

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 | 10 |
| 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 | 14 |
| 1.3.6 Logische Werte und logische Verknüpfungen | 15 |
| 1.3.7 Programme | 16 |
| 1.3.8 Bilder und Musikstücke | 16 |
| 1.4 Zahlendarstellungen | 17 |
| 1.4.1 Binärdarstellung | 17 |
| 1.4.2 Das Oktalsystem und das Hexadezimalsystem | 18 |
| 1.4.3 Umwandlung in das Binär-, Oktal- oder Hexadezimalsystem | 19 |
| 1.4.4 Arithmetische Operationen | 21 |
| 1.4.5 Darstellung ganzer Zahlen | 22 |
| 1.4.6 Die Zweierkomplementdarstellung | 23 |
| 1.5 Standardformate für ganze Zahlen | 25 |
| 1.5.1 Gleitpunktzahlen: Reelle Zahlen | 26 |
| 1.5.2 Real-Zahlenbereiche in Programmiersprachen | 29 |
| 1.5.3 Daten – Informationen | 29 |
| 1.5.4 Informationsverarbeitung – Datenverarbeitung | 30 |

| | | |
|----------|--|-----------|
| 1.6 | Hardware..... | 31 |
| 1.6.1 | PCs, Workstations, Mainframes, Super-Computer..... | 31 |
| 1.6.2 | Aufbau von Computersystemen..... | 32 |
| 1.6.3 | Der Rechner von außen..... | 33 |
| 1.6.4 | Das Innenleben | 34 |
| 1.6.5 | Ein Motherboard..... | 39 |
| 1.6.6 | Die Aufgabe der CPU..... | 41 |
| 1.6.7 | Die Organisation des Hauptspeichers..... | 43 |
| 1.6.8 | Speichermedien..... | 47 |
| 1.6.9 | Magnetplatten | 47 |
| 1.6.10 | Disketten | 49 |
| 1.6.11 | Festplattenlaufwerke | 49 |
| 1.6.12 | Optische Laufwerke | 51 |
| 1.6.13 | Vergleich von Speichermedien..... | 53 |
| 1.6.14 | Bildschirme | 54 |
| 1.6.15 | Text- und Grafikmodus..... | 55 |
| 1.7 | Von der Hardware zum Betriebssystem | 55 |
| 1.7.1 | Schnittstellen und Treiber..... | 57 |
| 1.7.2 | BIOS | 59 |
| 1.7.3 | Die Aufgaben des Betriebssystems..... | 60 |
| 1.7.4 | Prozess- und Speicherverwaltung..... | 60 |
| 1.7.5 | Dateiverwaltung..... | 61 |
| 1.7.6 | DOS, Windows und Linux..... | 63 |
| 1.7.7 | Bediensysteme | 64 |
| 1.8 | Anwendungsprogramme..... | 67 |
| 1.8.1 | Textverarbeitung..... | 67 |
| 1.8.2 | Zeichen und Schriftarten..... | 68 |
| 1.8.3 | Formatierung..... | 69 |
| 1.8.4 | Desktop Publishing | 70 |
| 1.8.5 | Textbeschreibungssprachen | 71 |
| 1.8.6 | Tabellenkalkulation: spread sheets | 75 |
| 1.8.7 | Der Rechner als Fenster zur Welt..... | 77 |
| 1.8.8 | Wie geht es weiter?..... | 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 | BASIC..... | 87 |
| 2.1.7 | Pascal | 88 |
| 2.1.8 | Java | 89 |

| | | |
|--------|--|-----|
| 2.1.9 | Prolog | 89 |
| 2.2 | Spezifikationen, Algorithmen, Programme | 91 |
| 2.2.1 | Spezifikationen | 91 |
| 2.2.2 | Algorithmen | 93 |
| 2.2.3 | Algorithmen als Lösung von Spezifikationen | 98 |
| 2.2.4 | Terminierung | 98 |
| 2.2.5 | Elementare Aktionen | 99 |
| 2.2.6 | Zuweisungen | 100 |
| 2.2.7 | Vom Algorithmus zum Programm | 101 |
| 2.2.8 | Ressourcen | 103 |
| 2.3 | Daten und Datenstrukturen | 104 |
| 2.3.1 | Der Begriff der Datenstruktur | 104 |
| 2.3.2 | Boolesche Werte | 105 |
| 2.3.3 | Zahlen | 108 |
| 2.3.4 | Natürliche Zahlen | 108 |
| 2.3.5 | Der Datentyp Integer | 110 |
| 2.3.6 | Rationale Zahlen | 112 |
| 2.3.7 | Reelle Zahlen | 112 |
| 2.3.8 | Mehrsortige Datenstrukturen | 113 |
| 2.3.9 | Zeichen | 115 |
| 2.3.10 | Zusammengesetzte Datentypen - Strings | 116 |
| 2.3.11 | Benutzerdefinierte Datenstrukturen | 118 |
| 2.3.12 | Informationsverarbeitung und Datenverarbeitung | 119 |
| 2.4 | Speicher, Variablen und Ausdrücke | 121 |
| 2.4.1 | Deklarationen | 121 |
| 2.4.2 | Initialisierung | 123 |
| 2.4.3 | Kontexte | 123 |
| 2.4.4 | Ausdrücke, Terme | 123 |
| 2.4.5 | Auswertung von Ausdrücken | 126 |
| 2.4.6 | Funktionsdefinitionen | 127 |
| 2.4.7 | Typfehler | 129 |
| 2.4.8 | Seiteneffekte | 130 |
| 2.5 | Der Kern imperativer Sprachen | 131 |
| 2.5.1 | Zuweisungen | 131 |
| 2.5.2 | Kontrollstrukturen | 133 |
| 2.5.3 | Drei Kontrollstrukturen genügen | 133 |
| 2.5.4 | Die sequentielle Komposition | 134 |
| 2.5.5 | Die Alternativanweisung | 135 |
| 2.5.6 | Die while-Schleife | 136 |
| 2.5.7 | Unterprogramme | 137 |
| 2.5.8 | Lauffähige Programme | 139 |
| 2.6 | Formale Beschreibung von Programmiersprachen | 139 |
| 2.6.1 | Lexikalische Regeln | 140 |

