DISCRETE MATHEMATICS

AND ITS APPLICATIONS

KENNETH H. ROSEN

Discrete Mathematics and Its Applications

Kenneth H. Rosen
AT&T Information Systems Laboratory

The Random House/Birkhäuser Mathematics Series

Random House
New York

Cover: Jasper Johns, *Between the Clock and the Bed,* 1981. Encaustic on canvas, $6' \frac{1}{8}'' \times 10' 6\frac{3}{8}''$. Collection, The Museum of Modern Art, New York. Given anonymously. Photograph © The Museum of Modern Art, New York.

First Edition

98765432

All inquiries should be addressed to Random House, Inc., 201 East 50th Street, New York, N.Y. 10022. Published in the United States by Random House, Inc., and simultaneously in Canada by Random House of Canada, Limited, Toronto.

Library of Congress Cataloging-in-Publication Data

Rosen, Kenneth H.
Discrete mathematics and its applications.

(The Random House/Birkhäuser mathematics series)
Bibliography: p.
Includes index.
1. Mathematics—1961–   . 2. Electronic data
processing—Mathematics. I. Title. II. Series.
QA39.2.R654 1987 511 87-20694
ISBN 0-394-36768-5

Preface

In writing this book I have been guided by two purposes that have resulted from my longstanding experience and interest in teaching discrete mathematics. For the student, my purpose was to write in a precise, readable manner with the concepts and techniques of discrete mathematics clearly presented and demonstrated. For the instructor, my purpose was carefully to design a flexible, comprehensive teaching tool that uses proven pedagogical techniques in mathematics.

This text is designed for a one- or two-term introductory discrete mathematics course to be taken by students in a wide variety of majors, including mathematics, computer science, and engineering. College algebra is the only prerequisite.

Goals of a Discrete Mathematics Course

A discrete mathematics course has more than one purpose. Students should learn a particular set of mathematical facts and how to apply them; but more importantly, such a course should teach students how to think mathematically. To achieve these goals, this text stresses mathematical reasoning and the different ways problems are solved. Five important themes are interwoven in this text: mathematical reasoning, combinatorial analysis, discrete structures, applications and modeling, and algorithmic thinking. A successful discrete mathematics course should blend and carefully balance all five of these themes.

1. *Mathematical Reasoning:* Students must understand mathematical reasoning in order to read, comprehend, and construct mathematical argu-

ments. This text starts with a discussion of mathematical logic, which serves as the foundation for the subsequent discussions of methods of proof. The technique of mathematical induction is stressed through many different types of examples of such proofs and a careful explanation of why mathematical induction is a valid proof technique.

2. *Combinatorial Analysis:* An important problem-solving skill is the ability to count or enumerate objects. The discussion of enumeration in this book begins with the basic techniques of counting. The stress is on performing combinatorial analysis to solve counting problems, not on applying formulae.
3. *Discrete Structures:* A course in discrete mathematics should teach students how to work with discrete structures, which are the abstract mathematical structures used to represent discrete objects and relationships between these objects. These discrete structures include sets, permutations, relations, graphs, trees, and finite state machines.
4. *Applications and Modeling:* Discrete mathematics has applications to almost every conceivable area of study. There are many applications to computer science in this text, as well as applications to such diverse areas as chemistry, botany, zoology, linguistics, geography, and business. Modeling with discrete mathematics is an extremely important problem-solving skill, which students have the opportunity to develop by constructing their own models in some of the exercises in the book.
5. *Algorithmic Thinking:* Certain classes of problems are solved by the specification of an algorithm. After an algorithm has been described, a computer program can be constructed implementing it. The mathematical portions of this activity, which include the specification of the algorithm, the verification that it works properly, and the analysis of the computer memory and time required to perform it, are all covered in this text. Algorithms are described using both English and an easily understood form of pseudocode.

Features

ACCESSIBILITY: There are no mathematical prerequisites beyond college algebra for this text. The few places in the book where calculus is referred to are explicitly noted. Most students should easily understand the pseudocode used in the text to express algorithms, regardless of whether they have formally studied programming languages. There is no formal computer science prerequisite.

Each chapter begins at an easily understood and accessible level. Once basic mathematical concepts have been carefully developed, more difficult material and applications to other areas of study are presented.

FLEXIBILITY: This text has been carefully designed for flexible use. The dependence of chapters on previous material has been minimized. Each chapter is divided into sections of approximately the same length, and each section is divided into subsections that form natural blocks of material for teaching. Instructors can easily pace their lectures using these blocks.

