

# Inhalt

<b>LE 1</b>	<b>1</b>	<b>Einführung – Computersysteme und Informatik</b>	<b>1</b>
	1	Einführung	2
	1.1	Aufbau und Funktionsweise eines Computersystems	3
	1.1.1	Die Zentraleinheit	5
	1.1.2	Bildschirm, Tastatur und Maus	8
	1.1.3	Weitere Ein- und Ausgabegeräte	12
	1.1.4	Externe Speicher	13
	1.1.5	Vernetzung	14
	1.1.6	Das Betriebssystem	17
	1.1.7	Fallstudie: Die Firma ProfiSoft	18
	1.2	Die Informatik	18
	1.3	Gliederung und Aufbau dieses Buches	25
<b>LE 2</b>	<b>1</b>	<b>Einführung – Internet, Web und HTML</b>	<b>35</b>
	1.4	Das Internet	36
	1.4.1	Der Aufbau des Internet	36
	1.4.2	Der Anschluss an das Internet	38
	1.4.3	Die Adressierung im Internet	39
	1.5	Dienste im Internet	40
	1.5.1	Elektronische Post im Internet (E-Mail)	41
	1.5.2	Nachrichtengruppen im Internet ( <i>newsgroups</i> )	45
	1.5.3	Plaudern im Internet ( <i>chat</i> )	48
	1.5.4	Dateien übertragen im Internet ( <i>ftp</i> )	49
	1.5.5	Computersysteme fernbedienen über das Internet ( <i>Telnet</i> )	50
	1.6	Das <i>World Wide Web</i> (WWW)	50
	1.6.1	Web-Browser	52
	1.6.2	Suchen und Finden im Internet	54
	1.7	Die Sprachen HTML und XHTML	57
	1.7.1	Grafiken und Bilder einbinden	60
	1.7.2	Erweiterungen von Web-Browsern ( <i>plug-ins</i> )	62
	1.7.3	Erstellen von XHTML-Seiten und Bereitstellen auf Web-Servern	64
	1.8	Fallstudie: Die Firma ProfiSoft im Internet	65

## Inhalt

- LE 3 2 Grundlagen der Programmierung – Einführung 71**
  - 2 Grundlagen der Programmierung 72
  - 2.1 Programm, Programmieren, Programmiersprachen 72
  - 2.2 Java-Applets und ihre Einbindung in HTML 79
  - 2.3 Compiler, Interpreter und Programmierumgebungen 84
  - 2.4 Das erste Java-Programm: »Hello World« 89
    - 2.4.1 Die erste Java-Anwendung: »Hello World« 89
    - 2.4.2 Notationen für die Syntax einer Programmiersprache 91
    - 2.4.3 Aufbau eines Java-Programms 93
    - 2.4.4 Das erste Java-Applet: »Hello World« 94
- LE 4 2 Grundlagen der Programmierung – Objekte und Klassen (Teil 1) 101**
  - 2.5 Zuerst die Theorie: Objekte und Klassen 102
    - 2.5.1 Intuitive Einführung 102
    - 2.5.2 Objekte 104
    - 2.5.3 Klassen 110
  - 2.6 Dann die Praxis: Objekte und Klassen in Java 115
    - 2.6.1 Deklaration von Klassen 115
    - 2.6.2 Visualisierung von Objekten 120
    - 2.6.3 GUI-Klassen 122
    - 2.6.4 Erzeugen und Referenzieren von Objekten 123
    - 2.6.5 Senden von Botschaften und Ausführen von Operationen 130
    - 2.6.6 Löschen von Objekten 132
- LE 5 2 Grundlagen der Programmierung – Objekte und Klassen (Teil 2) 141**
  - 2.6 Dann die Praxis: Objekte und Klassen in Java 142
    - 2.6.7 Aufbau und Start eines Applets 142
    - 2.6.8 Klassenattribute 144
    - 2.6.9 Klassenoperationen 145
    - 2.6.10 Aufbau und Start einer Java-Anwendung 147
  - 2.7 Assoziationen und Kardinalitäten 150
    - 2.7.1 Zuerst die Theorie: Assoziationen und ihre Kardinalitäten 150
    - 2.7.2 Dann die Praxis: Assoziationen in Java 153
  - 2.8 Zur Software-Technik: Integrierte Dokumentation, Verbalisierung, Schreibtischtest, CASE 159
    - 2.8.1 Prinzip der integrierten Dokumentation 160
    - 2.8.2 Prinzip der Verbalisierung 162
    - 2.8.3 Schreibtischtest und *debugging* 166
    - 2.8.4 Einsatz von CASE-Werkzeugen 169

