



Grundlagen der Informatik

**Bruno Lurz
Helmut Herold
Martin Lurz
Jürgen Wohlrab**

Jetzt registrieren & **besser lernen**

Mit Pearson MyLab zu mehr Lernerfolg

Die interaktive eLearning-Plattform Pearson MyLab erweitert unsere Lehrbücher um die digitale Welt. Selbst komplexe Inhalte werden so anschaulicher und leichter verständlich. Über die Theorie hinaus können Sie das Erlernete praktisch anwenden und unmittelbar erleben.

■ **Lernen wo und wann immer Sie wollen**

mit Ihrem persönlichen Lehrbuch als kommentierbaren eText.

■ **Prüfungen effizient vorbereiten**

mit vielzähligen Übungsaufgaben inklusive Lösungshinweisen und sofortigem Feedback.

■ **Komplexe Inhalte leichter verstehen**

dank interaktiver Zusätze wie z.B. Videos, interaktive Grafiken o.ä.

■ **Sie sind Dozent*in**

und möchten Zugang zu exklusiven Materialien für die Lehre bzw. MyLab in Ihrem Kurs einsetzen? Wenden Sie sich bitte an unsere Dozentenberater*innen und fordern Sie ihren persönlichen Zugang an.

<https://www.pearson.de/studium/dozierende/>



Grundlagen der Informatik

Inhaltsverzeichnis

Grundlagen der Informatik

Zugangscode

Inhaltsverzeichnis

Vorwort

Kapitel 1: Einleitung

1.1 Idee dieses Buches

1.2 Beispiele, Übungen und Rätsel

1.3 Begleitmaterial zu diesem Buch

1.4 MyLab Grundlagen der Informatik

1.5 Danksagung

1.6 Hinweis in eigener Sache

Teil I: Einführung in die Informatik

Kapitel 2: Die Historie und die Teilgebiete der Informatik

2.1 Rätsel: Streichholzprobleme

2.2 Der Begriff Informatik

2.3 Historische Entwicklung der Informatik

2.3.1 Der Abakus

2.3.2 Der Begriff Algorithmus und Ibn Musa Al-Chwarismi

2.3.3 Wichtige Stationen von 1500 bis 1930

2.3.4 Konrad Zuse und der erste funktionstüchtige Computer

2.3.5 Howard H. Aiken und die Mark I

2.3.6 John von Neumann

2.3.7 Generationen der elektronischen Datenverarbeitung

2.4 Einordnung und Einteilung der Informatik

2.4.1 Verschiedene Einsatzgebiete von Computern (Informatik)

2.4.2 Die Teilgebiete der Informatik

Inhaltsverzeichnis

2.4.3 Die Informatik und unsere Abhängigkeit von ihr

Kapitel 3: Speicherung und Interpretation von Information

3.1 Rätsel: Umfüllprobleme

3.2 Unterschiedliche Zahlensysteme

3.2.1 Das römische Zahlensystem

3.2.2 Positionssysteme

3.2.3 Positionssysteme bei natürlichen Zahlen

3.2.4 Positionssysteme bei gebrochenen Zahlen

3.3 Dual-, Oktal- und Hexadezimalsystem

3.3.1 Das Dualsystem und das Bit im Rechner

3.3.2 Konvertieren zwischen Dual- und Oktalsystem

3.3.3 Konvertieren zwischen Dual- und Hexadezimalsystem

3.4 Konvertierungsalgorithmen

3.4.1 Konvertieren von anderen Systemen in das Dezimalsystem

3.4.2 Konvertieren vom Dezimalsystem in andere Positionssysteme

3.4.3 Konvertieren echt gebrochener Zahlen

3.4.4 Konvertieren unecht gebrochener Zahlen

3.5 Rechenoperationen im Dualsystem

3.5.1 Addition

3.5.2 Subtraktion und Darstellung negativer Zahlen

3.5.3 Multiplikation und Division

3.5.4 Konvertieren durch sukzessive Multiplikation und Addition

3.6 Reelle Zahlen

3.6.1 Festpunktzahlen

3.6.2 Gleitpunktzahlen und das IEEE-Format

3.7 Codes zur Darstellung von Zeichen

3.7.1 ASCII-Code

3.7.2 Unicode

3.8 Weitere Codes für Zahlen und Zeichen

3.8.1 BCD-Code für Zahlen

3.8.2 Gray-Code

3.8.3 Barcode

