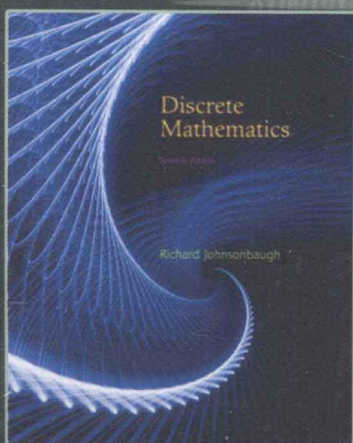


离散数学

(第七版)

Discrete Mathematics, Seventh Edition



英文版

[美] Richard Johnsonbaugh 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

国外计算机科学教材系列

离散数学

(第七版)

(英文版)

Discrete Mathematics, Seventh Edition

[美] Richard Johnsonbaugh 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书从算法分析和问题求解的角度,全面系统地介绍了离散数学的基础概念及相关知识。书中通过大量实例,深入浅出地讲解了数理逻辑、组合算法、图论、布尔代数、网络模型、形式语言与自动机理论等与计算机科学密切相关的前沿课题,既着重于各部分内容之间的紧密联系,又深入探讨了相关的概念、理论、算法和实际应用。本书内容叙述严谨、推演详尽,各章配有相当数量的习题,并且在书后提供了相应的提示和答案,为读者迅速掌握有关知识提供了有效的帮助。

本书既可作为计算机科学及计算数学等专业的本科生和研究生教材,也可作为工程技术人员和相关人员的参考书。

Original edition, entitled Discrete Mathematics, Seventh Edition, 9780131593183 by Richard Johnsonbaugh, published by Pearson Education, Inc., publishing as Prentice Hall, Copyright © 2009 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, Copyright © 2009.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in People's Republic of China excluding Hong Kong, Macau and Taiwan.

本书由 Pearson Education 培生教育出版亚洲有限公司授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

此版本仅限在中国大陆(不包括香港、澳门特别行政区以及台湾地区)发行与销售。

本书贴有 Pearson Education 培生教育出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2009-1393

图书在版编目(CIP)数据

离散数学=Discrete Mathematics:第7版:英文/(美)约翰逊鲍夫(Johnsonbaugh, R.)著.

北京:电子工业出版社,2009.4

(国外计算机科学教材系列)

ISBN 978-7-121-08534-5

I. 离… II. 约… III. 离散数学-教材-英文 IV. 0158

中国版本图书馆CIP数据核字(2009)第040620号

责任编辑:冯小贝

印 刷:北京市天竺颖华印刷厂

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

开 本:787×980 1/16 印张:49.5 字数:1267千字

印 次:2009年4月第1次印刷

定 价:79.00元

凡所购买电子工业出版社的图书有缺损问题,请向购买书店调换;若书店售缺,请与本社发行部联系。联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

出版说明

21世纪初的5至10年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入WTO后的今天,培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择 and 自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如Pearson Education培生教育出版集团、麦格劳—希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过与作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

主 任	杨芙清	北京大学教授 中国科学院院士 北京大学信息与工程学部主任 北京大学软件工程研究所所长
委 员	王 珊	中国人民大学信息学院教授 中国计算机学会副理事长，数据库专业委员会主任
	胡道元	清华大学计算机科学与技术系教授 国际信息处理联合会通信系统中国代表
	钟玉琢	清华大学计算机科学与技术系教授、博士生导师 清华大学深圳研究生院信息学部主任
	谢希仁	中国人民解放军理工大学教授 全军网络技术研究中心主任、博士生导师
	尤晋元	上海交通大学计算机科学与工程系教授 上海分布计算技术中心主任
	施伯乐	上海国际数据库研究中心主任、复旦大学教授 中国计算机学会常务理事、上海市计算机学会理事长
	邹 鹏	国防科学技术大学计算机学院教授、博士生导师 教育部计算机基础课程教学指导委员会副主任委员
	张昆藏	青岛大学信息工程学院教授

PREFACE

This updated edition is intended for a one- or two-term introductory course in discrete mathematics, based on my experience in teaching this course over many years and requests from users of previous editions. Formal mathematics prerequisites are minimal; calculus is *not* required. There are no computer science prerequisites. The book includes examples, exercises, figures, tables, sections on problem-solving, sections containing problem-solving tips, section reviews, notes, chapter reviews, self-tests, and computer exercises to help the reader master introductory discrete mathematics. In addition, an *Instructor's Guide* and website are available.

