

Hartmut Ernst • Jochen Schmidt • Gerd Beneken

# Grundkurs Informatik

Grundlagen und Konzepte für die  
erfolgreiche IT-Praxis – Eine umfassende,  
praxisorientierte Einführung

**5., vollständig überarbeitete Auflage**

**Springer Vieweg**

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Was ist eigentlich Informatik?	1
1.2	Zur Geschichte der Informatik	3
1.2.1	Frühe Zähl- und Rechensysteme	3
1.2.2	Die Entwicklung von Rechenmaschinen	4
1.2.3	Die Computer-Generationen	7
1.3	Prinzipieller Aufbau von Computern	12
1.3.1	Analog- und Digitalrechner	12
1.3.2	Das EVA-Prinzip	12
1.3.3	Zentraleinheit und Busstruktur	12
1.3.4	Systemkomponenten	15
1.4	Zahlensysteme und binäre Arithmetik	17
1.4.1	Darstellung von Zahlen	17
1.4.2	Umwandlung von Zahlen in verschiedene Darstellungssysteme	18
1.4.3	Binäre Arithmetik	24
1.4.4	Gleitkommazahlen	30
	Literatur	35
<b>2</b>	<b>Nachricht und Information</b>	<b>37</b>
2.1	Abgrenzung der Begriffe Nachricht und Information	37
2.2	Biologische Aspekte	39
2.2.1	Sinnesorgane	39
2.2.2	Datenverarbeitung im Gehirn	39
2.2.3	Der genetische Code	40
2.3	Diskretisierung von Nachrichten	43
2.3.1	Abtastung	43
2.3.2	Quantisierung	44
2.4	Wahrscheinlichkeit und Kombinatorik	47
2.4.1	Die relative Häufigkeit	47
2.4.2	Die mathematische Wahrscheinlichkeit	47
2.4.3	Totale Wahrscheinlichkeit und Bayes-Formel	49
2.4.4	Statistische Kenngrößen	54
2.4.5	Fakultät und Binomialkoeffizienten	55
2.4.6	Kombinatorik	56
2.5	Information und Wahrscheinlichkeit	60
2.5.1	Der Informationsgehalt einer Nachricht	60
2.5.2	Die Entropie einer Nachricht	62
2.5.3	Zusammenhang mit der physikalischen Entropie	64
	Literatur	68

<b>3</b>	<b>Codierung</b>	<b>69</b>
3.1	Grundbegriffe	69
3.1.1	Definition des Begriffs Codierung	69
3.1.2	Mittlere Wortlänge und Code-Redundanz	70
3.1.3	Beispiele für Codes	71
3.2	Code-Erzeugung	75
3.2.1	Codebäume	75
3.2.2	Der Huffman-Algorithmus	75
3.2.3	Der Fano-Algorithmus	79
3.3	Codesicherung	82
3.3.1	Stellendistanz und Hamming-Distanz	82
3.3.2	m-aus-n-Codes	85
3.3.3	Codes mit Paritätsbits	85
3.3.4	Fehlertolerante Gray-Codes	89
3.3.5	Definition linearer Codes	91
3.3.6	Lineare Hamming-Codes	94
3.3.7	Zyklische Codes und Code-Polynome	98
3.3.8	CRC-Codes	100
3.3.9	Sicherung nicht-binärer Codes	103
3.3.10	Reed-Solomon Codes	107
3.4	Datenkompression	115
3.4.1	Vorbemerkungen und statistische Datenkompression	115
3.4.2	Arithmetische Codierung	116
3.4.3	Laufängen-Codierung	118
3.4.4	Differenz-Codierung	120
3.4.5	Der LZW-Algorithmus	123
3.4.6	Datenreduktion durch unitäre Transformationen (JPEG)	128
	Literatur	135
<b>4</b>	<b>Verschlüsselung</b>	<b>137</b>
4.1	Klassische Verfahren	140
4.1.1	Substitutions-Chiffren	140
4.1.2	Transpositions-Chiffren und Enigma	143
4.2	Moderne symmetrische Verfahren	150
4.2.1	Der Data Encryption Standard (DES)	150
4.2.2	Der Advanced Encryption Standard (AES)	151
4.2.3	One-Time-Pads und Stromchiffren	154
4.3	Moderne asymmetrische Verfahren	156
4.3.1	Diffie-Hellman Schlüsselaustausch	156
4.3.2	Der RSA-Algorithmus	160
4.3.3	Digitale Unterschrift	165
	Literatur	168
<b>5</b>	<b>Computerhardware und Maschinensprache</b>	<b>169</b>
5.1	Digitale Grundsaltungen	169
5.1.1	Stromkreise	169