3.9 Duale Größenangaben

Inhaltsverzeichnis

3.10 Die Grunddatentypen in der Programmiersprache C/C++

Kapitel 4: Boolesche Algebra

4.1 Rätsel: Analytische Rätsel (1)

4.2 George Boole und seine Algebra mit nur zwei Werten

4.3 Operatoren

4.4 Boolesche Schaltungen

4.5 Boolesche Rechenregeln

4.6 Funktionen

Kapitel 5: Hardwarekomponenten eines Computers

5.1 Rätsel: Analytische Rätsel (2)

5.2 Aufbau von Computersystemen

5.2.1 Zentraleinheit und Peripheriegeräte

5.2.2 EVA und das von-Neumann'sche-Rechnermodell

5.3 Die heutigen Personal Computer (PCs)

5.4 Die Zentraleinheit

5.4.1 Der Prozessor

5.4.2 Der Arbeitsspeicher

5.4.3 ROMs zur Speicherung von Programmen und konstanten Daten

5.4.4 Das BIOS

5.4.5 Busse und Schnittstellen (Anschlüsse)

5.5 Die Peripherie

5.5.1 Massenspeicher

5.5.2 Eingabegeräte

5.5.3 Ausgabegeräte

5.6 Modell eines einfachen Prozessorsystems

5.7 Alternative Rechnerarchitekturen (Neuronale Netze)

Kapitel 6: Vom Programm zum Maschinenprogramm

6.1 Rätsel: Analytische Rätsel (3)

6.2 Entwicklung eines Programms

6.3 Programmierwerkzeuge

6.3.1 Unterschiedliche Arten der Übersetzung

6.3.2 Der Compiler

Inhaltsverzeichnis

6.3.3 Der Linker

6.3.4 Der Lader (und Locator)

6.3.5 Der Debugger

Teil II: Praktische Informatik

Kapitel 7: Programmiersprachen

7.1 Rätsel: Analytische Rätsel (4)

7.2 Höhere Programmiersprachen

7.3 Grundlagen der Programmierung

7.3.1 Spezifikation einer Aufgabenstellung

7.3.2 Der Begriff Algorithmus

7.3.3 Formulierung und Darstellung eines Algorithmus

7.3.4 Programm = Daten + Algorithmus

7.4 Datentypen und Operatoren in C/C++ und Java

7.4.1 Datentypen und Konstanten

7.4.2 Bezeichner

7.4.3 Grundlegende Operatoren

7.4.4 Die logischen Operatoren &&, || und !

7.4.5 Die Shift-Operatoren << und >>

7.4.6 Die Bit-Operatoren &, |, Kapitel 94:und

7.4.7 Die Postfix- und Präfixoperatoren ++ und --

7.4.8 Prioritäten und Assoziativitäten der Operatoren

7.5 Formulierung von Algorithmen in C/C++ und Java

7.5.1 Sequenz

7.5.2 Verzweigungen mit if

7.5.3 Verzweigungen mit switch

7.5.4 for-Schleife (Schleife mit der Abfrage am Anfang)

7.5.5 while-Schleife (Schleife mit der Abfrage am Anfang)

7.5.6 dowhile-Schleife (Schleife mit der Abfrage am Ende)

7.5.7 Abbruch von Schleifen mit break

7.5.8 Abbruch eines einzelnen Schleifendurchlaufs mit continue

7.5.9 Abbruch mehrerer geschachtelter Schleifen mit goto

7.5.10 Programmabbruch mit exit

7.5.11 Allgemeines zu Funktionen bzw. Methoden

7.5.12 Rekursive Funktionen bzw. rekursive Methoden

Inhaltsverzeichnis

- 7.5.13 Arrays
- 7.5.14 Strings
- 7.5.15 Zufallszahlen
- 7.5.16 Argumente auf der Kommandozeile
- 7.5.17 Ausnahmen (Exceptions) in Java
- 7.5.18 Dateien
- 7.5.19 Strukturen in C/C++

7.6 Objektorientierte Programmierung mit Java

- 7.6.1 Meilensteine in der Softwareentwicklung
- 7.6.2 Einführung in die Objektorientierung
- 7.6.3 Klassen und Objekte
- 7.6.4 Konstruktoren
- 7.6.5 Vererbung und Polymorphismus
- 7.6.6 GUI-Programmierung in Java

7.7 Programmierung mit Python

- 7.7.1 Grundlagen
- 7.7.2 Einfache Datentypen
- 7.7.3 Strukturierte Datentypen (Datenstrukturen)
- 7.7.4 Operatoren
- 7.7.5 Formulierung von Algorithmen in Python
- 7.7.6 Dateien
- 7.7.7 Funktionen und Methoden
- 7.7.8 Klassen und Objekte