In the early 1980s there were few textbooks appropriate for an introductory course in discrete mathematics. However, there was a need for a course that extended students' mathematical maturity and ability to deal with abstraction, which also included useful topics such as combinatorics, algorithms, and graphs. The original edition of this book (1984) addressed this need and significantly influenced the development of discrete mathematics courses. Subsequently, discrete mathematics courses were endorsed by many groups for several different audiences, including mathematics and computer science majors. A panel of the Mathematical Association of America (MAA) endorsed a year-long course in discrete mathematics. The Educational Activities Board of the Institute of Electrical and Electronics Engineers (IEEE) recommended a freshman discrete mathematics course. The Association for Computing Machinery (ACM) and IEEE accreditation guidelines mandated a discrete mathematics course. This edition, like its predecessors, includes topics such as algorithms, combinatorics, sets, functions, and mathematical induction endorsed by these groups. It also addresses understanding and constructing proofs and, generally, expanding mathematical maturity.

Logic and Proofs Changes

The changes in this book, the seventh edition, result from comments and requests from numerous users and reviewers of previous editions of the book. The biggest change from the sixth to the seventh edition occurs in Chapters 1–3. The first chapter in the sixth edition, Logic and Proofs, has been divided into two chapters in the seventh edition: Sets and Logic (Chapter 1) and Proofs (Chapter 2). Except for the section on sets, Chapters 2 (The Language of Mathematics) and 3 (Relations) in the sixth edition have been combined into Chapter 3 (Functions, Sequences, and Relations) in the seventh edition. Pre-publication reviews have been very enthusiastic about these changes.

The section on sets is now the first section of the book. This change permits the use of set terminology throughout the book. It makes sets available for proofs in examples and exercises, thus providing more interesting examples earlier than in the previous editions. We can even use sets to introduce proofs (e.g., proving that two sets are equal, proving that one set is a subset of another) before fully discussing proofs and proof techniques.

The discussion about how to construct proofs has been greatly expanded. Sections 2.1 and 2.2 are new extended sections on mathematical systems and proof techniques. In addition, there are expanded subsections on proofs of equivalence and existence proofs (including constructive and nonconstructive existence proofs). Nearly every proof is preceded by a Discussion section and/or accompanied by a figure. The Problem-Solving Tips sections include expanded advice, examples, and so on, on how to do proofs, how to write up proofs, and common errors in proofs. There are two new Problem-Solving Corners, one on quantifiers, the other on proofs (see Proving Some Properties of Real Numbers).

The discussion of arguments and rules of inference for propositions has been moved to follow the discussion of propositions. The rules of inference for quantified statements are integrated within the quantifiers section.

The number of examples and exercises has been vastly expanded. In the sixth edition, there were approximately 1370 worked examples and exercises in the first three chapters. In the seventh edition, there are approximately 1640 worked examples and exercises in the first three chapters. Of course, not just the quantity of examples and exercises is important, it is also the quality. In most of the examples found in the sixth edition, the discussion has been expanded and additional motivation has been added.

Other Changes from the Sixth Edition

Other changes from the sixth edition are as follows:

- The description and notation for integers (\mathbf{Z} , \mathbf{Z}^+ , \mathbf{Z}^- , \mathbf{Z}^{nonneg}), rational numbers (with \mathbf{Z} replaced by \mathbf{Q}), and real numbers (with \mathbf{Z} replaced by \mathbf{R}) are introduced early (in Section 1.1, Sets).
- Proofs, rather than sketches of proofs as in the sixth edition, are provided for Theorems 5.1.17 and 5.1.22, which give the greatest common divisor and least common multiple of two integers given their prime factorizations.
- Recursive algorithms are given (Algorithms 5.3.9 and 5.3.10) to compute the greatest common divisor of two integers a and b , $\gcd(a, b)$, and to compute integers s and t satisfying $\gcd(a, b) = sa + tb$.
- The Inclusion-Exclusion Principle has been added to Section 6.1.
- Internet addressing is now included in Section 6.1.
- Several practice exercises have been added to Section 6.1 that specifically say to use either the Multiplication Principle or the Addition Principle. These exercises precede other exercises that require the reader to figure out which Principle to use or require using both Principles.
- The section on Generalized Permutations and Combinations (Section 6.6 in the sixth edition) now follows Sections 6.1 and 6.2 (Basic Principles, Permutations and Combinations) since generalized permutations and combinations are so closely related to the material in Sections 6.1 and 6.2.
- Several relatively straightforward “warm-up” exercises have been added before the main pigeonhole exercises (Section 6.8).
- More exercises on graph isomorphism have been added (Section 8.6). The exercises have been divided into those that ask for a proof that two given graphs are isomorphic, those that ask for a proof that two given graphs are not isomorphic, and those that ask the reader to determine, with a proof, whether two given graphs are isomorphic.

