理科类系列教材

PEARSON
Addison Wesley

改编版

Discrete Mathematics
(Fourth Edition)

离散数学（第4版）

- □ **Dossey**
- □ **Otto**
- □ **Spence**
- □ **Vanden Eynden**
- □ 俞正光 陆 玫 改编

原著

世界优秀教材中国版

理科类系列教材

改编版

Discrete Mathematics
(Fourth Edition)

离散数学（第4版）

John A. Dossey
Albert D. Otto
Lawrence E. Spence
Charles Vanden Eynden

Illinios State University

原著

俞正光　陆　玫　改编

清华大学

高等教育出版社
Higher Education Press

出版者的话

　　为适应当前我国高校各类创新人才培养的需要，大力推进教育部倡导的双语教学，配合教育部实施的"高等学校教学质量与教学改革工程"和"精品课程"建设的需要，我社开始有计划、大规模地开展了海外优秀理科系列教材的影印及改编工作。海外优秀教材在立体化配套、多种教学资源的整合以及为课程提供整体教学解决方案等方面对我们有不少可资借鉴之处。但一个不容忽视的问题是，外版教材与我国现行的教学内容、教学体系、教学模式和习惯等存在着巨大的差异。譬如，重点课程的原版教材通常很厚，内容很多，容量是国内自编教材的好几倍，国外的情况是，老师未必会都讲，剩下大量的内容留给学生自学；而国内的情况则不尽相同。受国内教学学时所限，完全照搬是不合时宜的。教材的国际化必须与本民族的文化教育传统相融合，在原有的基础上吸收国外优秀教材的长处，这使得我们需要对外文原版教材进行适当的改编。改编不是简单地使内容增删，而是结合国内教学特点，引进国外先进的教学思想，在教学内容和方式上更中国化，使之更符合国内的课程设置及教学环境。

　　在引进改编海外优秀教材的过程中，我们坚持了两条原则：1. 精选版本，打造精品系列；2. 慎选改编者，保证品质。

　　首先，我们和 Pearson Education, John Wiley & Sons, McGraw-Hill 以及 Thomson Learning 等国外出版公司进行了广泛接触，经推荐并在国内专家的协助下，提交引进版权总数达 200 余种，学科专业领域涉及数学、物理、化学化工、地理、环境等。收到样书后，我们聘请了国内高校一线教师、专家学者参与这些原版教材的评介工作，从中遴选出了一批优秀教材进行改编，并组织出版。这批教材普遍具有以下特点：(1) 基本上是近几年出版的，在国际上被广泛使用，在同类教材中具有相当的权威性；(2) 高版次，历经多年教学实践检验，内容翔实准确，反映时代要求；(3) 各种教学资源配套整

齐，为师生提供了极大的便利；(4) 插图精美、丰富，图文并茂，与正文相辅相成；(5) 语言简练、流畅，可读性强，比较适合非英语国家的学生阅读。

其次，慎选改编者。原版教材确定后，随之碰到的问题是寻找合适的改编者。要改编一本教材，必须要从头到尾吃透它，有这样的精力自编一本教材都绰绰有余了。我们与国内众多高等院校的众多专家学者进行了广泛的接触和细致的协商，几经酝酿，最终确定下来改编者。大多数改编者都是有国外留学背景的中青年学者，他们既有相当高的学术水平，又热爱教学，活跃在教学第一线。他们能够承担此任，不单是因为他们了解引进版教材的知识结构、表达方式和写作方法，更重要的是他们有精力、有热情，愿意付出，有的甚至付出了比写一本新教材更多的劳动。我们向他们表示最真诚的谢意。

在努力降低引进教材售价方面，高等教育出版社做了大量和细致的工作，这套引进改编的教材体现了一定的权威性、系统性、先进性和经济性等特点。

这套教材出版后，我们将结合各高校的双语教学计划，开展大规模的宣传、培训工作，及时地将本套丛书推荐给高校使用。在使用过程中，我们衷心希望广大高校教师和同学提出宝贵的意见和建议。如有好的教材值得引进，也请与高等教育出版社高等理科分社联系。

联系电话：010-58581384(数学)；010-58581374(物理)；010-58581380(化学化工)。E-mail: xuke@hep.com.cn。

高等教育出版社
2005 年 5 月

Preface

Today an increasing proportion of the applications of mathematics involves discrete rather than continuous models. The main reason for this trend is the integration of the computer into more and more of modern society. This book is intended for a one-semester introductory course in discrete mathematics.

Prerequisites   Even though a course taught from this book requires few formal mathematical prerequisites, students are assumed to have the mathematical maturity ordinarily obtained by taking at least two years of high school mathematics, including problem-solving and algorithmic skills, and the ability to think abstractly.