7.8 Portable GUI-Programmierung mit Qt

- 7.8.1 Allgemeines zu Qt
- 7.8.2 Grundlegende Konzepte und Konstrukte von Qt
- 7.8.3 Das Signal-Slot-Konzept von Qt

7.9 Programmierung paralleler Abläufe (Parallel-Programmierung)

- 7.9.1 Konzepte und HW-Architekturen für parallele Abläufe
- 7.9.2 SW-Konzepte und Erstellung paralleler Programme
- 7.9.3 Parallele Programmierung mit Threads
- 7.9.4 Parallele Programmierung mit openMP
- 7.9.5 Besondere Probleme bei paralleler Bearbeitung
- 7.9.6 Ausblick

7.10 Funktionale Programmierung (Scala, F#)

Inhaltsverzeichnis

Kapitel 8: Datenstrukturen und Algorithmen

8.1 Rätsel: Analytische Rätsel (5)

8.2 Grundlegende Datenstrukturen

8.2.1 Allgemeine Eigenschaften von Daten

8.2.2 Basis-Datentypen

8.2.3 Datenstruktur = Daten + Operationen

8.2.4 Verkettete Listen

8.2.5 Binäre Suche in einfach verketteten Listen (Ski listen)

8.2.6 Stack (Stapel)

8.2.7 Queue (Warteschlange)

8.3 Bäume

8.3.1 Grundlegendes zu Bäumen

8.3.2 Binäre Bäume

8.3.3 Sich selbst balancierende Binärbäume

8.3.4 Splay-Bäume

8.3.5 B-Bäume

8.3.6 Baumrekursion bei Bäumen mit mehr als zwei Zweigen

8.4 Komplexität von Algorithmen und O-Notation

8.4.1 Zeitaufwand

8.4.2 Speicherplatzbedarf

8.4.3 Klassifikation von Algorithmen

8.4.4 Die O-Notation

8.4.5 Wahl eines Algorithmus

8.4.6 Einfache Optimierungen bei der Implementierung

8.5 Elementare Sortieralgorithmen

8.5.1 Grundsätzliches zu Sortieralgorithmen

8.5.2 Bubble-Sort

8.5.3 Insert-Sort

8.5.4 Select-Sort

8.5.5 Zeitmessungen für Bubble-, Insert- und Select-Sort

8.5.6 Distribution Count-Sort (Bucket-Sort)

8.6 Shell-Sort

8.7 Quicksort

8.8 Mergesort

Inhaltsverzeichnis

- 8.8.1 Rekursiver Mergesort für Arrays
- 8.8.2 Nicht-rekursiver Mergesort für Arrays
- 8.8.3 Analyse des Mergesort
- 8.8.4 Mischen von zwei sortierten Arrays

8.9 Backtracking

- 8.9.1 Finden in einem Labyrinth
- 8.9.2 Das Achtdamen-Problem
- 8.9.3 Rekursives Füllen von Figuren
- 8.9.4 Sudoku
- 8.9.5 Branch-and-Bound-Verfahren

Kapitel 9: Betriebssysteme

- 9.1 Rätsel: Überquerung einer Hängebrücke
- 9.2 Der Begriff Betriebssystem
- 9.3 Die Geschichte von Betriebssystemen
- 9.4 Grundaufgaben von Betriebssystemen
- 9.5 Aufbau und Dienste von Betriebssystemen
 - 9.5.1 Schichtenaufbau
 - 9.5.2 Prozesse, Threads, Scheduling
 - 9.5.3 Synchronisationsmechanismen
 - 9.5.4 Zeitdienste (Timer)
 - 9.5.5 Speicherverwaltung
 - 9.5.6 Dateiverwaltung und Dateisysteme
 - 9.5.7 Geräteverwaltung und Treiber
 - 9.5.8 Benutzerschnittstelle (Kommandozeile bzw. GUI)
 - 9.5.9 Programmierschnittstelle (API)
- 9.6 Besonderheiten bei Embedded Systems

Kapitel 10: Rechnernetze und das Internet

- 10.1 Rätsel: Synthetische Rätsel (1)
- 10.2 Grundlagen der Vernetzung von Rechnern
- 10.3 Das ISO/OSI-Modell und Internet-Protokolle
- 10.4 Internet-Protokolle in Rechnernetzen
 - 10.4.1 Grundbegriffe zu TCP/IP-Netzen
 - 10.4.2 TCP/IP-Protokolle