- In Section 9.3, there are several new backtracking exercises including the popular Sudoku puzzle.
- More examples and exercises are included to highlight common errors (for example, the subsection Some Common Errors that precedes the Section 2.1 Review Exercises discusses some common errors in proofs, and Example 6.2.24 illustrates a common counting error).
- A number of recent books and articles have been added to the list of references. Several book references have been updated to current editions.
- The number of worked examples has been increased to over 650. (There were approximately 600 in the sixth edition.)
- The number of exercises has been increased to nearly 4200. (There were approximately 4000 in the sixth edition.)

Contents and Structure

This book includes

- Sets and logic (including quantifiers). Practical examples such as using the Google search engine are included (Example 1.2.13). Translating between English and symbolic expressions is discussed as is logic in programming languages. A *logic game*, which offers an alternative way to determine whether a quantified propositional function is true or false, is discussed in Example 1.6.15.
- Proofs (Chapter 2). Proof techniques discussed include direct proofs, counterexamples, proof by contradiction, proof by contrapositive, proofs by cases, proofs of equivalence, existence proofs (constructive and nonconstructive), and mathematical induction. Loop invariants are presented as a practical application of mathematical induction. We also include a brief, optional section on resolution proofs (a proof technique that can be automated).
- Functions, sequences, sum and product notations, strings, and relations (Chapter 3), including motivating examples such as the new 13-character international standard book number (ISBN), an introduction to hash functions, and pseudorandom number generators (Section 3.1); an application of partial orders to task scheduling (Section 3.3); and relational databases (Section 3.6).
- A thorough discussion of algorithms, recursive algorithms, and the analysis of algorithms (Chapter 4). A number of examples of algorithms are presented before getting into big-oh and related notations (Sections 4.1 and 4.2), thus providing a gentle introduction and motivating the formalism that follows. An algorithmic approach is taken throughout this book. We mention that many modern algorithms do not have all the properties of classical algorithms (e.g., many modern algorithms are not general, deterministic, or even finite). To illustrate the point, an example is given of a randomized algorithm (Example 4.2.4). The algorithms are written in a flexible form of pseudocode, which resembles currently popular languages such as C, C++, and Java. (The book does not assume any computer science prerequisites; the description of the pseudocode used is given in Appendix C.) Among the algorithms presented are tiling (Section 4.4), the Euclidean algorithm for finding the greatest common divisor (Section 5.3), the RSA public-key encryption algorithm (Section 5.4), generating combinations and permutations (Section 6.4), merge sort (Section 7.3), Dijkstra's shortest-path algorithm (Section 8.4), backtracking algorithms (Section 9.3), breadth-first and depth-first search (Section 9.3), tree traversals (Section 9.6), evaluating a game tree (Section 9.9), finding a

maximal flow in a network (Section 10.2), finding a closest pair of points (Section 13.1), and computing the convex hull (Section 13.2).