<b>LE 6</b>	<b>2</b>	<b>Grundlagen der Programmierung – Ereignisse und Attribute</b>	<b>177</b>
	2.9	Einführung in die Ereignisverarbeitung	178
	2.9.1	Zuerst die Theorie: Das Delegations-Ereignis-Modell	178
	2.9.2	Dann die Praxis: Ereignisse und ihre Verarbeitung in Java	181
	2.10	Attribute und ihre Typen	192
	2.10.1	Zuerst die Theorie: Eigenschaften und Verhalten von Attributen	192
	2.10.2	Dann die Praxis: Attribute in Java	197
	2.10.3	Die Zuweisung	198
	2.10.4	Einfache Typen, ihre Werte und Operationen	200
	2.10.5	Ausdrücke	204
	2.10.6	Typumwandlungen	208
	2.10.7	Referenztypen	209
<b>LE 7</b>	<b>2</b>	<b>Grundlagen der Programmierung – Operationen</b>	<b>217</b>
	2.11	Operationen und ihre Parameter	218
	2.11.1	Zuerst die Theorie: Operationen, Prozeduren und Funktionen	218
	2.11.2	Dann die Praxis: Parameterlose Prozeduren in Java	224
	2.11.3	Dann die Praxis: Prozeduren mit Eingabeparametern in Java	225
	2.11.4	Dann die Praxis: Objekte als Eingabeparameter in Java	228
	2.11.5	Dann die Praxis: Prozeduren mit Ausgabeparametern	230
	2.11.6	Dann die Praxis: Funktionen in Java	231
	2.11.7	Dann die Praxis: Objekte als Ergebnisparameter in Java	232
	2.11.8	Überladen von Operationen	235
	2.11.9	Dann die Praxis: Konstruktoren in Java	236
	2.11.10	Objekte als Eingabeparameter in Konstruktoren	238
	2.11.11	Rekursion	239
	2.12	Einführung in die Grafik-Programmierung in Java	243
	2.12.1	Zeichnen von Objekten	243
	2.12.2	Bilder laden und anzeigen	248

## Inhalt

- LE 8 2 Grundlagen der Programmierung – Kontrollstrukturen 259**
  - 2.13 Kontrollstrukturen 260
    - 2.13.1 Die Sequenz 261
    - 2.13.2 Die Auswahl 261
    - 2.13.3 Die Wiederholung 268
    - 2.13.4 Der Aufruf 278
    - 2.13.5 Geschachtelte Kontrollstrukturen 278
    - 2.13.6 Anordnung von Auswahlanweisungen 283
    - 2.13.7 Auswahl von Kontrollstrukturen 286
    - 2.13.8 Strukturierte Programmierung 286
    - 2.13.9 Behandlung von Ausnahmen 290
  
- LE 9 2 Grundlagen der Programmierung – Vererbung und Polymorphismus 303**
  - 2.14 Zuerst die Theorie: Vererbung 304
    - 2.14.1 Intuitive Einführung 304
    - 2.14.2 Einfach- und Mehrfachvererbung 305
    - 2.14.3 Überschreiben und Verbergen 310
    - 2.14.4 Polymorphismus 311
  - 2.15 Dann die Praxis: Vererbung in Java 314
    - 2.15.1 Die Java-Syntax und -Semantik der Vererbung 314
    - 2.15.2 Java-Beispiel für die Vererbung 316
    - 2.15.3 Java-Beispiel für das Überschreiben und Verbergen 326
    - 2.15.4 Sonderfälle 329
  