Inhaltsverzeichnis

- 10.5 Hubs, Switches, Router und Gateways
- 10.6 Grundlagen der Socket-Programmierung
- 10.7 Verteilte Anwendungen
- 10.8 Das World Wide Web (WWW)
 - 10.8.1 Komponenten und Konzepte des WWW
 - 10.8.2 Einführung in HTML
 - 10.8.3 Einführung in CSS
 - 10.8.4 Datenstrukturen zum Informationsaustausch
 - 10.8.5 Clientseitige Web-Programmierung mit JavaScript
 - 10.8.6 Serverseitige Webentwicklung
- 10.9 Gefahren durch Software
 - 10.9.1 Arten von Schadsoftware
 - 10.9.2 Pufferüberläufe (Buffer Overflows)

Kapitel 11: Datenbanksysteme

- 11.1 Rätsel: Synthetische Rätsel (2)
- 11.2 Grundlegendes zu Datenbanksystemen
 - 11.2.1 Aufgaben einer Datenbank
 - 11.2.2 Vorteile von Datenbanken
 - 11.2.3 Datenunabhängigkeit
- 11.3 Datenmodelle
 - 11.3.1 Das Entity-Relationship-Modell
 - 11.3.2 Das relationale Datenmodell
 - 11.3.3 Die relationale Algebra
- 11.4 Die Datenbanksprache SQL
 - 11.4.1 Datendefinition
 - 11.4.2 Einfügen, Ändern und Löschen von Datensätzen
 - 11.4.3 Anfragen mit select

Kapitel 12: Software Engineering

- 12.1 Rätsel: Synthetische Rätsel (3)
- 12.2 Die Software-Krise
- 12.3 Eine geeignete Software-Architektur
- 12.4 UML-Diagramme für die Modellierung
 - 12.4.1 Statische Modellierung in UML

Inhaltsverzeichnis

12.4.2 Dynamische Modellierung in UML

12.5 Modellierungsmöglichkeiten für die Software

12.6 Notwendigkeit von Prozessen

12.7 Der wichtige Prozess "84Requirement Engineering"93

12.7.1 Das UML-Anwendungsfalldiagramm (Use Case Diagram)

12.7.2 Das UML-Aktivitätsdiagramm

12.7.3 Genaue Klärung der Kundenanforderungen

12.8 Prozessmodelle

12.8.1 Schwer- und leichtgewichtige Prozessmodelle

12.8.2 Das Wasserfall-Modell

12.8.3 Das V-Modell

12.8.4 Inkrementelle und iterative Prozessmodelle

12.8.5 Agiles Vorgehen mit eXtreme Programming (XP)

12.9 Qualität eines Software-Produktes aus Kundensicht

Teil III: Technische Informatik

Kapitel 13: Transistoren, Chips und logische Bausteine

13.1 Rätsel: Synthetische Rätsel (4)

13.2 Transistoren

13.2.1 Funktionsweise und Aufbau von Transistoren

13.2.2 Realisierung boolescher Funktionen mit Transistoren

13.3 Chips

13.3.1 Geschichtliche Entwicklung

13.3.2 Herstellungsprozess

13.4 Logische Bausteine

13.4.1 Gatter

13.4.2 Decoder

13.4.3 Encoder

13.4.4 Multiplexer (Selektor)

13.4.5 Demultiplexer

Kapitel 14: Schaltnetze

14.1 Rätsel: Ein dialektisches Rätsel

14.2 Normalformen von Schaltfunktionen

14.2.1 Disjunktive Normalform (DNF)

Inhaltsverzeichnis

- 14.2.2 Konjunktive Normalform (KNF)
- 14.2.3 Allgemeines Verfahren beim Erstellen einer Schaltung
- 14.2.4 Schaltkreisrealisierung durch PLAs

14.3 Entwurf von Schaltnetzen

14.4 Minimierung logischer Ausdrücke

- 14.4.1 Karnaugh-Veitch-Diagramme (KV-Diagramme)
- 14.4.2 Don't Care Argumente
- 14.4.3 Quine-McCluskey-Verfahren

14.5 Addiernetze

- 14.5.1 Paralleladdierer
- 14.5.2 Paralleladdierer und -subtrahierer
- 14.5.3 Carry-Select-Addiernetze
- 14.5.4 Carry-Save-Addiernetze
- 14.5.5 Multiplizierer