- A full discussion of the “big oh,” omega, and theta notations for the growth of functions (Section 4.3). Having all of these notations available makes it possible to make precise statements about the growth of functions and the time and space required by algorithms.
- An introduction to number theory (Chapter 5). This chapter includes classical results (e.g., divisibility, the infinitude of primes, fundamental theorem of arithmetic), as well as algorithmic number theory (e.g., the Euclidean algorithm to find the greatest common divisor, exponentiation using repeated squaring, computing s and t such that $\gcd(a, b) = sa + tb$, computing an inverse modulo an integer). The major application is the RSA public-key cryptosystem (Section 5.4). The calculations required by the RSA public-key cryptosystem can be performed using the algorithms previously developed in the chapter.
- Combinations, permutations, discrete probability, and the Pigeonhole Principle (Chapter 6). Two optional sections (Sections 6.5 and 6.6) treat discrete probability.
- Recurrence relations and their use in the analysis of algorithms (Chapter 7).
- Graphs, including coverage of graph models of parallel computers, the knight’s tour, Hamiltonian cycles, graph isomorphisms, and planar graphs (Chapter 8). Theorem 8.4.3 gives a simple, short, elegant proof of the correctness of Dijkstra’s algorithm.
- Trees, including binary trees, tree traversals, minimal spanning trees, decision trees, the minimum time for sorting, and tree isomorphisms (Chapter 9).
- Networks, the maximal flow algorithm, and matching (Chapter 10).
- A treatment of Boolean algebras that emphasizes the relation of Boolean algebras to combinatorial circuits (Chapter 11).
- An approach to automata emphasizing modeling and applications (Chapter 12). The *SR* flip-flop circuit is discussed in Example 12.1.11. Fractals, including the von Koch snowflake, are described by special kinds of grammars (Example 12.3.19).
- An introduction to computational geometry (Chapter 13).
- Appendixes on matrices, basic algebra, and pseudocode.
- A strong emphasis on the interplay among the various topics. As examples, mathematical induction is closely tied to recursive algorithms (Section 4.4); the Fibonacci sequence is used in the analysis of the Euclidean algorithm (Section 5.3); many exercises throughout the book require mathematical induction; we show how to characterize the components of a graph by defining an equivalence relation on the set of vertices (see the discussion following Example 8.2.13); and we count the number of nonisomorphic n -vertex binary trees (Theorem 9.8.12).
- A strong emphasis on reading and doing proofs. Most proofs of theorems are illustrated with annotated figures and/or motivated by special Discussion sections. Separate sections (Problem-Solving Corners) show students how to attack and solve problems and how to do proofs. Special end-of-section Problem-Solving Tips highlight the main problem-solving techniques of the section.
- A large number of applications, especially applications to computer science.
- Figures and tables to illustrate concepts, to show how algorithms work, to elucidate proofs, and to motivate the material. Several figures illustrate proofs of theorems. The captions of these figures provide additional explanation and insight into the proofs.

- Section review exercises.
- Notes sections with suggestions for further reading.
- Chapter reviews.
- Chapter self-tests.
- Computer exercises.
- A reference section containing more than 160 references.
- Front and back endpapers that summarize the mathematical and algorithm notation used in the book.

Each chapter is organized as follows:

Overview
 Section
 Section Review Exercises
 Section Exercises
 Section
 Section Review Exercises
 Section Exercises
 ⋮
 Notes
 Chapter Review
 Chapter Self-Test
 Computer Exercises

Section review exercises review the key concepts, definitions, theorems, techniques, and so on of the section. All section review exercises have answers in the back of the book. Although intended for reviews of the sections, section review exercises can also be used for placement and pretesting.

Notes contain suggestions for further reading. Chapter reviews provide reference lists of the key concepts of the chapters. Chapter self-tests contain four exercises per section, with answers in the back of the book.

Computer exercises include projects, implementation of some of the algorithms, and other programming related activities. Although there is no programming prerequisite for this book and no programming is introduced in the book, these exercises are provided for those readers who want to explore discrete mathematics concepts with a computer.

In addition, most chapters have Problem-Solving Corners.

Exercises

The book contains nearly 4200 exercises, 145 of which are computer exercises. Exercises felt to be more challenging than average are indicated with a star, *. Exercise numbers in color (approximately one-third of the exercises) indicate that the exercise has a hint or solution in the back of the book. The solutions to most of the remaining exercises may be found in the *Instructor's Guide*. A handful of exercises are clearly identified as requiring calculus. No calculus concepts are used in the main body of the book and, except for these marked exercises, no calculus is needed to solve the exercises.

Examples

The book contains over 650 worked examples. These examples show students how to tackle problems in discrete mathematics, demonstrate applications of the theory, clarify proofs, and help motivate the material.

Problem-Solving Corners

The Problem-Solving Corner sections help students attack and solve problems and show them how to do proofs. Written in an informal style, each is a self-contained section following the discussion of the subject of the problem. Rather than simply presenting a proof or a solution to a problem, in these sections the intent is to show alternative ways of attacking a problem, to discuss what to look for in trying to obtain a solution to a problem, and to present problem-solving and proof techniques.

Each Problem-Solving Corner begins with a statement of a problem. After stating the problem, ways to attack the problem are discussed. This discussion is followed by techniques for finding a solution. After a solution is found, a formal solution is given to show how to correctly write up a formal solution. Finally, the problem-solving techniques used in the section are summarized. In addition, some of these sections include a Comments subsection, which discusses connections with other topics in mathematics and computer science, provides motivation for the problem, and lists references for further reading about the problem. Exercises conclude some Problem-Solving Corners.