| | | |
|---------|---|-----|
| 2.6.2 | Syntaktische Regeln..... | 141 |
| 2.6.3 | Semantische Regeln..... | 144 |
| 2.7 | Erweiterung der Kernsprache | 144 |
| 2.7.1 | Bedingte Anweisung..... | 145 |
| 2.7.2 | Fallunterscheidung..... | 146 |
| 2.7.3 | do-Schleife..... | 147 |
| 2.7.4 | Allgemeinere Schleifenkonstrukte..... | 149 |
| 2.7.5 | Die for-Schleife..... | 149 |
| 2.7.6 | Arrays - indizierte Variablen | 151 |
| 2.8 | Rekursive Funktionen und Prozeduren..... | 153 |
| 2.8.1 | Rekursive Programme..... | 154 |
| 2.8.2 | Die Türme von Hanoi | 155 |
| 2.8.3 | Spielstrategien als rekursive Prädikate – Backtracking..... | 157 |
| 2.8.4 | Wechselseitige Rekursion..... | 158 |
| 2.8.5 | Induktion – Rekursion | 158 |
| 2.8.6 | Allgemeine Rekursion | 159 |
| 2.8.7 | Endrekursion..... | 161 |
| 2.8.8 | Lineare Rekursion..... | 162 |
| 2.8.9 | Eine Programmtransformation..... | 164 |
| 2.9 | Typen, Module, Klassen und Objekte | 165 |
| 2.9.1 | Strukturiertes Programmieren..... | 167 |
| 2.9.2 | Blockstrukturierung | 167 |
| 2.9.3 | Strukturierung der Daten | 168 |
| 2.9.4 | Objektorientierte Konstruktion neuer Datentypen..... | 173 |
| 2.9.5 | Modulares Programmieren | 175 |
| 2.9.6 | Schnittstellen – Interfaces..... | 176 |
| 2.9.7 | Objektorientiertes Programmieren..... | 179 |
| 2.9.8 | Vererbung | 181 |
| 2.9.9 | Summentypen in objektorientierten Sprachen..... | 183 |
| 2.9.10 | Datenkapselung..... | 185 |
| 2.10 | Verifikation..... | 187 |
| 2.10.1 | Vermeidung von Fehlern..... | 188 |
| 2.10.2 | Zwischenbehauptungen | 188 |
| 2.10.3 | Partielle Korrektheit..... | 189 |
| 2.10.4 | Zerlegung durch Zwischenbehauptungen..... | 190 |
| 2.10.5 | Zuweisungsregel..... | 191 |
| 2.10.6 | Rückwärtsbeweis..... | 193 |
| 2.10.7 | if-else-Regel..... | 195 |
| 2.10.8 | Abschwächungsregel und einarmige Alternative | 195 |
| 2.10.9 | Invarianten und while-Regel..... | 196 |
| 2.10.10 | Starke und schwache Invarianten | 199 |
| 2.10.11 | Programm-Verifizierer..... | 201 |
| 2.10.12 | do-Schleife..... | 203 |
| 2.10.13 | Terminierung | 204 |

| | |
|--|------------|
| 2.10.14 Beweis eines Programmschemas | 204 |
| 2.11 Zusammenfassung | 205 |
| 3 Die Programmiersprache Java | 207 |
| 3.1 Geschichte von Java | 209 |
| 3.2 Die lexikalischen Elemente von Java | 209 |
| 3.2.1 Kommentare | 210 |
| 3.2.2 Bezeichner | 210 |
| 3.2.3 Schlüsselwörter | 211 |
| 3.2.4 Literale | 211 |
| 3.3 Datentypen und Methoden | 213 |
| 3.3.1 Variablen | 213 |
| 3.3.2 Referenz-Datentypen | 214 |
| 3.3.3 Arrays | 215 |
| 3.3.4 Methoden | 216 |
| 3.3.5 Klassen und Instanzen | 218 |
| 3.3.6 Objekte und Referenzen | 219 |
| 3.3.7 Objekt- und Klassenkomponenten | 220 |
| 3.3.8 Attribute | 221 |
| 3.3.9 Überladung | 223 |
| 3.3.10 Konstruktoren | 223 |
| 3.3.11 Aufzählungstypen | 225 |
| 3.4 Ausführbare Java-Programme | 226 |
| 3.4.1 Java-Dateien – Übersetzungseinheiten | 228 |
| 3.4.2 Programme | 228 |
| 3.4.3 Packages | 229 |
| 3.4.4 Standard-Packages | 231 |
| 3.5 Ausdrücke und Anweisungen | 232 |
| 3.5.1 Arithmetische Operationen | 232 |
| 3.5.2 Vergleichsoperationen | 233 |
| 3.5.3 Boolesche Operationen | 234 |
| 3.5.4 Bitweise Operationen | 234 |
| 3.5.5 Zuweisungsausdrücke | 234 |
| 3.5.6 Anweisungsausdrücke | 236 |
| 3.5.7 Sonstige Operationen | 236 |
| 3.5.8 Präzedenz der Operatoren | 237 |
| 3.5.9 Einfache Anweisungen | 238 |
| 3.5.10 Blöcke | 239 |
| 3.5.11 Alternativ-Anweisungen | 239 |
| 3.5.12 switch-Anweisung | 240 |
| 3.5.13 Schleifen | 241 |
| 3.5.14 Die for-Anweisung | 242 |
| 3.5.15 break- und continue-Anweisungen | 244 |