5.1.2	Dioden, Transistoren und integrierte Schaltkreise	171
5.1.3	Logische Gatter	174
5.2	Boolesche Algebra und Schaltfunktionen	177
5.2.1	Aussagenlogik	177
5.2.2	Der boolesche Verband	179
5.2.3	Das boolesche Normalform-Theorem	181
5.2.4	Vereinfachen boolescher Ausdrücke	182
5.3	Schaltnetze und Schaltwerke	185
5.3.1	Schaltnetze	185
5.3.2	Spezielle Schaltnetze	186
5.3.3	Schaltwerke	189
5.4	Die Funktion einer CPU am Beispiel des M68000	194
5.4.1	Die Anschlüsse der CPU M68000	194
5.4.2	Der innere Aufbau der CPU M68000	199
5.4.3	Befehlsformate und Befehlsausführung	204
5.4.4	Adressierungsarten	208
5.5	Maschinensprache und Assembler	213
5.5.1	Einführung	213
5.5.2	Der Befehlssatz des M68000	214
5.5.3	Programmbeispiele	223
	Literatur	226
<b>6</b>	<b>Rechnerarchitektur</b>	<b>227</b>
6.1	Überblick	227
6.2	Die von-Neumann-Architektur	230
6.2.1	Komponenten eines von-Neumann-Rechners	230
6.2.2	Operationsprinzip	232
6.3	Befehlssatz	234
6.3.1	Mikroprogramme und CISC	234
6.3.2	Reduced Instruction Set Computer: RISC	235
6.3.3	Abwärtskompatibilität	235
6.4	Klassifikation nach Flynn	236
6.5	Parallelität innerhalb einer Befehlssequenz	238
6.5.1	Fließbandverarbeitung – Optimierte Befehlsausführung	238
6.5.2	Superskalare Mikroprozessoren	239
6.5.3	VLW – Very Long Instruction Word	240
6.6	Parallelität in Daten nutzen: Vektorprozessoren und Vektorrechner	240
6.7	Parallele Ausführung mehrerer Befehlssequenzen	241
6.7.1	Simultanes Multi-Threading innerhalb einer CPU	243
6.7.2	Multi-Core-CPU	243
6.7.3	Multiprozessor-Systeme	244
6.7.4	Multicomputer-Systeme	245
6.8	Speicherhierarchie	245
6.8.1	Speichertechnologien: Register, Cache und Hauptspeicher	245
6.8.2	Caching	247
6.8.3	Memory Management Unit und virtueller Speicher	249

6.8.4	Festplatten	249
6.8.5	Flash-Speicher und Solid State Disks	250
6.9	Ein- und Ausgabe	251
6.9.1	Unterbrechungen (Interrupts)	251
6.9.2	Direct Memory Access	252
6.10	Verbindungsstrukturen	253
6.10.1	Gemeinsamer Bus	253
6.10.2	Zugriffsprotokolle für Busse und gemeinsame Speicher	255
6.10.3	Punkt-Zu-Punkt Verbindungen	256
6.10.4	Weitere Verbindungsstrukturen	256
6.10.5	Allgemeine topologische Verbindungsstrukturen	257
6.11	Mikrocontroller und Spezialprozessoren	259
6.11.1	Mikrocontroller	259
6.11.2	Digitale Signalprozessoren	260
6.11.3	Grafikprozessoren	260
	Literatur	261
<b>7</b>	<b>Rechnernetze</b>	<b>263</b>
7.1	Das OSI-Schichtenmodell der Datenkommunikation	263
7.2	Bitübertragungsschicht	267
7.3	Technologien der Sicherungsschicht	274
7.3.1	Netze im Nahbereich (PAN)	274
7.3.2	Lokale Netze: LAN und WLAN	275
7.3.3	Vorgriff: Leitungs- und Paketvermittlung	279
7.3.4	Datenfernübertragung und der Zugang zum Internet	281
7.3.5	Die Behandlung von Übertragungsfehlern	286
7.4	Netzwerk- und Transportschicht: TCP/IP und das Internet	287
7.4.1	Überblick über das Internet	287
7.4.2	IP: Internet Protocol	290
7.4.3	TCP: Transmission Control Protocol	292
7.4.4	UDP: User Datagram Protocol	293
7.5	Anwendungsschicht: Von DNS bis HTTP und URIs	294
7.5.1	DNS: Domain Name System	294
7.5.2	E-Mail	295
7.5.3	IRC: Internet Relay Chat	296
7.5.4	FTP: File Transfer Protocol	296
7.5.5	SSH: secure shell und TELNET: teletype network	296
7.5.6	HTTP: Hypertext Transfer Protocol	297
7.5.7	URL Uniform Resource Identifier	298
	Literatur	299
<b>8</b>	<b>Betriebssysteme</b>	<b>301</b>
8.1	Überblick	301
8.1.1	Aufgaben	303
8.1.2	Betriebsarten	304
8.2	Betriebssystem-Architekturen	305