Instructor Supplement

An *Instructor's Guide* is available at no cost from the publisher to instructors who adopt or sample this book. It should be requested from your local Prentice Hall representative. The *Instructor's Guide* contains solutions to most of the exercises not included in the book.

Website

The website

<http://condor.depaul.edu/~rjohnson/dm7th>

for the seventh edition contains



- Expanded explanations of difficult material and links to other sites for additional information about discrete mathematics topics. The icon shown signals that an expanded explanation or a link is at the book's website.
- Supplementary material
- Computer programs
- An errata list.

Acknowledgments

I received helpful comments from many persons, including Gary Andrus, Kendall Atkinson, Greg Bachelis, André Berthiaume, Gregory Brewster, Robert Busby, David G. Cantor, Tim Carroll, Joseph P. Chan, Hon-Wing Cheng, I-Ping Chu, Robert Crawford, Henry D'Angelo, Jerry Delazzer, Br. Michael Driscoll, Carl E. Eckberg, Herbert Enderton, Susanna Epp, Bob Fisher, Brendan Frey, Dennis Garity, Gerald Gordon, Jerrold Grossman, Reino Hakala, Mark Herbster, Craig Jensen, Steve Jost, Martin Kalin, Aaron Keen, Nicholas Krier, Warren Krueger, Glenn Lancaster, Miguel Lerma, Donald E. G. Malm, Nick Meshes, Truc Nguyen, Suely Oliveira, Kevin Phelps, Jenni Plane, Randall Pruim, Mansur Samadzadeh, Sigrid (Anne) Settle, David Stewart, James H. Stoddard, Chaim Goodman Strauss, Bogdan Suceava, Michael Sullivan, Edward J. Williams, Anthony S. Wojcik, and Hanyi Zhang. Thanks also to all of the users of my book for their helpful letters and e-mail.

Special thanks for this edition go to Greg Brewster, my DePaul colleague, for consulting about Internet addressing.

For reviewing the manuscript for this edition, thanks go to Scott Annin, California State University, Fullerton; Natacha Fontes-Merz, Westminster College; Ronald I. Greenberg, Loyola University; John Greiner, Rice University; Eitan Grinspun, Columbia University; Wu Jing, Fayetteville State University; Harold Reiter, UNC Charlotte; and Christopher N. Swanson, Ashland University.

I am grateful to my friendly copy editor, Patricia Johnsonbaugh, for her careful reading of the entire manuscript, improving the exposition, catching miscues I wrote but should not have, and help with the index.

I have received consistent support from the staff at Prentice Hall. Special thanks for their help go to Dee Bernhard, Associate Editor; Scott Disanno, Senior Managing Editor; Marcia Horton, Editorial Director; Jennifer Lonschein, Editorial Assistant; Holly Stark, Senior Editor; and Irwin Zucker, Production Editor.

Richard Johnsonbaugh

CONTENTS

Preface XI

1 → Sets and Logic 1

- 1.1 Sets 2
- 1.2 Propositions 14
- 1.3 Conditional Propositions and Logical Equivalence 21
- 1.4 Arguments and Rules of Inference 31
- 1.5 Quantifiers 36
- 1.6 Nested Quantifiers 51
- Problem-Solving Corner:** Quantifiers 60
- Notes 62
- Chapter Review 62
- Chapter Self-Test 63
- Computer Exercises 64

2 → Proofs 66

- 2.1 Mathematical Systems, Direct Proofs, and Counterexamples 67
- 2.2 More Methods of Proof 76
- Problem-Solving Corner:** Proving Some Properties of Real Numbers 87
- 2.3 Resolution Proofs[†] 90
- 2.4 Mathematical Induction 93
- Problem-Solving Corner:** Mathematical Induction 106
- 2.5 Strong Form of Induction and the Well-Ordering Property 108
- Notes 115
- Chapter Review 115
- Chapter Self-Test 116
- Computer Exercises 116

[†]This section can be omitted without loss of continuity.