| | | |
|----------|--|------------|
| 3.6 | Klassen und Objekte | 245 |
| 3.6.1 | Vererbung | 246 |
| 3.6.2 | Späte Bindung (Late Binding) | 251 |
| 3.6.3 | Finale Komponenten | 252 |
| 3.6.4 | Zugriffsrechte von Feldern und Methoden | 253 |
| 3.6.5 | Attribute von Klassen | 253 |
| 3.6.6 | Abstrakte Klassen | 253 |
| 3.6.7 | Rekursiv definierte Klassen | 255 |
| 3.6.8 | Schnittstellen (Interfaces) | 257 |
| 3.6.9 | Generische Datentypen | 260 |
| 3.6.10 | Ausnahmen - Exceptions | 261 |
| 3.6.11 | Zusicherungen – Assertions | 265 |
| 3.6.12 | Threads..... | 269 |
| 3.6.13 | Producer-Consumer mit Threads | 273 |
| 3.7 | Grafische Benutzeroberflächen mit Java (AWT) | 275 |
| 3.7.1 | Ein erstes Fenster | 276 |
| 3.7.2 | Ereignisse | 277 |
| 3.7.3 | Beispiel für eine Ereignisbehandlung | 278 |
| 3.7.4 | Buttons | 280 |
| 3.7.5 | Grafikausgabe in Fenstern | 281 |
| 3.7.6 | Maus-Ereignisse..... | 282 |
| 3.7.7 | Paint | 285 |
| 3.7.8 | Weitere Bedienelemente von Programmen und Fenstern..... | 287 |
| 3.8 | Dateien: Ein- und Ausgabe | 288 |
| 3.8.1 | Dateidialog..... | 288 |
| 3.8.2 | Schreiben einer Datei..... | 289 |
| 3.8.3 | Lesen einer Datei | 290 |
| 3.8.4 | Testen von Dateieigenschaften | 291 |
| 3.9 | Wurzeln und Weiterentwicklung von Java | 291 |
| 4 | Algorithmen und Datenstrukturen | 293 |
| 4.1 | Suchalgorithmen | 295 |
| 4.1.1 | Lineare Suche | 296 |
| 4.1.2 | Binäre Suche | 297 |
| 4.1.3 | Lineare Suche vs. binäre Suche | 299 |
| 4.1.4 | Komplexität von Algorithmen | 300 |
| 4.2 | Einfache Sortierverfahren | 302 |
| 4.2.1 | Datensätze und Schlüssel..... | 302 |
| 4.2.2 | Invarianten und Assertions | 305 |
| 4.2.3 | BubbleSort | 307 |
| 4.2.4 | SelectionSort..... | 309 |
| 4.2.5 | InsertionSort..... | 311 |
| 4.2.6 | Laufzeitvergleiche der einfachen Sortieralgorithmen..... | 313 |

| | | |
|--------|--|-----|
| 4.2.7 | ShellSort und CombSort | 314 |
| 4.3 | Schnelle Sortieralgorithmen | 315 |
| 4.3.1 | Divide and Conquer – teile und herrsche | 315 |
| 4.3.2 | QuickSort | 316 |
| 4.3.3 | Die Partitionierung | 317 |
| 4.3.4 | Korrektheit von QuickSort | 318 |
| 4.3.5 | Komplexität von QuickSort | 319 |
| 4.3.6 | MergeSort | 320 |
| 4.3.7 | DistributionSort (RadixSort) | 322 |
| 4.3.8 | Wieso und wie gut funktioniert DistributionSort? | 323 |
| 4.3.9 | Einsatz und Implementierung von DistributionSort | 324 |
| 4.3.10 | Laufzeit der schnellen Sortieralgorithmen | 326 |
| 4.3.11 | Externes Sortieren | 328 |
| 4.4 | Abstrakte Datenstrukturen | 329 |
| 4.4.1 | Datenstruktur = Menge + Operationen | 329 |
| 4.4.2 | Die axiomatische Methode | 330 |
| 4.5 | Stacks | 331 |
| 4.5.1 | Stackoperationen | 331 |
| 4.5.2 | Implementierung durch ein Array | 333 |
| 4.5.3 | Implementierung durch eine Liste | 334 |
| 4.5.4 | Auswertung von Postfix-Ausdrücken | 336 |
| 4.5.5 | Entkursionierung | 336 |
| 4.5.6 | Stackpaare | 337 |
| 4.6 | Queues, Puffer, Warteschlangen | 339 |
| 4.6.1 | Implementierung durch ein „zirkuläres“ Array | 339 |
| 4.6.2 | Implementierung durch eine zirkuläre Liste | 341 |
| 4.6.3 | Anwendung von Puffern | 341 |
| 4.7 | Listen | 342 |
| 4.7.1 | Einfach verkettete Listen | 343 |
| 4.7.2 | Der Listeniterator forEach | 346 |
| 4.7.3 | Listen als Verallgemeinerung von Stacks und Queues | 347 |
| 4.7.4 | Doppelt verkettete Listen | 348 |
| 4.7.5 | Geordnete Listen und Skip-Listen | 348 |
| 4.7.6 | Adaptive Listen | 349 |
| 4.7.7 | Generische Listen | 349 |
| 4.7.8 | Behälter (Collections) | 350 |
| 4.8 | Bäume | 352 |
| 4.8.1 | Beispiele von Bäumen | 354 |
| 4.8.2 | Binärbäume | 354 |
| 4.8.3 | Implementierung von Binärbäumen | 355 |
| 4.8.4 | Traversierungen | 357 |
| 4.8.5 | Kenngrößen von Binärbäumen | 360 |
| 4.8.6 | Binäre Suchbäume | 361 |
| 4.8.7 | Implementierung von binären Suchbäumen | 362 |