- LE 10 2 Grundlagen der Programmierung – Schnittstellen, Pakete, Ereignisse 337**
  - 2.16 Schnittstellen 338
    - 2.16.1 Zuerst die Theorie: Schnittstellen in der Software-Entwicklung 338
    - 2.16.2 Dann die Praxis:
      - Das Java-Schnittstellenkonzept 338
    - 2.16.3 Dann die Praxis: Die Java-Syntax und -Semantik für Schnittstellen 342
    - 2.16.4 Schnittstellen und Polymorphismus 343
    - 2.16.5 Konstanten in Schnittstellen 345
    - 2.16.6 Leere Implementierung von Schnittstellen 346
    - 2.16.7 Schnittstellen und Vererbung 348
  - 2.17 Pakete 349
    - 2.17.1 Zuerst die Theorie: Pakete in der Software-Entwicklung 349
    - 2.17.2 Dann die Praxis: Pakete in Java 350
      - 2.17.2.1 Import von Paketen 350

- 2.17.2.2 Zugriffsrechte und Sichtbarkeit 354
- 2.18 Die Java-Ereignisverarbeitung im Detail 356
  - 2.18.1 Ereignistypen 356
  - 2.18.2 Ereignisabhörer 356
  - 2.18.3 Adapterklassen 359
  - 2.18.4 Registrierung 360
  - 2.18.5 Vorgehensweise 360
  - 2.18.6 Beispiel: Ein einfacher UML-Editor 363
  - 2.18.7 Anonyme Klassen 372

**LE 11 2 Grundlagen der Programmierung – Datenstrukturen 377**

- 2.19 Datenstrukturen 378
  - 2.19.1 Felder 380
    - 2.19.1.1 Erzeugen und Benutzen von Feldern 381
    - 2.19.1.2 Die Java-Syntax und -Semantik für Felder 382
    - 2.19.1.3 Felder als Container 387
    - 2.19.1.4 Das *Singleton*-Muster 391
  - 2.19.2 Die Klasse Vector 394
  - 2.19.3 Iteratoren 399
    - 2.19.3.1 Zuerst die Theorie: Das Iterator-Muster 399
    - 2.19.3.2 Dann die Praxis: Iteratoren in Java 400
  - 2.19.4 Die Hüllklassen für einfache Typen 405
  - 2.19.5 Die *String*-Klassen von Java 408
    - 2.19.5.1 Die Klasse String 408
    - 2.19.5.2 Die Klasse StringBuffer 410
    - 2.19.5.3 Die Klasse StringTokenizer 411
  - 2.19.6 Die Klasse Stack 414
  - 2.19.7 Die Klasse Hashtable 418
  - 2.19.8 Aufzählungen mit enum 422

**LE 12 2 Grundlagen der Programmierung – Persistenz 431**

- 2.20 Persistenz und Datenhaltung 432
  - 2.20.1 Persistenz und Datenhaltung in Java 432
  - 2.20.2 Dateiströme 436
  - 2.20.3 Filterströme 440
  - 2.20.4 Eine einfache Indexverwaltung 441
    - 2.20.4.1 Zuerst die Theorie: Dateioorganisation 441
    - 2.20.4.2 Dann die Praxis: Direktzugriffsspeicher in Java 443
  - 2.20.5 Drei-Schichten-Architektur 448
  - 2.20.6 Die Serialisierung von Objekten 456

## Inhalt

- LE 13 3      **Algorithmik und Software-Technik –  
Algorithmen und ihre Verifikation** 469**
- 3      Algorithmik und Software-Technik 470
  - 3.1      Algorithmen und ihre Eigenschaften 471
  - 3.2      Verifikation 476
    - 3.2.1      Intuitive Einführung 476
    - 3.2.2      Zusicherungen 479
    - 3.2.3      Spezifizieren mit Anfangs- und  
Endebedingung 481
    - 3.2.4      Anfangs-/Endebedingungen und Zusicherungen  
in Java 483
    - 3.2.5      Verifikationsregeln 486
    - 3.2.6      Termination von Schleifen 492
    - 3.2.7      Entwickeln von Schleifen 493
- LE 14 3      **Algorithmik und Software-Technik –  
Testen von Programmen** 503**
- 3.3      Testen von Programmen 504
    - 3.3.1      Einführung und Überblick 504
    - 3.3.2      Klassifikation von Testverfahren 511
    - 3.3.3      Das Zweigüberdeckungstestverfahren 512
    - 3.3.4      Die Bedingungsüberdeckungstestverfahren 514
    - 3.3.5      Funktionale Testverfahren 517
    - 3.3.6      Funktionale Äquivalenzklassenbildung 518
    - 3.3.7      Grenzwertanalyse 522
    - 3.3.8      Test spezieller Werte 525
    - 3.3.9      Kombiniertes Funktions- und Strukturtest 526
    - 3.3.10      Testen von Klassen und ihren Unterklassen 528
    - 3.3.11      Testgetriebenes Programmieren 533
- LE 15 3      **Algorithmik und Software-Technik –  
Überprüfung von Dokumenten und  
Verbesserung des Prozesses** 547**
- 3.4      Überprüfung von Dokumenten 548
    - 3.4.1      *Walkthrough* 550
    - 3.4.2      Inspektion 552
  - 3.5      Verbesserung des Entwicklungsprozesses 561
    - 3.5.1      Zur Psychologie des Programmierens 562
    - 3.5.2      Selbstkontrolliertes Programmieren 566
- LE 16 3      **Algorithmik und Software-Technik –  
Aufwand von Algorithmen** 575**
- 3.6      Aufwand von Algorithmen 576
    - 3.6.1      Einführung und Überblick 576
    - 3.6.2      Beispiel einer Aufwandsberechnung: Mischen 577

