnetzen	632
8	Das Internet	635
8.0.1	Bildung von Standards im Internet	636
8.1	Die TCP/IP Protokolle	638
8.1.1	Die Protokolle TCP und UDP	639
8.1.2	Das IP Protokoll	641
8.2	IP-Adressen	643
8.2.1	Adressklassen	643
8.2.2	Adressübersetzung	645
8.3	Das System der Domain-Namen	649

8.4	Intranet, Firewall und virtuelle private Netzwerke	651
8.5	Die Dienste im Internet.	653
8.5.1	E-Mail	654
8.5.2	News	658
8.5.3	FTP.	659
8.5.4	Secure Shell	660
8.5.5	Gopher	660
8.6	Das World Wide Web.	661
8.6.1	HTTP	663
8.6.2	HTML	664
8.6.3	Die Struktur eines HTML-Dokumentes	667
8.6.4	Querverweise: Links	668
8.6.5	Tabellen und Frames	669
8.6.6	Formulare.	670
8.6.7	Style Sheets	671
8.6.8	Weitere Möglichkeiten von HTML	672
8.7	Web-Programmierung	672
8.7.1	JavaScript.	672
8.7.2	Applets.	675
8.7.3	Die Struktur eines Applets.	676
8.7.4	Der Lebenszyklus eines Applets	677
8.7.5	Interaktionen	678
8.7.6	PHP	680
8.7.7	XML.	683
8.7.8	JSON	691
8.7.9	DOM, und Web 2.0.	692
8.7.10	Cloud Computing.	694
9	Theoretische Informatik und Compilerbau	695
9.1	Analyse von Programmtexten.	695
9.1.1	Lexikalische Analyse.	696
9.1.2	Syntaxanalyse	697
9.2	Reguläre Sprachen	698
9.2.1	Reguläre Ausdrücke.	699
9.2.2	Automaten und ihre Sprachen	701
9.2.3	Implementierung endlicher Automaten	703
9.2.4	ϵ -Transitionen und nichtdeterministische Automaten	704
9.2.5	Automaten für reguläre Sprachen	704
9.2.6	Von nichtdeterministischen zu deterministischen Automaten.	705
9.2.7	Anwendung: flex	706
9.3	Kontextfreie Sprachen	707
9.3.1	Kontextfreie Grammatiken	708

9.3.2	Ableitungen	709
9.3.3	Stackautomaten (Kellerautomaten)	710
9.3.4	Stackautomaten für beliebige kontextfreie Sprachen	712
9.3.5	Nichtdeterministische Algorithmen und Backtracking	712
9.3.6	Inhärent nichtdeterministische Sprachen	715
9.3.7	Ableitungsbaum, Syntaxbaum	715
9.3.8	Abstrakte Syntaxbäume	716
9.4	Grundlagen des Compilerbaus	717
9.4.1	Parsen durch rekursiven Abstieg (recursive descent)	718
9.4.2	LL(1)-Grammatiken	719
9.4.3	Äquivalente Grammatiken	721
9.4.4	Top-down und bottom-up	723
9.4.5	Shift-Reduce Parser	724
9.4.6	Die Arbeitsweise von Shift-Reduce-Parsern	725
9.4.7	Bottom-up Parsing	726
9.4.8	Konflikte	727
9.4.9	Ein nichtdeterministischer Automat mit Stack	727
9.4.10	Übergang zum deterministischen Automaten	730
9.4.11	Präzedenz	732
9.4.12	LR(1) und LALR(1)	733
9.4.13	Parsergeneratoren	734
9.4.14	lex/flex & yacc/bison	736
9.4.15	Grammatische Aktionen	737
9.4.16	Fehlererkennung	739
9.4.17	Synthetisierte Werte	739
9.4.18	Symboltabellen	740
9.4.19	Codeoptimierung	741
9.5	Berechenbarkeit	742
9.5.1	Berechenbare Funktionen	742
9.5.2	Codierungen	743
9.5.3	Grenzfälle berechenbarer Funktionen	745
9.5.4	Diagonalisierung	746
9.5.5	Nicht berechenbare Funktionen	746
9.5.6	Algorithmenbegriff und Churchsche These	747
9.5.7	Turingmaschinen	748
9.5.8	Turing-Post Programme	750
9.5.9	Turing-berechenbare Funktionen	751
9.5.10	Registermaschinen	751
9.5.11	GOTO-Programme	752
9.5.12	While-Programme	753
9.5.13	For-Programme (Loop-Programme)	755
9.5.14	Effiziente Algorithmen als For-Programme	756
9.5.15	Elementare (primitive) Rekursion	757
9.5.16	Allgemeine Rekursion (μ -Rekursion)	758