| | | |
|----------|---|------------|
| 4.8.8 | Balancierte Bäume | 369 |
| 4.8.9 | AVL-Bäume | 370 |
| 4.8.10 | 2-3-4-Bäume | 372 |
| 4.8.11 | B-Bäume | 373 |
| 4.8.12 | Vollständige Bäume | 374 |
| 4.8.13 | Heaps | 375 |
| 4.8.14 | HeapSort | 379 |
| 4.8.15 | Priority-Queues | 380 |
| 4.8.16 | Bäume mit variabler Anzahl von Teilbäumen | 380 |
| 4.9 | Graphen | 381 |
| 4.9.1 | Wege und Zusammenhang | 382 |
| 4.9.2 | Repräsentationen von Graphen | 383 |
| 4.9.3 | Traversierungen | 385 |
| 4.9.4 | Tiefensuche und Backtracking | 386 |
| 4.9.5 | Breitensuche | 387 |
| 4.9.6 | Transitive Hülle | 388 |
| 4.9.7 | Kürzeste Wege | 389 |
| 4.9.8 | Schwere Probleme für Handlungsreisende | 392 |
| 4.9.9 | Eine Implementierung des TSP | 393 |
| 4.10 | Zeichenketten | 397 |
| 4.10.1 | Array-Implementierung | 397 |
| 4.10.2 | Nullterminierte Strings | 397 |
| 4.10.3 | Stringoperationen | 398 |
| 4.10.4 | Suchen in Zeichenketten | 399 |
| 4.10.5 | Der Boyer-Moore-Algorithmus | 399 |
| 5 | Rechnerarchitektur | 401 |
| 5.1 | Vom Transistor zum Chip | 401 |
| 5.1.1 | Chips | 402 |
| 5.1.2 | Chipherstellung | 403 |
| 5.1.3 | Kleinste Chip-Strukturen | 404 |
| 5.1.4 | Chipfläche und Anzahl der Transistoren | 405 |
| 5.1.5 | Weitere Chip-Parameter | 405 |
| 5.1.6 | Speicherbausteine | 406 |
| 5.1.7 | Logikbausteine | 407 |
| 5.1.8 | Schaltungsentwurf | 407 |
| 5.2 | Boolesche Algebra | 409 |
| 5.2.1 | Serien-parallele Schaltungen | 409 |
| 5.2.2 | Serien-parallele Schaltglieder | 410 |
| 5.2.3 | Boolesche Terme | 411 |
| 5.2.4 | Schaltfunktionen | 412 |
| 5.2.5 | Gleichungen | 413 |
| 5.2.6 | SP-Schaltungen sind monoton | 414 |
| 5.2.7 | Negation | 415 |

| | | |
|--------|---|-----|
| 5.2.8 | Boolesche Terme..... | 416 |
| 5.2.9 | Dualität..... | 417 |
| 5.2.10 | Realisierung von Schaltfunktionen..... | 417 |
| 5.2.11 | Konjunktive Normalform..... | 419 |
| 5.2.12 | Umwandlung in DNF oder KNF..... | 419 |
| 5.2.13 | Aussagenlogik..... | 420 |
| 5.2.14 | Mengenalgebra..... | 421 |
| 5.3 | Digitale Logik..... | 421 |
| 5.3.1 | Gatter mit mehreren Ausgängen..... | 426 |
| 5.3.2 | Logik-Gitter..... | 427 |
| 5.3.3 | Programmierbare Gitterbausteine..... | 429 |
| 5.3.4 | Rückgekoppelte Schaltungen..... | 430 |
| 5.3.5 | Anwendungen von Flip-Flops..... | 432 |
| 5.3.6 | Technische Schwierigkeiten..... | 433 |
| 5.3.7 | Die Konstruktion der Hardwarekomponenten..... | 434 |
| 5.3.8 | Schalter, Codierer, Decodierer..... | 434 |
| 5.3.9 | Speicherzellen..... | 435 |
| 5.3.10 | Register..... | 436 |
| 5.3.11 | Die Arithmetisch-Logische Einheit..... | 438 |
| 5.4 | Von den Schaltgliedern zur CPU..... | 442 |
| 5.4.1 | Busse..... | 443 |
| 5.4.2 | Mikrocodegesteuerte Operationen..... | 444 |
| 5.4.3 | Der Zugang zum Hauptspeicher..... | 447 |
| 5.4.4 | Der Mikrobefehlsspeicher – das ROM..... | 449 |
| 5.4.5 | Sprünge..... | 449 |
| 5.4.6 | Berechnete Sprünge..... | 450 |
| 5.4.7 | Der Adressrechner..... | 452 |
| 5.4.8 | Ein Mikroprogramm..... | 453 |
| 5.4.9 | Maschinenbefehle..... | 454 |
| 5.4.10 | Der Maschinenspracheinterpretierer..... | 456 |
| 5.4.11 | Argumente..... | 457 |
| 5.5 | Assemblerprogrammierung..... | 458 |
| 5.5.1 | Maschinensprache und Assembler..... | 458 |
| 5.5.2 | Register der 80x86-Familie..... | 459 |
| 5.5.3 | Allzweckregister und Spezialregister..... | 461 |
| 5.5.4 | Flag-Register..... | 461 |
| 5.5.5 | Arithmetische Flags..... | 463 |
| 5.5.6 | Größenvergleiche..... | 464 |
| 5.5.7 | Logische Operationen..... | 466 |
| 5.5.8 | Sprünge..... | 467 |
| 5.5.9 | Struktur eines vollständigen Assemblerprogrammes..... | 468 |
| 5.5.10 | Ein Beispielprogramm..... | 469 |
| 5.5.11 | Testen von Assemblerprogrammen..... | 471 |
| 5.5.12 | Speicheradressierung..... | 472 |