3.6.3	Beispiel einer Aufwandsberechnung mit Aufruf von Operationen: Mischsortieren	584
3.6.4	Beispiel einer Aufwandsberechnung bei rekursiven Programmen: Türme von Hanoi	592
3.6.5	Beispiel einer Aufwandsabschätzung: Berechnung von $\pi$	596
<b>LE 17</b>	<b>3</b>	<b>Algorithmik und Software-Technik – Listen und Bäume</b>
		601
3.7	Listen	602
3.7.1	Zuerst die Theorie: Attribute und Operationen einer verketteten Liste	603
3.7.2	Dann die Praxis: Verkettete Listen in Java	605
3.7.3	Beispiel: Verwaltung eines Lexikons	612
3.7.4	Klassifikation von Listen	617
3.8	Bäume	619
3.8.1	Zuerst die Theorie: Strukturelle Eigenschaften und Begriffe binärer Bäume	620
3.8.2	Zuerst die Theorie: Attribute und Operationen eines binären Suchbaums	622
3.8.3	Dann die Praxis: Binäre Suchbäume in Java	627
3.8.4	Ausblick: Allgemeine Bäume und gerichtete Graphen	631
<b>LE 18</b>	<b>3</b>	<b>Algorithmik und Software-Technik – Suchen &amp; Sortieren sowie Generische Typen</b>
		637
3.9	Suchen	638
3.9.1	Elementare Suchverfahren	638
3.9.2	Schlüssel-Transformationen	641
3.9.3	Suchen in Texten	642
3.10	Sortieren	643
3.10.1	Schnelle Sortierverfahren	644
3.10.2	Klassifikation von Sortierverfahren	650
3.11	Generische Typen	650
3.11.1	Generische Operationen	652
3.11.2	Generische Klassen	655
<b>LE 19</b>	<b>4</b>	<b>Anwendungen – Dialoggestaltung</b>
		665
4	Anwendungen	666
4.1	Zuerst die Theorie: Software-Ergonomie	667
4.2	Zuerst die Theorie: Zur Dialoggestaltung	668
4.2.1	Fenster	670
4.2.2	Menüs	673
4.2.3	Gestaltungs- und Bewertungskriterien für den Dialog	676