14.6 Prinzipieller Aufbau einer ALU

Kapitel 15: Schaltwerke

15.1 Rätsel: Waldlauf, Schnapsgläser und mehr

15.2 Synchroner und asynchroner Schaltwerke

15.3 Schaltungen mit Delays

- 15.3.1 4-Bit-Ringzähler als synchrones Schaltwerk
- 15.3.2 Delays
- 15.3.3 Realisierung von Delays mit Flipflops

15.4 Zähler und Frequenzteiler

- 15.4.1 Synchroner 4-Bit-Ringzähler mit JK-Flipflops
- 15.4.2 Asynchroner 4-Bit-Ringzähler mit T-Flipflops
- 15.4.3 Synchroner BCD-Zähler (Mod-10) mit T-Flipflops
- 15.4.4 Asynchroner BCD-Zähler (Mod-10) mit JK-Flipflops

15.5 Schieberegister

15.6 Entwurf synchroner Schaltwerke mittels Automaten

- 15.6.1 Kurze Einführung in die Automatentheorie
- 15.6.2 Entwurf von Schaltwerken mit Moore- und Mealy-Automaten

Kapitel 16: Prozessorarchitekturen, Speicher und Caches

16.1 Rätsel: Schachbrett-Quadrate, Flickermuster, Kreuzformfirma

Inhaltsverzeichnis

16.2 CISC und RISC

16.3 Pipelining (Fließbandverarbeitung)

16.3.1 Unterschiedliche Phasen beim Pipelining

16.3.2 Geschwindigkeitsgewinn beim Pipelining

16.3.3 Hazards beim Pipelining

16.4 Speicher für Prozessoren

16.5 Caches

16.5.1 Das Lokalitätsprinzip und der Cache-Controller

16.5.2 Der Lesezugriff

16.5.3 Vollasoziative und direktabgebildete Caches

16.5.4 Der Schreibzugriff

16.6 Virtueller Speicher

16.6.1 Paging

16.6.2 Segmentierung

Teil IV: Theoretische Informatik

Kapitel 17: Automatentheorie und formale Sprachen

17.1 Rätsel: Weg durch ein Labyrinth und um die Ecke gedacht

17.2 Lexikalische und syntaktische Analyse

17.3 Reguläre Sprachen und endliche Automaten

17.3.1 Alphabet, Wort und Sprache

17.3.2 Reguläre Ausdrücke

17.3.3 Endliche Automaten und reguläre Sprachen

17.3.4 Realisierung endlicher Automaten

17.3.5 lex -- Ein Werkzeug für die lexikalische Analyse

17.4 Kontextfreie Sprachen und Kellerautomaten

17.4.1 Kontextfreie Grammatiken

17.4.2 Kellerautomaten

17.4.3 yacc -- Ein Werkzeug für die Syntaxanalyse

17.4.4 lex und yacc im Zusammenspiel

17.4.5 Rekursion bei der Syntaxanalyse

17.5 Die unterschiedlichen Phasen eines Compilers

Kapitel 18: Berechenbarkeitstheorie

18.1 Rätsel: Kneipen, Ei, stehen gebliebene Uhr und Alter

Inhaltsverzeichnis

18.2 Berechenbare Funktionen

18.3 Nicht berechenbare Funktionen

18.3.1 Das Diagonalverfahren von Cantor

18.3.2 Nicht durch einen Algorithmus berechenbare Funktionen

18.3.3 Die Church'sche Algorithmus-Definition

18.4 Berechenbarkeitskonzepte

18.4.1 Turingmaschinen

18.4.2 Turing-berechenbare Funktionen

18.4.3 Registermaschinen

18.4.4 GOTO- und WHILE-Programme

18.4.5 LOOP-Programme (FOR-Programme)

18.4.6 Primitive Rekursion

18.4.7 μ -Rekursion

18.4.8 Die Ackermann-Funktion

18.4.9 Die Church'sche These und die Chomsky-Hierarchie

18.5 Prinzipiell unlösbare Probleme

18.5.1 Entscheidbare Mengen

18.5.2 semi-entscheidbare Mengen (Game of Life und Halteproblem)