| | | |
|----------|--|------------|
| 5.5.13 | Operationen auf Speicherblöcken..... | 473 |
| 5.5.14 | Multiplikation und Division..... | 474 |
| 5.5.15 | Shift-Operationen..... | 475 |
| 5.5.16 | LOOP-Befehle | 477 |
| 5.5.17 | Der Stack..... | 477 |
| 5.5.18 | Einfache Unterprogramme..... | 478 |
| 5.5.19 | Parameterübergabe und Stack..... | 480 |
| 5.5.20 | Prozeduren und Funktionen | 481 |
| 5.5.21 | Makros | 482 |
| 5.5.22 | Assembler unter DOS | 483 |
| 5.5.23 | Assembler unter Windows..... | 485 |
| 5.6 | RISC-Architekturen | 486 |
| 5.6.1 | CISC..... | 486 |
| 5.6.2 | Von CISC zu RISC | 487 |
| 5.6.3 | RISC-Prozessoren | 488 |
| 5.6.4 | Pipelining..... | 489 |
| 5.6.5 | Superskalare Architekturen..... | 490 |
| 5.6.6 | Cache-Speicher | 491 |
| 5.6.7 | Leistungsvergleich | 491 |
| 5.6.8 | Konkrete RISC-Architekturen | 492 |
| 5.7 | Architektur der Intel-PC-Mikroprozessorfamilie | 494 |
| 5.7.1 | Adressierung | 499 |
| 5.7.2 | Die Segmentierungseinheit | 500 |
| 5.7.3 | Adressübersetzung | 501 |
| 5.7.4 | Datenstrukturen und Befehle des Pentium..... | 502 |
| 5.7.5 | MMX-Befehle..... | 503 |
| 5.7.6 | Betriebsarten des Pentium | 503 |
| 5.7.7 | Ausblick | 503 |
| 6 | Betriebssysteme | 505 |
| 6.1 | Basis-Software | 506 |
| 6.2 | Betriebsarten | 508 |
| 6.2.1 | Teilhaberbetrieb | 508 |
| 6.2.2 | Client-Server-Systeme | 508 |
| 6.3 | Verwaltung der Ressourcen | 510 |
| 6.3.1 | Dateisystem..... | 510 |
| 6.3.2 | Dateioperationen..... | 512 |
| 6.3.3 | Prozesse | 512 |
| 6.3.4 | Bestandteile eines Prozesses..... | 513 |
| 6.3.5 | Threads..... | 514 |
| 6.3.6 | Prozessverwaltung | 514 |
| 6.3.7 | Prozesskommunikation..... | 516 |
| 6.3.8 | Kritische Abschnitte – wechselseitiger Ausschluss..... | 517 |
| 6.3.9 | Semaphore und Monitore..... | 519 |

| | | |
|--------|---|-----|
| 6.3.10 | Deadlocks | 520 |
| 6.3.11 | Speicherverwaltung | 521 |
| 6.4 | Das Betriebssystem UNIX | 525 |
| 6.4.1 | Linux | 526 |
| 6.4.2 | Das UNIX-Dateisystem | 526 |
| 6.4.3 | Dateinamen | 528 |
| 6.4.4 | Dateirechte | 528 |
| 6.4.5 | Pfade | 528 |
| 6.4.6 | Special files | 530 |
| 6.4.7 | Externe Dateisysteme | 530 |
| 6.4.8 | UNIX-Shells | 530 |
| 6.4.9 | UNIX-Kommandos | 531 |
| 6.4.10 | Optionen | 532 |
| 6.4.11 | Datei-Muster | 533 |
| 6.4.12 | Standard-Input/Standard-Output | 534 |
| 6.4.13 | Dateibearbeitung | 534 |
| 6.4.14 | Reguläre Ausdrücke | 536 |
| 6.5 | UNIX-Prozesse | 537 |
| 6.5.1 | Pipes | 537 |
| 6.5.2 | Sind Pipes notwendig? | 538 |
| 6.5.3 | Prozess-Steuerung | 540 |
| 6.5.4 | Multitasking | 542 |
| 6.5.5 | UNIX-Shell-Programmierung | 543 |
| 6.5.6 | Die C-Shell | 543 |
| 6.5.7 | Kommando-Verknüpfungen | 544 |
| 6.5.8 | Variablen | 544 |
| 6.5.9 | Shell-Scripts | 546 |
| 6.5.10 | Ausführung von Shell-Scripts | 546 |
| 6.5.11 | UNIX-Kommandos und Shell-Kommandos | 547 |
| 6.5.12 | UNIX als Mehrbenutzersystem | 548 |
| 6.5.13 | Verbindung zu anderen Rechnern | 549 |
| 6.5.14 | Weltweiter Rechnerzugang | 549 |
| 6.5.15 | UNIX-Tools | 550 |
| 6.5.16 | Editoren | 551 |
| 6.5.17 | C und C++ | 552 |
| 6.5.18 | Scanner- und Parsergeneratoren | 553 |
| 6.5.19 | Projektbearbeitung | 555 |
| 6.6 | X Window System | 555 |
| 6.6.1 | Window-Manager und Terminal Emulator | 557 |
| 6.6.2 | Grafische Oberflächen | 557 |
| 6.7 | MS-DOS und MS-Windows | 558 |
| 6.7.1 | Dynamic Link Libraries | 559 |
| 6.7.2 | Object Linking and Embedding | 560 |
| 6.7.3 | Windows NT, Windows 2000 und Windows XP | 560 |