## **Inhalt**

- 4.3 Dann die Praxis: Dialog-Programmierung in Java 678
  - 4.3.1 Java-Fenster 679
  - 4.3.2 Java-Menüs 694
- 4.4 Dann die Praxis: Entwicklung der Dialog-Schnittstelle 701
  - 4.4.1 Von der Skizze zum Prototyp 701
  - 4.4.2 Anbindung des Fachkonzepts: Das MVC-Muster 706
- LE 20 4      **Anwendungen – E/A-Gestaltung 717****
  - 4.5 Zuerst die Theorie: Zur E/A-Gestaltung 718
    - 4.5.1 Interaktionselemente 718
    - 4.5.2 Gestaltung von Fenstern 725
  - 4.6 Dann die Praxis: E/A-Programmierung in Java 731
    - 4.6.1 Java-Interaktionselemente 732
    - 4.6.2 Das MVC-Muster am Beispiel einer Java-Tabelle 738
    - 4.6.3 *Layout-Manager* 747
- LE 21 4      **Anwendungen –  
kaufmännisch/technisch 757****
  - 4.7 Kaufmännisch/administrative Anwendungen 758
    - 4.7.1 Vom OOA-Modell zur Benutzungsoberfläche 758
    - 4.7.2 Anbindung der Benutzungsoberfläche an das Fachkonzept 765
    - 4.7.3 Fallbeispiel: Artikel- und Lieferantenverwaltung 771
    - 4.7.4 Das JANUS-System: Automatisierte Generierung von Benutzungsoberflächen 773
    - 4.7.5 Automatisierte Generierung von Anwendungen: Das JANUS-System und die MDA 774
  - 4.8 Technische Anwendungen 778
    - 4.8.1 Fallbeispiel: Wetterstation 778
- LE 22 4      **Anwendungen – Grafik/Multimedia 787****
  - 4.9 Grafik-Anwendungen 788
    - 4.9.1 Zuerst die Theorie: Geometrische Formen und ihre affinen Transformationen 788
    - 4.9.2 Dann die Praxis: Affine Transformationen in Java 793
    - 4.9.3 Doppelpufferung 799
    - 4.9.4 Fallbeispiel: Roboteranimation 801
  - 4.10 Multimedia-Anwendungen 809
    - 4.10.1 Zuerst die Theorie: Nebenläufigkeit durch *threads* 809

4.10.2	Dann die Praxis: <i>Threads</i> in Java	811
4.10.3	Fallbeispiel: Pool-Billard	815
<b>LE 23</b>	<b>5</b>	<b>Ausblicke – Einführung in C++</b>
	5	Ausblicke
	5.1	Einführung in C++
	5.2	Klassen in C++
	5.3	Trennung Schnittstelle – Implementierung
	5.4	Dynamische und statische Objekte
	5.5	Vererbung und Polymorphismus
	5.6	Klassenattribute und Klassenoperationen
	5.7	Generische Typen ( <i>templates</i> )
<b>LE 24</b>	<b>5</b>	<b>Ausblicke – Einführung in C++ (2. Teil) und C#</b>
	5.8	Die Standardbibliothek STL
	5.8.1	Die Klasse <i>queue</i>
	5.8.2	Die Klasse <i>string</i>
	5.8.3	Die Stromklassen
	5.8.4	Dateizugriff
	5.9	Einführung in C#
	5.9.1	Ein Überblick über .Net
	5.9.2	Ein Überblick über C#
	5.9.3	Einfache und strukturierte Typen
	5.9.4	Klassen
<b>Anhang A</b>		<b>Checklisten, Richtlinien, Erstellungsregeln</b>
		<b>Für alle Produkte</b>
		A-1 Richtlinie zur Versionskennzeichnung
		A-2 Bezeichner-Richtlinien
		A-3 Checkliste Klassen
		A-4 Checkliste Attribute
		A-5 Checkliste Operationen
		A-6 Checkliste Vererbung
		A-7 Checkliste Assoziationen
		<b>Für die Definitions- und Entwurfsphase</b>
		A-8 Checkliste OOA-Modell
		A-9 Checkliste OOD-Modell
		<b>Für die Implementierungsphase</b>
		A-10 Checkliste Programmierfehler
		A-11 Java-Formatierungs-Richtlinien
		A-12 Java-Dokumentations-Richtlinien

## **Inhalt**

### **Erstellungsregeln**

A-13 Von UML-Klassen zu Java-Klassen 905

A-14 Java-Dokumentations-Richtlinien 906

A-15 Von UML-Klassen zu JAVA-Klassen 907

## **Anhang B**

**Prozessverbesserung und ihre Formulare 908**

### **Inspektion**

B-1 Beispiel eines Inspektionsprotokolls 908

B-2 Formular zur Erfassung von

Inspektionsdaten 909

### **Formular zum Entwicklungsprozess**

B-3 Formular für ein Fehlerbuch 910

## **Anhang C**

**Praktika 911**

Praktikum 1 (nach der Lehreinheit 3) 911

Praktikum 2 (nach der Lehreinheit 8) 911

Praktikum 3 (nach der Lehreinheit 12) 912

Praktikum 4 (nach der Lehreinheit 15) 912

**Verzeichnis der Hervorhebungsboxen 913**

**Verzeichnis der Programme 914**

**Namens- und Organisationsindex 915**

**Sachindex 917**