8.3	Aufgaben eines Betriebssystems im Detail	309
8.3.1	Prozessverwaltung	309
8.3.2	Synchronisation	313
8.3.3	Interprozess-Kommunikation	316
8.3.4	Speicherverwaltung und virtueller Speicher	317
8.3.5	Geräteverwaltung und -treiber	320
8.3.6	Dateiverwaltung	320
8.4	Benutzerschnittstelle: Shell und GUI	322
8.4.1	Kommandozeilen-Interpreter am Beispiel UNIX	322
8.4.2	Besonderheiten am Beispiel der UNIX-Shell	323
8.4.3	Grafische Benutzerschnittstelle	326
8.5	Beispiele für Betriebssysteme	327
8.5.1	Microsoft-Windows	328
8.5.2	UNIX, LINUX und Android	328
8.6	Betriebssystem-Virtualisierung	330
8.6.1	Anwendungsbereiche	331
8.6.2	Hypervisoren	331
8.6.3	Virtuelle Maschinen	333
8.6.4	Grundlegende Aktivitäten der Virtualisierung	334
	Literatur	335

## **9 Datenbanken 337**

9.1	Einführung und Definition	337
9.2	Relationale Datenbankmanagement-Systeme	341
9.2.1	Relationen	341
9.2.2	Schlüssel	342
9.2.3	Beziehungen (Relationships)	343
9.3	Relationale Algebra	343
9.4	Die Datenbanksprache SQL	347
9.4.1	SQL als deklarative Sprache	348
9.4.2	Definition des Datenbankschemas	348
9.4.3	Einfügen, Ändern und Löschen von Daten	350
9.4.4	Suchen mit SELECT	351
9.4.5	Programmiersprachen und SQL	352
9.5	NoSQL	353
9.6	Transaktionen, OLTP und ACID	355
9.7	OLAP, Data Warehousing und Data-Mining	356
9.8	Semi-Strukturierte Daten mit XML	362
9.8.1	Der Aufbau von XML-Dokumenten	363
9.8.2	Wohlgeformtheit und Validität	365
9.8.3	XML-Schema	365
9.8.4	XPath	366
9.8.5	XSL: Extended Style Sheet Language	366
	Literatur	369

<b>10</b>	<b>Automatentheorie und formale Sprachen</b>	<b>371</b>
10.1	Grundbegriffe der Automatentheorie	371
10.1.1	Definition von Automaten	371
10.1.2	Darstellung von Automaten	374
10.1.3	Die akzeptierte Sprache von Automaten	376
10.1.4	Kellerautomaten	383
10.1.5	Turing-Maschinen	386
10.2	Einführung in die Theorie der formalen Sprachen	393
10.2.1	Definition von formalen Sprachen	393
10.2.2	Die Chomsky-Hierarchie	394
10.2.3	Das Pumping-Theorem	400
10.2.4	Die Analyse von Wörtern	403
10.2.5	Compiler	408
	Literatur	413
<b>11</b>	<b>Algorithmen – Berechenbarkeit und Komplexität</b>	<b>415</b>
11.1	Berechenbarkeit	417
11.1.1	Entscheidungsproblem und Church-Turing These	417
11.1.2	Das Halteproblem	420
11.1.3	Satz von Rice und weitere unentscheidbare Probleme	422
11.1.4	LOOP-, WHILE- und GOTO-Berechenbarkeit	423
11.1.5	Primitiv rekursive und $\lambda$ -rekursive Funktionen	426
11.2	Komplexität	430
11.2.1	Die Ordnung der Komplexität: $\lambda$ -Notation	431
11.2.2	Analyse von Algorithmen	435
11.2.3	Die Komplexitätsklassen P und NP	439
11.2.4	NP-vollständige Probleme	441
11.2.5	Weitere Komplexitätsklassen	445
11.3	Probabilistische Algorithmen	448
11.3.1	Pseudo-Zufallszahlen	448
11.3.2	Monte-Carlo-Methoden	452
11.3.3	Probabilistischer Primzahltest	455
11.4	Rekursion	459
11.4.1	Definition und einführende Beispiele	459
11.4.2	Rekursive Programmierung und Iteration	460
11.4.3	Backtracking	464
	Literatur	466
<b>12</b>	<b>Suchen und Sortieren</b>	<b>469</b>
12.1	Einfache Suchverfahren	469
12.1.1	Sequentielle Suche	469
12.1.2	Binäre Suche	470
12.1.3	Interpolationssuche	471
12.1.4	Radix-Suche	473
12.2	Suchen von Mustern in Zeichenketten	474
12.2.1	Musterabgleich durch sequentielles Vergleichen	474