| | | |
|----------|---|------------|
| 6.7.4 | Windows XP | 562 |
| 6.8 | Alternative PC-Betriebssysteme | 562 |
| 7 | Rechnernetze | 565 |
| 7.1 | Rechner-Verbindungen | 566 |
| 7.1.1 | Signalübertragung | 566 |
| 7.1.2 | Physikalische Verbindung | 568 |
| 7.1.3 | Synchronisation | 570 |
| 7.1.4 | Bitcodierungen | 571 |
| 7.2 | Datenübertragung mit Telefonleitungen | 572 |
| 7.2.1 | ISDN | 573 |
| 7.2.2 | DSL, ADSL und T-DSL | 574 |
| 7.2.3 | ADSL2+ | 576 |
| 7.3 | Protokolle und Netze | 576 |
| 7.3.1 | Das OSI-Modell | 577 |
| 7.3.2 | Netze | 579 |
| 7.3.3 | Netztopologien | 580 |
| 7.3.4 | Netze von Netzen | 583 |
| 7.3.5 | Zugriffsverfahren | 585 |
| 7.3.6 | Wettkampfverfahren: CSMA-CD | 586 |
| 7.4 | Netztechnologien | 587 |
| 7.4.1 | Ethernet | 587 |
| 7.4.2 | FDDI | 588 |
| 7.4.3 | ATM | 589 |
| 7.4.4 | SONET/SDH | 590 |
| 7.5 | Drahtlose Netze | 592 |
| 7.5.1 | Bluetooth | 593 |
| 7.5.2 | WLAN | 594 |
| 8 | Das Internet | 601 |
| 8.0.1 | Bildung von Standards im Internet | 602 |
| 8.1 | Die TCP/IP Protokolle | 604 |
| 8.1.1 | Die Protokolle TCP und UDP | 605 |
| 8.1.2 | Das IP Protokoll | 607 |
| 8.2 | IP-Adressen | 609 |
| 8.2.1 | Adressklassen | 610 |
| 8.2.2 | Adressübersetzung | 612 |
| 8.3 | Das System der Domain-Namen | 616 |
| 8.3.1 | DNS-lookup in Java | 618 |
| 8.3.2 | Programmierung einer TCP Verbindungen | 620 |

| | | |
|----------|---|------------|
| 8.4 | Intranets, Firewalls und virtuelle private Netzwerke | 624 |
| 8.5 | Die Dienste im Internet | 626 |
| 8.5.1 | E-Mail | 627 |
| 8.5.2 | News | 631 |
| 8.5.3 | FTP | 632 |
| 8.5.4 | Telnet | 633 |
| 8.5.5 | Gopher | 634 |
| 8.6 | Das World Wide Web | 634 |
| 8.6.1 | HTTP | 636 |
| 8.6.2 | HTML | 638 |
| 8.6.3 | Die Struktur eines HTML-Dokumentes | 640 |
| 8.6.4 | Querverweise: Links | 641 |
| 8.6.5 | Tabellen und Frames | 642 |
| 8.6.6 | Formulare | 644 |
| 8.6.7 | Style Sheets | 645 |
| 8.6.8 | Weitere Möglichkeiten von HTML | 646 |
| 8.7 | Web-Programmierung | 646 |
| 8.7.1 | JavaScript | 646 |
| 8.7.2 | Applets | 649 |
| 8.7.3 | Die Struktur eines Applets | 650 |
| 8.7.4 | Der Lebenszyklus eines Applet | 651 |
| 8.7.5 | Interaktionen | 651 |
| 8.7.6 | PHP | 653 |
| 8.7.7 | XML | 656 |
| 8.7.8 | DOM, Ajax und Web 2.0 | 664 |
| 9 | Theoretische Informatik und Compilerbau | 667 |
| 9.1 | Analyse von Programmtexten | 667 |
| 9.1.1 | Lexikalische Analyse | 668 |
| 9.1.2 | Syntaxanalyse | 669 |
| 9.2 | Reguläre Sprachen | 670 |
| 9.2.1 | Reguläre Ausdrücke | 671 |
| 9.2.2 | Automaten und ihre Sprachen | 673 |
| 9.2.3 | Implementierung endlicher Automaten | 675 |
| 9.2.4 | -Transitionen und nichtdeterministische Automaten | 676 |
| 9.2.5 | Automaten für reguläre Sprachen | 676 |
| 9.2.6 | Von nichtdeterministischen zu deterministischen Automaten | 677 |
| 9.2.7 | Anwendung: flex | 678 |
| 9.3 | Kontextfreie Sprachen | 679 |
| 9.3.1 | Kontextfreie Grammatiken | 680 |
| 9.3.2 | Ableitungen | 681 |
| 9.3.3 | Stackautomaten (Kellerautomaten) | 682 |
| 9.3.4 | Stackautomaten für beliebige kontextfreie Sprachen | 684 |