WRITING STYLE: The writing style in this book is direct and pragmatic. Precise mathematical language is used without excessive formalism and abstraction. Notation is introduced and used when appropriate. Care has been taken to balance the mix of notation and words in mathematical statements.

MATHEMATICAL RIGOR AND PRECISION: All definitions and theorems in this text are stated extremely carefully so that students will appreciate the precision of language and rigor needed in mathematics. Proofs are motivated and developed slowly; their steps are all carefully justified. Recursive definitions are explained and used extensively.

FIGURES AND TABLES: This text contains more than 450 figures. The figures are designed to illustrate key concepts and steps of proofs. Color has been carefully used in figures to illustrate important points. Whenever possible, tables have been used to summarize key points and illuminate quantitative relationships.

WORKED EXAMPLES: Over 500 examples are used to illustrate concepts, relate different topics, and introduce applications. In the examples, a question is first posed, then its solution is presented with the appropriate amount of detail.

APPLICATIONS: The applications included in this text demonstrate the utility of discrete mathematics in the solution of real-world problems. Applications to a wide variety of areas including computer science, psychology, chemistry, engineering, linguistics, biology, business, and many other areas are included in this text.

ALGORITHMS: Results in discrete mathematics are often expressed in terms of algorithms; hence, key algorithms are introduced in each chapter of the book. These algorithms are expressed in words and in an easily understood form of structured pseudocode, which is described and specified in Appendix 2. The computational complexity of the algorithms in the text are also analyzed at an elementary level.

KEY TERMS AND RESULTS: A list of key terms and results follows each chapter. The key terms include only the most important that students should learn, not every term defined in the chapter.

EXERCISES: There are over 1850 exercises in the text. There are many different types of questions posed. There is an ample supply of straightforward exercises that develop basic skills, as well as a good supply of challenging exercises. Exercises are stated clearly and unambiguously, and all are carefully graded for level of difficulty. Exercise sets contain special discussions, with exercises, that develop new concepts not covered in the text, permitting students to discover new ideas through their own work. Exercises that are somewhat more difficult than average are marked with a single star; those that are much more challenging are marked with two stars. Exercises whose solutions require calculus are explicitly noted. Exercises that develop results used in the text are clearly identified with the symbol ➲. Answers to all odd-numbered exercises are provided at the back of the text. The answers include proofs in which most of the steps are clearly spelled out.

SUPPLEMENTARY EXERCISE SETS: Each chapter is followed by a rich and varied set of supplementary exercises. These exercises are generally more difficult than those in the exercise sets following the sections. The supplementary exercises reinforce the concepts of the chapter and integrate different topics more effectively.

COMPUTER PROJECTS: Each chapter is followed by a set of computer projects. The 135 computer projects tie together what students may have learned in computing and in discrete mathematics. Computer projects that are more difficult than average, from both a mathematical and a programming point of view, are marked with a star, and those that are extremely challenging are marked with two stars.

CLASS TESTING: The manuscript has been used a number of times in my own introductory course, as well as in that of Jerrold Grossman at Oakland University. The student evaluations and comments gathered during class testing have had a significant and positive influence on the development of the text.

APPENDIXES: There are three appendixes to the text. The first covers exponential and logarithmic functions, reviewing some basic material used heavily in the course; the second specifies the pseudocode used to describe algorithms in this text; and the third discusses generating functions.

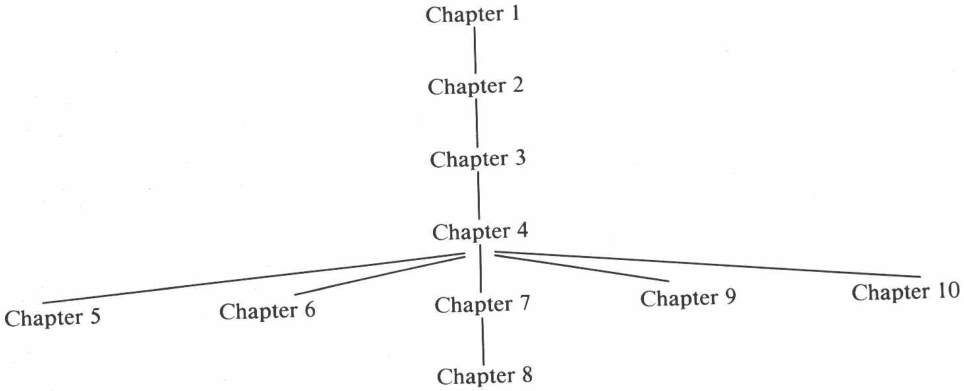
SUGGESTED READING: A list of suggested readings for each chapter is provided in a section at the end of the text. These suggested readings include books at or below the level of this text, more difficult books, expository articles, and articles in which discoveries in discrete mathematics were originally published.