12.2.2	Musterabgleich durch Automaten	475
12.2.3	Die Verfahren von Boyer-Moore und Knuth-Morris-Pratt	476
12.2.4	Ähnlichkeit von Mustern und Levenshtein-Distanz	478
12.3	Gestreute Speicherung (Hashing)	480
12.3.1	Hash-Funktionen	480
12.3.2	Kollisionsbehandlung	482
12.3.3	Komplexitätsberechnung	487
12.4	Direkte Sortierverfahren	489
12.4.1	Vorbemerkungen	489
12.4.2	Sortieren durch direktes Einfügen (Insertion Sort)	492
12.4.3	Sortieren durch direktes Auswählen (Selection Sort)	494
12.4.4	Sortieren durch direktes Austauschen (Bubblesort)	496
12.5	Höhere Sortierverfahren	499
12.5.1	Shellsort	499
12.5.2	Quicksort	500
12.5.3	Vergleich der Sortierverfahren	504
12.6	Sortieren externer Dateien	506
12.6.1	Grundprinzipien des sequentiellen Datenzugriffs	506
12.6.2	Sequentielle Speicherorganisation	509
12.6.3	Direktes Mischen (Direct Merge, Mergesort)	513
12.6.4	Natürliches Mischen (Natural Merge)	516
12.6.5	n-Band-Mischen	518
	Literatur	521

## **13 Bäume und Graphen** **523**

13.1	Binärbäume	523
13.1.1	Definitionen	523
13.1.2	Speichern und Durchsuchen von Binärbäumen	525
13.1.3	Binäre Suchbäume	530
13.1.4	Ausgleichen von Bäumen und AVL-Bäume	537
13.1.5	Heaps und Heapsort	541
13.2	Vielwegbäume	548
13.2.1	Rückführung auf Binärbäume	548
13.2.2	Definition von (a,b)-Bäumen und B-Bäumen	549
13.2.3	Operationen auf B-Bäumen	552
13.3	Graphen	560
13.3.1	Definitionen und einführende Beispiele	560
13.3.2	Speicherung von Graphen	564
13.3.3	Suchen, Einfügen und Löschen von Knoten und Kanten	570
13.3.4	Durchsuchen von Graphen	571
13.3.5	Halbordnung und topologisches Sortieren	584
13.3.6	Minimal spannende Bäume	587
13.3.7	Union-Find Algorithmen	591
	Literatur	595