| | | |
|--------|--|-----|
| 9.3.5 | Nichtdeterministische Algorithmen und Backtracking..... | 684 |
| 9.3.6 | Inhärent nichtdeterministische Sprachen..... | 686 |
| 9.3.7 | Ableitungsbaum, Syntaxbaum..... | 687 |
| 9.3.8 | Abstrakte Syntaxbäume..... | 688 |
| 9.4 | Grundlagen des Compilerbaus..... | 689 |
| 9.4.1 | Parsen durch rekursiven Abstieg (recursive descent)..... | 690 |
| 9.4.2 | LL(1)-Grammatiken..... | 691 |
| 9.4.3 | Äquivalente Grammatiken..... | 693 |
| 9.4.4 | Top-down und bottom-up..... | 694 |
| 9.4.5 | Shift-Reduce Parser..... | 696 |
| 9.4.6 | Die Arbeitsweise von Shift-Reduce-Parsern..... | 697 |
| 9.4.7 | Bottom-up Parsing..... | 698 |
| 9.4.8 | Konflikte..... | 699 |
| 9.4.9 | Ein nichtdeterministischer Automat mit Stack..... | 699 |
| 9.4.10 | Übergang zum deterministischen Automaten..... | 702 |
| 9.4.11 | Präcedenz..... | 705 |
| 9.4.12 | LR(1) und LALR(1)..... | 706 |
| 9.4.13 | Parsegeneratoren..... | 707 |
| 9.4.14 | lex/flex & yacc/bison..... | 708 |
| 9.4.15 | Grammatische Aktionen..... | 709 |
| 9.4.16 | Fehlererkennung..... | 711 |
| 9.4.17 | Synthetisierte Werte..... | 712 |
| 9.4.18 | Symoltabellen..... | 712 |
| 9.4.19 | Codeoptimierung..... | 713 |
| 9.5 | Berechenbarkeit..... | 714 |
| 9.5.1 | Berechenbare Funktionen..... | 714 |
| 9.5.2 | Beispiele berechenbarer Funktionen..... | 715 |
| 9.5.3 | Diagonalisierung..... | 717 |
| 9.5.4 | Nicht berechenbare Funktionen..... | 718 |
| 9.5.5 | Algorithmenbegriff und Churchsche These..... | 719 |
| 9.5.6 | Turingmaschinen..... | 720 |
| 9.5.7 | Turing-Post Programme..... | 722 |
| 9.5.8 | Turing-berechenbare Funktionen..... | 723 |
| 9.5.9 | Registermaschinen..... | 723 |
| 9.5.10 | GOTO-Programme..... | 724 |
| 9.5.11 | While-Programme..... | 725 |
| 9.5.12 | For-Programme (Loop-Programme)..... | 727 |
| 9.5.13 | Effiziente Algorithmen als For-Programme..... | 728 |
| 9.5.14 | Elementare (primitive) Rekursion..... | 729 |
| 9.5.15 | Allgemeine Rekursion (μ -Rekursion)..... | 730 |
| 9.5.16 | Die Ackermannfunktion..... | 731 |
| 9.5.17 | Berechenbare Funktionen – Churchsche These..... | 732 |
| 9.5.18 | Gödelisierung..... | 733 |
| 9.5.19 | Aufzählbarkeit und Entscheidbarkeit..... | 734 |

| | | |
|-----------|--|------------|
| 9.5.20 | Unlösbare Aufgaben..... | 735 |
| 9.5.21 | Semantische Probleme sind unentscheidbar | 736 |
| 9.6 | Komplexitätstheorie..... | 737 |
| 9.6.1 | Rückführung auf ja/nein-Probleme | 738 |
| 9.6.2 | Entscheidungsprobleme und Sprachen | 739 |
| 9.6.3 | Maschinenmodelle und Komplexitätsmaße | 739 |
| 9.6.4 | Sprachen und ihre Komplexität..... | 740 |
| 9.6.5 | Effiziente parallele Lösungen | 741 |
| 9.6.6 | Nichtdeterminismus | 742 |
| 9.6.7 | Die Klasse NP | 743 |
| 9.6.8 | Reduzierbarkeit | 744 |
| 9.6.9 | Der Satz von Cook | 746 |
| 9.6.10 | NP-Vollständigkeit..... | 748 |
| 9.6.11 | CLIQUE ist NP-vollständig | 749 |
| 9.6.12 | Praktische Anwendung von SAT-Problemen | 750 |
| 9.6.13 | $P = NP$? | 752 |
| 10 | Datenbanksysteme | 753 |
| 10.1 | Datenbanken und Datenbanksysteme | 753 |
| 10.2 | Datenmodelle | 755 |
| 10.2.1 | Entity/Relationship-Modell..... | 755 |
| 10.2.2 | Das Relationale Datenbankmodell | 757 |
| 10.2.3 | Relationen | 758 |
| 10.2.4 | Die relationale Algebra | 759 |
| 10.2.5 | Erweiterungen des relationalen Datenmodells..... | 759 |
| 10.2.6 | Abbildung eines E/R-Datenmodells in ein relationales Modell..... | 760 |
| 10.3 | Die Anfragesprache SQL..... | 761 |
| 10.3.1 | Datendefinition..... | 761 |
| 10.3.2 | Einfache Anfragen | 763 |
| 10.3.3 | Gruppierung und Aggregate..... | 764 |
| 10.3.4 | Verknüpfung verschiedener Relationen | 765 |
| 10.3.5 | Einfügen, Ändern und Löschen von Datensätzen | 765 |
| 10.3.6 | Mehrbenutzerbetrieb | 766 |
| 10.4 | Anwendungsprogrammierung in Java | 768 |
| 10.4.1 | Das SQL-Paket in Java..... | 769 |
| 10.4.2 | Aufbau einer Verbindung | 770 |
| 10.4.3 | Anfragen..... | 770 |
| 10.5 | Zusammenfassung | 772 |
| 11 | Grafikprogrammierung | 773 |
| 11.1 | Hardware..... | 773 |
| 11.1.1 | Auflösungen | 774 |
| 11.1.2 | Farben..... | 774 |