How to Use This Book

This text has been carefully written and constructed to support discrete mathematics courses at several levels. The table below identifies the core and optional sections. An introductory one-term course in discrete mathematics at the sophomore level can be based on the core sections of the text with other sections covered at the discretion of the instructor. A two-term introductory course could include all the optional mathematics sections in addition to the core sections. A course with a strong computer science emphasis can be taught by covering some or all of the optional computer science sections.

Chapter	*Core Sections*	*Optional Computer Science Sections*	*Optional Mathematics Sections*
1	1.1–1.9 (as needed)		
2	2.1–2.3, 2.5 (as needed)	2.4	
3	3.1–3.3	3.4, 3.5	
4	4.1–4.4	4.6	4.5
5	5.1, 5.4	5.3	5.2, 5.5
6	6.1, 6.3, 6.5	6.2	6.4, 6.6
7	7.1–7.5		7.6–7.8
8	8.1	8.2, 8.3, 8.4	8.5, 8.6
9		9.1–9.4	
10		10.1–10.4	

Instructors using this book can adjust the level of difficulty of their course by omitting the more challenging examples at the end of sections as well as the more challenging exercises. The dependence of chapters on earlier chapters is shown in the following chart.



Ancillaries

STUDENT SOLUTIONS GUIDE: The student solutions guide, available separately, contains full solutions to all the odd-numbered problems in the exercise sets. These solutions explain why a particular method is used and why it works. For some exercises, one or two other possible approaches are described to show that a problem may be solved in several different ways.

INSTRUCTOR'S MANUAL: The instructor's manual contains full solutions to even-numbered exercises in the text. This manual also provides suggestions on how to teach the material in each chapter of the book, including the points to stress in each section and how to put the material into perspective. Furthermore, this manual contains sample examination questions for each chapter. The solutions to these sample questions are provided as well.

COMPUTER PROJECT SOLUTION GUIDE: A manual containing solutions to the computer projects is available to instructors who adopt the text. This manual gives the code in Pascal for these projects, including sample input and output. The programs are available on a disc that will run on an MS-DOS PC such as the AT&T PC6300.

Acknowledgments

I would like to thank the students in the discrete mathematics classes I have taught at Monmouth College during the past three years. Their feedback, suggestions, and corrections on the manuscript have made this a better text. I thank Jerrold Grossman of Oakland University for also classroom testing the manuscript, and for using his eagle eye in careful technical reviewing. The precision and clarity of the definitions, theorems, and exercises in the book, as well as the accuracy of the examples and answers, has been assured by Jerry's careful proofreading. I would like to thank Doug Riecken for the mammoth amount of work he did in preparing solutions for the computer projects.

I thank the many, many reviewers who have read all or part of the manuscript. Their encouragement, criticisms, and suggestions have been invaluable.

Russell Campbell, University of Northern Iowa
Tim Carroll, Bloomsburg University
Ron Davis, Millersville University
Thomas Dowling, Ohio State University
Ladnor Geissinger, University of North Carolina
Paul Gormley, Villanova University
Jerrold Grossman, Oakland University

Ken Johnson, North Dakota State University
David Jonah, Wayne State University
Gary Klatt, University of Wisconsin
Nicholas Krier, Colorado State University
Ernie Manes, University of Massachusetts
Francis Masat, Glassboro State College
Robert McGuigan, Westfield State College
John Michaels, SUNY Brockport
Charles Parry, Virginia Polytechnic Institute and State University
Harold Reiter, University of Maryland
Douglas Shier, Clemson University
Wallace Terwilligen, Bowling Green State University
Thomas Upson, Rochester Institute of Technology
Roman Voronka, New Jersey Institute of Technology
James Walker, University of South Carolina

I would like to thank the executives of Random House, New York, for their dedication to entering mathematics textbook publishing and for their strong support for this project, particularly Barry Fetterolf, Editor in Chief, and Seibert Adams, General Manager of the College Division. My editor, Wayne Yuhasz, deserves a great deal of credit for the overall quality of this text. He has been a constant source of enthusiasm and encouragement. He fully understands the role of discrete mathematics in the evolving mathematics curriculum and appreciates the delicate interplay between mathematics, computer science, and pedagogical issues. I appreciate the efforts of all the editorial staff of Random House who have worked with Wayne and me on this project: Ann Ritter, Anne Wightman, and Louise Bush. The production staff of Random House in Cambridge—Margaret Pinette, Project Manager; her assistant, Pamela Niebauer; Michael Weinstein, Production Manager; and his assistant, Susan Brown—have done an admirable job.