18.5.3 Unberechenbarkeit (Fleißiger Biber)

Kapitel 19: Komplexitätstheorie

19.1 Rätsel: Falsche Uhrzeit, Kalenderrechnen und mehr

19.2 Die Klasse P für praktisch lösbare Probleme

19.3 Nichtdeterminismus und die Klasse NP

19.3.1 Das SAT-Problem als erstes NP-Problem

19.3.2 Reduzierung auf ja/nein-Probleme mit zugehörigen Sprachen

19.3.3 Nichtdeterminismus

19.3.4 Die Klasse NP

19.4 Der Satz von Cook und NP-Vollständigkeit

19.4.1 Das Dreifarbenproblem als Spezialfall des SAT-Problems

19.4.2 NP-Vollständigkeit

19.4.3 $P = NP?$

19.4.4 Das 3SAT-Problem

19.4.5 Das Cliquesproblem

19.4.6 Das Rucksack- und Teilsummen-Problem

Inhaltsverzeichnis

19.4.7 Das Hamilton-Problem

19.4.8 Das Problem des Handlungsreisenden

19.4.9 Hierarchie der NP-vollständigen Probleme

19.5 Approximationsalgorithmen

Teil V: Codes, Kompression, Kryptografie

Kapitel 20: Fehlertolerante Codes

20.1 Rätsel: Auf der Demo mit Bruder und Schwester

20.2 Motivation für fehlertolerante Codes

20.3 "84k aus n"93-Codes

20.4 Der Hammingabstand eines Codes

20.5 Eindimensionale Parity-Prüfung

20.6 Zweidimensionale Parity-Prüfung

20.7 Hamming-Codes

20.8 CRC-Kodierung

Kapitel 21: Datenkompression

21.1 Rätsel: Tierseuche

21.2 Verlustbehaftete und verlustlose Kompression

21.3 Codes mit variabel langen Codewörtern

21.4 Fano-Bedingung für Dekodierbarkeit eines Codes

21.5 Lauflängenkodierung ("84run-length encoding"93)

21.6 Shannon-Fano-Kodierung

21.7 Huffman-Kodierung

21.8 Arithmetische Kodierung

21.9 Lempel-Ziv-Kodierungen

21.9.1 Der LZ77-Algorithmus

21.9.2 Der LZSS-Algorithmus

21.9.3 Der LZ78-Algorithmus

21.9.4 Der LZW-Algorithmus

21.9.5 Varianten der Lempel-Ziv-Kodierung

Kapitel 22: Kryptografie

22.1 Rätsel: Weinflasche und Erben von Weinfässern

Inhaltsverzeichnis

- 22.2 Allgemeines zu Kryptosystemen
- 22.3 Einfache Verschlüsselungsmethoden
 - 22.3.1 Cäsar-Chiffre
 - 22.3.2 Chiffre mit eigener Zuordnungstabelle
- 22.4 Vigenère-Verschlüsselungsmethoden
- 22.5 Verschlüsselung mittels Zufallsfolgen
- 22.6 Kryptosysteme mit öffentlichen Schlüsseln
 - 22.6.1 Eigenschaften von Public-Key-Systemen
 - 22.6.2 Der Satz von Euler
 - 22.6.3 Schlüsselerzeugung beim RSA-Algorithmus
 - 22.6.4 Ver- und Entschlüsselung mit dem RSA-Algorithmus

Weiterführende Literatur

Sachregister

Copyright

Copyright

Daten, Texte, Design und Grafiken dieses eBooks, sowie die eventuell angebotenen eBook-Zusatzdaten sind urheberrechtlich geschützt. Dieses eBook stellen wir lediglich als **persönliche Einzelplatz-Lizenz** zur Verfügung!

Jede andere Verwendung dieses eBooks oder zugehöriger Materialien und Informationen, einschließlich

- der Reproduktion,
- der Weitergabe,
- des Weitervertriebs,
- der Platzierung im Internet, in Intranets, in Extranets,
- der Veränderung,
- des Weiterverkaufs und
- der Veröffentlichung

bedarf der **schriftlichen Genehmigung** des Verlags. Insbesondere ist die Entfernung oder Änderung des vom Verlag vergebenen Passwort- und DRM-Schutzes ausdrücklich untersagt!

Bei Fragen zu diesem Thema wenden Sie sich bitte an: **info@pearson.de**

Zusatzdaten

Möglicherweise liegt dem gedruckten Buch eine CD-ROM mit Zusatzdaten oder ein Zugangscode zu einer eLearning Plattform bei. Die Zurverfügungstellung dieser Daten auf unseren Websites ist eine freiwillige Leistung des Verlags. **Der Rechtsweg ist ausgeschlossen.** Zugangscodes können Sie darüberhinaus auf unserer Website käuflich erwerben.

Hinweis

Dieses und viele weitere eBooks können Sie rund um die Uhr und legal auf unserer Website herunterladen:

<https://www.pearson-studium.de>