3 → Functions, Sequences, and Relations 117

- 3.1 Functions 117
Problem-Solving Corner: Functions 135
- 3.2 Sequences and Strings 136
- 3.3 Relations 148
- 3.4 Equivalence Relations 159
Problem-Solving Corner: Equivalence Relations 166
- 3.5 Matrices of Relations 168
- 3.6 Relational Databases[†] 173
Notes 178
Chapter Review 178
Chapter Self-Test 179
Computer Exercises 180

4 → Algorithms 181

- 4.1 Introduction 181
- 4.2 Examples of Algorithms 186
- 4.3 Analysis of Algorithms 193
Problem-Solving Corner: Design and Analysis of an Algorithm 211
- 4.4 Recursive Algorithms 213
Notes 220
Chapter Review 221
Chapter Self-Test 221
Computer Exercises 222

5 → Introduction to Number Theory 223

- 5.1 Divisors 223
- 5.2 Representations of Integers and Integer Algorithms 234
- 5.3 The Euclidean Algorithm 248
Problem-Solving Corner: Making Postage 259
- 5.4 The RSA Public-Key Cryptosystem 260
Notes 263
Chapter Review 263
Chapter Self-Test 263
Computer Exercises 264

6 → Counting Methods and the Pigeonhole Principle 265

- 6.1 Basic Principles 265
Problem-Solving Corner: Counting 277

[†]This section can be omitted without loss of continuity.

6.2	Permutations and Combinations	278
	Problem-Solving Corner: Combinations	291
6.3	Generalized Permutations and Combinations	293
6.4	Algorithms for Generating Permutations and Combinations	299
6.5	Introduction to Discrete Probability [†]	305
6.6	Discrete Probability Theory [†]	309
6.7	Binomial Coefficients and Combinatorial Identities	320
6.8	The Pigeonhole Principle	325
	Notes	330
	Chapter Review	330
	Chapter Self-Test	330
	Computer Exercises	332

7 → Recurrence Relations 333

7.1	Introduction	333
7.2	Solving Recurrence Relations	345
	Problem-Solving Corner: Recurrence Relations	358
7.3	Applications to the Analysis of Algorithms	361
	Notes	373
	Chapter Review	373
	Chapter Self-Test	374
	Computer Exercises	374

8 → Graph Theory 376

8.1	Introduction	377
8.2	Paths and Cycles	388
	Problem-Solving Corner: Graphs	399
8.3	Hamiltonian Cycles and the Traveling Salesperson Problem	400
8.4	A Shortest-Path Algorithm	407
8.5	Representations of Graphs	412
8.6	Isomorphisms of Graphs	417
8.7	Planar Graphs	425
8.8	Instant Insanity [†]	431
	Notes	435
	Chapter Review	436
	Chapter Self-Test	437
	Computer Exercises	438

9 → Trees 440

9.1	Introduction	440
9.2	Terminology and Characterizations of Trees	448
	Problem-Solving Corner: Trees	453

[†]This section can be omitted without loss of continuity.

9.3	Spanning Trees	454
9.4	Minimal Spanning Trees	461
9.5	Binary Trees	467
9.6	Tree Traversals	474
9.7	Decision Trees and the Minimum Time for Sorting	480
9.8	Isomorphisms of Trees	486
9.9	Game Trees [†]	496
	Notes	505
	Chapter Review	505
	Chapter Self-Test	506
	Computer Exercises	508

10 → Network Models 510

10.1	Introduction	510
10.2	A Maximal Flow Algorithm	516
10.3	The Max Flow, Min Cut Theorem	524
10.4	Matching	528
	Problem-Solving Corner: Matching	533
	Notes	534
	Chapter Review	535
	Chapter Self-Test	536
	Computer Exercises	536

11 → Boolean Algebras and Combinatorial Circuits 537

11.1	Combinatorial Circuits	537
11.2	Properties of Combinatorial Circuits	544
11.3	Boolean Algebras	549
	Problem-Solving Corner: Boolean Algebras	554
11.4	Boolean Functions and Synthesis of Circuits	556
11.5	Applications	561
	Notes	570
	Chapter Review	570
	Chapter Self-Test	571
	Computer Exercises	572

12 → Automata, Grammars, and Languages 573

12.1	Sequential Circuits and Finite-State Machines	573
12.2	Finite-State Automata	579
12.3	Languages and Grammars	585

[†]This section can be omitted without loss of continuity.

12.4	Nondeterministic Finite-State Automata	595
12.5	Relationships Between Languages and Automata	602
	Notes	608
	Chapter Review	609
	Chapter Self-Test	609
	Computer Exercises	611

13 → **Computational Geometry** 612

13.1	The Closest-Pair Problem	612
13.2	An Algorithm to Compute the Convex Hull	617
	Notes	625
	Chapter Review	625
	Chapter Self-Test	625
	Computer Exercises	626

Appendix 627

A → **Matrices** 627

B → **Algebra Review** 631

C → **Pseudocode** 644

References 650

Hints and Solutions to Selected Exercises 655

Index 754