I am grateful for the support given to me by my management at AT&T Information Systems Laboratories during the preparation of this book. I thank my supervisors Garland Grammer and Len Stier, my department head Mac Mugglin, my director Mike Taylor, and my executive director John Sheehan for their support. I would like to thank Eileen Conyers, Mary DeVarti, Ann Postell, and Diane Stevens of AT&T for their help with the preparation of materials for the book.

To the Student

What is discrete mathematics? Discrete mathematics is the part of mathematics devoted to the study of discrete objects. (Here *discrete* means consisting of distinct or unconnected elements.) The kind of problems solved using discrete mathematics include: How many ways are there to choose a valid password on a computer system? What is the probability of winning a lottery? Is there a link between two computers in a network? What is the shortest path between two cities using a transportation system? How can a list of integers be sorted so that the integers are in increasing order? How many steps are required to do such a sorting? How can a circuit be designed that adds two integers? You will learn the discrete structures and techniques needed to solve problems such as these.

More generally, discrete mathematics is used whenever objects are counted, when relationships between finite sets are studied, and when processes involving a finite number of steps are analyzed. A key reason for the growth in the importance of discrete mathematics is that information is stored and manipulated by computing machines in a discrete fashion.

There are several important reasons for studying discrete mathematics. First, through this course you can develop your mathematical maturity, that is, your ability to understand and create mathematical arguments. You will not get very far in your studies in the mathematical sciences without these skills. Second, discrete mathematics is the gateway to more advanced courses in all parts of the mathematical sciences. Math courses based on the material studied in discrete mathematics include logic, set theory, number theory, linear algebra, abstract algebra, combinatorics, graph theory, and probability theory (the discrete part of the subject). Discrete mathematics provides the mathematical foundations for many computer science courses including data structures, algorithms, data base theory, automata theory, formal languages, compiler theory, and operating systems. Students find these courses much more difficult when they have not had the appropriate mathematical foundations from discrete math. Also, discrete mathematics contains the necessary mathematical background for solving problems in operations research, including many discrete optimization techniques, chemistry, engineering, biology, and so on. In the text, we will study applications to some of these areas.

Key to the Exercises

no marking	a routine exercise
*	a difficult exercise
**	an extremely challenging exercise
→⃝	an exercise containing a result used in the text
(requires calculus)	an exercise whose solution requires the use of limits

Contents

1

The Foundations: Logic, Sets, and Functions 1

2

The Fundamentals: Algorithms, The Integers, and Matrices 77

3

Mathematical Reasoning 129

4

Counting 184

5

Advanced Counting Techniques 233

6

Relations 278

10

Modeling Computation 551

1

The Foundations: Logic, Sets, and Functions

This chapter reviews the foundations of discrete mathematics. Three important topics are covered: logic, sets, and functions. The rules of logic specify the precise meaning of mathematical statements. For instance, these rules tell us what statements such as, "There exists an integer that is greater than 100 that is a power of 2," and, "For every integer n the sum of the positive integers not exceeding n is $n(n + 1)/2$," mean. Furthermore, logic is the basis of all mathematical reasoning. Also, logic has practical applications to the design of computing machines.

Much of discrete mathematics is devoted to the study of discrete structures, which are used to represent discrete objects. All discrete structures are built up from sets, which are collections of objects. Examples of discrete structures built up from sets include combinations, which are unordered collections of objects used extensively in counting; relations, which are sets of ordered pairs that represent relationships between objects; graphs, which are sets of vertices and edges that connect vertices; and finite state machines, which are used to model computing machines.

The concept of a function is extremely important in discrete mathematics. A function assigns to each element of a set precisely one element of a set. Such useful structures as sequences and strings are special types of functions. Functions are used to represent the number of steps a procedure uses to solve a problem. The analysis of algorithms uses terminology and concepts related to the growth of functions. Recursive functions, defined by specifying their values at positive integers in terms of their values at smaller positive integers, are used to solve many counting problems.