| | | |
|-----------|---|------------|
| 11.2 | Grafikroutinen für Rastergrafik | 775 |
| 11.2.1 | Bresenham Algorithmus | 777 |
| 11.3 | Einfache Programmierbeispiele | 778 |
| 11.3.1 | Mandelbrot- und Julia-Mengen | 780 |
| 11.3.2 | Turtle-Grafik | 784 |
| 11.3.3 | L-Systeme | 787 |
| 11.3.4 | Ausblick | 790 |
| 11.4 | 3-D-Grafikprogrammierung | 791 |
| 11.4.1 | Sichtbarkeit | 792 |
| 11.4.2 | Beleuchtungsmodelle | 793 |
| 11.4.3 | Ray-Tracing | 795 |
| 11.4.4 | Die Radiosity Methode | 796 |
| 11.4.5 | Ausblick | 797 |
| 12 | Software-Entwicklung | 799 |
| 12.1 | Methoden und Werkzeuge für Projekte | 800 |
| 12.2 | Vorgehensmodelle | 802 |
| 12.2.1 | Code and fix-Verfahren | 802 |
| 12.2.2 | Wasserfall-Modelle | 803 |
| 12.2.3 | Transformations-Modelle | 806 |
| 12.2.4 | Nichtsequentielle Vorgehensmodelle | 806 |
| 12.2.5 | Prototyping und Spiralmodelle | 807 |
| 12.2.6 | Modelle zur inkrementellen Systementwicklung | 808 |
| 12.2.7 | Evolutionäre Entwicklungsmodelle | 808 |
| 12.2.8 | Modelle zur objektorientierten Systementwicklung | 809 |
| 12.3 | Traditionelle Methoden zur Programmentwicklung | 810 |
| 12.3.1 | Strukturierte Programmierung | 811 |
| 12.3.2 | Schrittweise Verfeinerung und Top-down-Entwurf | 811 |
| 12.3.3 | Geheimnisprinzip, Daten-Abstraktion und Modularisierung | 812 |
| 12.3.4 | Strukturierte Analyse- und Entwurfstechniken | 813 |
| 12.3.5 | Entity/Relationship-Modellierung | 814 |
| 12.3.6 | Systematische Test-, Review- und Inspektionsverfahren | 815 |
| 12.4 | Objektorientierte Software-Entwicklungsmethoden | 815 |
| 12.4.1 | Prinzipien der Objektorientierung | 815 |
| 12.4.2 | Objektorientierter Entwurf | 816 |
| 12.5 | Objektorientierte Analyse und Modellierung | 817 |
| 12.5.1 | Standardisierung der objektorientierten Modellierung | 817 |
| 12.5.2 | Die Modellierungssprache UML | 818 |
| 12.5.3 | Software-Architektur | 822 |
| 12.5.4 | Entwurfsmuster und Frameworks | 823 |
| 12.5.5 | Aspekt-orientierte Entwicklung | 823 |
| 12.5.6 | Modell-getriebene Architektur | 824 |

| | | |
|----------|---|------------|
| 12.6 | Projekt-Management..... | 825 |
| 12.6.1 | Projektinitialisierung und -planung..... | 825 |
| 12.6.2 | Projektsteuerung und -koordination..... | 826 |
| 12.6.3 | Projektabschluss und -bericht | 827 |
| 12.7 | Software-Qualitätssicherung..... | 827 |
| 12.7.1 | Qualitätsnormen und Zertifizierung..... | 829 |
| 12.8 | Werkzeuge und Programmierumgebungen | 831 |
| 12.8.1 | Klassifizierung von Werkzeugen | 831 |
| 12.8.2 | Werkzeuge zur Analyse und Modellierung..... | 832 |
| 12.8.3 | Werkzeuge für Spezifikation und Entwurf | 832 |
| 12.8.4 | Programmier-Werkzeuge | 833 |
| 12.8.5 | Test- und Fehlerbehebungs-Werkzeuge..... | 833 |
| 12.8.6 | Tätigkeitsübergreifende Werkzeuge | 835 |
| 12.8.7 | Entwicklungs-Umgebungen | 836 |
| A | Literatur | 839 |
| A.1 | Einführende Bücher | 839 |
| A.2 | Lehrbücher der Informatik..... | 840 |
| A.3 | Programmieren in Pascal | 841 |
| A.4 | Programmieren in Java | 841 |
| A.5 | Algorithmen und Datenstrukturen | 842 |
| A.6 | Rechnerarchitektur..... | 843 |
| A.7 | Betriebssysteme | 844 |
| A.8 | Rechnernetze..... | 844 |
| A.9 | Internet..... | 845 |
| A.10 | Theoretische Informatik und Compilerbau..... | 846 |
| A.11 | Datenbanken | 847 |
| A.12 | Grafikprogrammierung | 848 |
| A.13 | Software-Entwicklung | 849 |
| A.14 | Mathematischer Hintergrund | 852 |
| A.15 | Sonstiges | 852 |
| | Stichwortverzeichnis | 853 |
| | Stichwortverzeichnis | 877 |