9.5.17	Die Ackermannfunktion	759
9.5.18	Berechenbare Funktionen – Churchsche These.....	760
9.5.19	Gödelisierung.....	761
9.5.20	Aufzählbarkeit und Entscheidbarkeit.....	762
9.5.21	Unlösbare Aufgaben	763
9.5.22	Semantische Probleme sind unentscheidbar	764
9.6	Komplexitätstheorie	765
9.6.1	Rückführung auf ja/nein-Probleme	766
9.6.2	Entscheidungsprobleme und Sprachen	767
9.6.3	Maschinenmodelle und Komplexitätsmaße.....	767
9.6.4	Sprachen und ihre Komplexität	768
9.6.5	Effiziente parallele Lösungen	769
9.6.6	Nichtdeterminismus.....	770
9.6.7	Die Klasse NP	771
9.6.8	Reduzierbarkeit	772
9.6.9	Der Satz von Cook.....	774
9.6.10	NP-Vollständigkeit	776
9.6.11	CLIQUE ist NP-vollständig.....	777
9.6.12	Praktische Anwendung von SAT-Problemen	778
9.6.13	$P = NP$?	780
10	Datenbanksysteme	781
10.1	Datenbanken und Datenbanksysteme.....	781
10.2	Datenmodelle	783
10.2.1	Entity/Relationship-Modell	783
10.2.2	Das Relationale Datenbankmodell.....	785
10.2.3	Relationen	786
10.2.4	Die relationale Algebra	787
10.2.5	Erweiterungen des relationalen Datenmodells	788
10.2.6	Vom E/R-Datenmodell zu einem relationalen Modell	788
10.3	Die Anfragesprache SQL	789
10.3.1	Datendefinition	789
10.3.2	Einfache Anfragen	791
10.3.3	Gruppierung und Aggregate	792
10.3.4	Verknüpfung verschiedener Relationen.....	793
10.3.5	Einfügen, Ändern und Löschen von Datensätzen	793
10.3.6	Mehrbenutzerbetrieb	794
10.4	Anwendungsprogrammierung in Java	796
10.4.1	Das SQL-Paket in Java	797
10.4.2	Aufbau einer Verbindung	798
10.4.3	Anfragen	799
10.5	Zusammenfassung	800

11 Grafikprogrammierung	801
11.1 Hardware	801
11.1.1 Auflösungen	802
11.1.2 Farben	802
11.2 Rastergrafik und Vektorgrafik	803
11.2.1 Umrechnung in Rastergrafik	804
11.2.2 Bresenham Algorithmus	805
11.2.3 Quadrees und Octrees	806
11.3 Einfache Programmierbeispiele	808
11.3.1 Mandelbrot- und Julia-Mengen	810
11.3.2 Turtle-Grafik	813
11.3.3 L-Systeme	815
11.3.4 Ausblick	818
11.4 3-D-Grafikprogrammierung	818
11.4.1 Sichtbarkeit	819
11.4.2 Beleuchtungsmodelle	820
11.4.3 Ray-Tracing	823
11.4.4 Photon-Mapping	824
11.4.5 Die Radiosity Methode	825
11.4.6 Ausblick	825
12 Softwareentwicklung	827
12.1 Herausforderungen an die Softwareentwicklung	828
12.2 Softwareentwicklungsprozesse	830
12.2.1 Wasserfallmodelle	830
12.2.2 Nichtsequentielle Vorgehensmodelle	833
12.2.3 Modelle zur inkrementellen Softwareentwicklung	834
12.3 Anforderungsanalyse	837
12.4 Objektorientierte Softwareentwicklungsmethoden	838
12.4.1 Prinzipien der Objektorientierung	839
12.4.2 Standardisierung der objektorientierten Modellierung	840
12.4.3 Die Modellierungssprache UML	840
12.5 Grundprinzipien des Softwareentwurfs	844
12.5.1 Strukturierte Programmierung	844
12.5.2 Schrittweise Verfeinerung und Top-Down-Entwurf	844
12.5.3 Geheimnisprinzip, Datenabstraktion und Modularisierung	845
12.5.4 Softwarearchitektur	846
12.5.5 Entwurfsmuster	847
12.5.6 Frameworks und Bibliotheken	847
12.6 Qualitätssicherung von Softwaresystemen	848
12.6.1 Testen von Softwaresystemen	850
12.6.2 Qualitätsnormen und Zertifizierung	852

12.7 Projektmanagement	853
12.7.1 Projektinitialisierung und -planung	854
12.7.2 Projektsteuerung und -koordination.	854
12.7.3 Projektabschluss und -bericht	855
12.8 Versionsverwaltung von Softwareprojekten.	855
12.9 Softwarewartung und -evolution	856
A Literatur	857
A.1 Einführende Bücher	857
A.2 Lehrbücher der Informatik	857
A.3 Programmieren in Java und anderen Sprachen.	858
A.4 Algorithmen und Datenstrukturen	859
A.5 Rechnerarchitektur	859
A.6 Betriebssysteme	860
A.7 Rechnernetze.	860
A.8 Internet	861
A.9 Theoretische Informatik und Compilerbau.	862
A.10 Datenbanken	863
A.11 Grafikprogrammierung	864
A.12 Software-Entwicklung	864
A.13 Mathematischer Hintergrund	866
Stichwortverzeichnis	867