<b>14 Höhere Programmiersprachen und C</b>	<b>597</b>
14.1 Zur Struktur höherer Programmiersprachen	597
14.1.1 Überblick über höhere Programmiersprachen	597
14.1.2 Ebenen des Informationsbegriffs in Programmiersprachen	602
14.1.3 Systeme und Strukturen	604
14.2 Methoden der Syntaxbeschreibung	607
14.2.1 Die Backus-Naur Form	607
14.2.2 Syntaxgraphen	611
14.2.3 Eine einfache Sprache als Beispiel: C -	611
14.3 Einführung in die Programmiersprache C	616
14.3.1 Der Aufbau von C-Programmen	617
14.3.2 Einfache Datentypen	621
14.3.3 Strukturierte Standard-Datentypen	626
14.3.4 Operatoren und Ausdrücke	629
14.3.5 Anweisungen	632
14.3.6 Funktionen	636
14.3.7 Ein- und Ausgabefunktionen	641
14.3.8 Verarbeitung von Zeichenketten	643
14.3.9 Das Zeigerkonzept in C	645
14.4 Sequentielle Datenstrukturen mit C	654
14.4.1 Vorbemerkungen zu Algorithmen und Datenstrukturen	654
14.4.2 Lineare Listen	656
14.4.3 Stapel und Schlangen	660
Literatur	665
<b>15 Objektorientierte Programmiersprachen und Java</b>	<b>667</b>
15.1 Entstehung objektorientierter Sprachen	667
15.2 Einführung in die Programmiersprache Java	670
15.2.1 Grundlegender Aufbau eines Java-Programms	672
15.2.2 Syntax ähnlich wie in C	674
15.2.3 Datentypen und Variablen: Statische Typisierung	675
15.3 Klassen und Objekte	680
15.3.1 Attribute und Methoden	680
15.3.2 Statische Attribute und Methoden	682
15.3.3 Pakete (Packages)	683
15.3.4 Kapselung und Geheimnisprinzip	685
15.3.5 Vererbung und Polymorphie	687
15.4 Fortgeschrittene Java-Themen	690
15.4.1 Generische Klassen, Behälter und Algorithmen	690
15.4.2 Ausnahmen und Fehlerbehandlung	694
15.4.3 Annotationen und Reflection	695
15.4.4 Testgetriebene Entwicklung mit Java	697
15.4.5 Threads, Streams und parallele Verarbeitung	698
15.4.6 Lambda-Ausdrücke und funktionale Programmierung	702
15.4.7 Das Java-Ökosystem	702
Literatur	704

<b>16 Anwendungsprogrammierung im Internet</b>	<b>707</b>
16.1 Client-Server-Systeme	707
16.2 Grundlegende Technologien	708
16.2.1 HTML	708
16.2.2 DOM: Domain Object Model	714
16.2.3 CSS: Cascading Style Sheets	715
16.3 Webanwendungen	719
16.3.1 HTML Formulare	719
16.3.2 Auswertung von Formularen	721
16.4 JavaScript	723
16.4.1 Grundlegende Eigenschaften	723
16.4.2 Funktionen	725
16.4.3 Objekte und Prototypen	727
16.4.4 JSON: JavaScript Object Notation	729
16.4.5 JavaScript und DOM	730
16.4.6 Ereignisgesteuerte Programmierung mit JavaScript	732
16.4.7 AJAX: Asynchronous JavaScript And XML	733
16.5 Serverseitige Skripte mit PHP	734
16.5.1 Grundlegende Eigenschaften	734
16.5.2 Arrays	736
16.5.3 Funktionen	738
16.5.4 Objektorientierte Programmierung in PHP	740
16.5.5 Datenübergabe von HTML-Formularen an PHP-Skripte	741
16.5.6 Sitzungsdaten: Session und Cookie	742
16.5.7 Datei- und Datenbankzugriff mit PHP	743
Literatur	745
<b>17 Software-Engineering</b>	<b>747</b>
17.1 Überblick	747
17.1.1 Was ist Software?	747
17.1.2 Was bedeutet Engineering?	748
17.1.3 Warum ist Software-Engineering schwierig?	749
17.2 Tätigkeiten im Software-Lebenszyklus	751
17.2.1 Anforderungsanalyse und Spezifikation	751
17.2.2 Architekturentwurf	752
17.2.3 Implementierung	752
17.2.4 Test und Integration	752
17.2.5 Inbetriebnahme	753
17.2.6 Wartung und Weiterentwicklung	753
17.3 Querschnittsdisziplinen	754
17.3.1 Projektmanagement	754
17.3.2 Qualitätsmanagement	755
17.3.3 Konfigurationsmanagement	756
17.4 Vorgehensmodelle	757
17.4.1 Basismodelle	759
17.4.2 V-Modell XT als plangetriebenes Vorgehensmodell	761

17.4.3 Serum als agiles Vorgehensmodell (-Framework)	763
17.5 Modelle im Software-Engineering	765
17.5.1 Vom Problem zur Lösung	765
17.5.2 Die Unified Modeling Language	766
17.5.3 Ausgewählte Diagramme der UML im Detail	768
17.6 Hilfsmittel für den Entwurf von Algorithmen	776
17.6.1 Pseudocode	776
17.6.2 Flussdiagramme	776
17.6.3 Struktogramme nach Nassi-Shneiderman	777
17.6.4 Entscheidungstabellen	780
Literatur	783
<b>Index</b>	<b>785</b>