Approach   This book has a strong algorithmic emphasis that serves to unify the material. Algorithms are presented in English so that knowledge of a particular programming language is not required.

Choice of Topics   The choice of topics is based upon the recommendations of various professional organizations, including those of the MAA's Panel on Discrete Mathematics in the First Two Years, the NCTM's *Principles and Standards for School Mathematics*, and the CBMS's recommendations for the mathematical education of teachers.
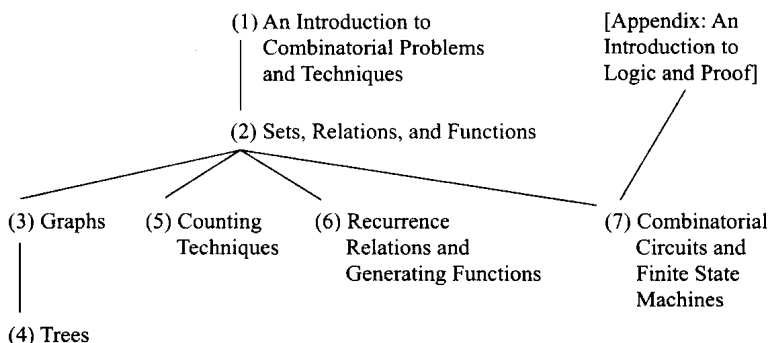
Flexibility   Although designed for a one-semester course, the book contains more material than can be covered in either one semester or two quarters. Consequently, instructors will have considerable freedom to choose topics tailored to the particular needs and interests of their students. Users of previous editions have reported considerable success in courses ranging from freshman-level courses for computer science students to upper-level courses for mathematics majors. The present edition continues to allow instructors the flexibility to devise a course that is appropriate for a variety of different types of students.

Changes in the Fourth Edition   At the suggestion of several users of the third edition, additional historical comments have been added; these are included at the end of each chapter. In addition, Chapters 3 and 4 have been rewritten so as to give the breadth-first search algorithm a more prominent role. (It now appears in Section 3.3 and is used in Sections 3.4 and 4.2.) Many examples in Chapters 3 and 4 have also

been rewritten to be more useful to instructors who do not wish to discuss the details of the formal presentations of the algorithms. These examples now precede the algorithms and better reveal how the algorithms work without requiring discussion of the formal algorithms themselves. The previously separate sections on spanning trees and minimal and maximal spanning trees have been combined into a new Section 4.2, and the introductory material on matrices has been removed from Chapter 3. Another new appendix (Appendix B) describes the looping and branching structures used in the book's algorithms. Additional changes to the exposition have also been made throughout the book to improve the clarity of the writing.

Exercises   The exercise sets in this book have been designed for flexibility. Many straightforward computational and algorithmic exercises are included after each section. These exercises give students hands-on practice with the concepts and algorithms of discrete mathematics and are especially important for students whose mathematical backgrounds are weak. Other exercises extend the material in the text or introduce new concepts not treated there. Exercise numbers in color indicate the more challenging problems. An instructor should choose those exercises appropriate to his or her course and students. Answers to odd-numbered computational exercises appear at the end of the book. At the end of each chapter, a set of **Supplementary Exercises** is provided. These reprise the most important concepts and techniques of the chapter and also explore new ideas not covered elsewhere.

Chapter Independence   The sequence of chapters allows considerable flexibility in teaching a course from this book. The following diagram shows the logical dependence of the chapters. Although this book assumes only the familiarity with logic and proof ordinarily gained in high-school geometry, an appendix (Appendix A) is provided for those who prefer a more formal treatment. If this appendix is covered, it may be taught at any time as an independent unit or in combination with Chapter 7.

```
        (1) An Introduction to          [Appendix: An
          | Combinatorial Problems       Introduction to
          | and Techniques               Logic and Proof]
          |                                 /
        (2) Sets, Relations, and Functions /
       __/____/_____    /
      /      /      \                  \ /
(3) Graphs  (5) Counting  (6) Recurrence  (7) Combinatorial
  |         Techniques     Relations and      Circuits and
  |                        Generating Functions  Finite State
  |                                              Machines
(4) Trees
```

Chapters 1 and 2 are introductory in nature. Chapter 1, which should be covered fairly quickly, gives a sampling of the sort of discrete problems the course treats. Some questions are raised that will not be answered until later in the book. Section 1.4 contains a discussion of complexity that some instructors may want to omit or delay until students have had more experience with algorithms. An instructor may wish to cover only the illustrative algorithms in this section that are most relevant to his or her students.

Chapter 2 reviews various basic topics, including sets, relations, functions, and mathematical induction. It can be taught more or less rapidly depending on the mathematical backgrounds of the students and the level of the course. It should be possible for students with good mathematics backgrounds to be able to read much of Chapter 2 on their own. The remaining chapters are, as the diagram shows, independent except that Chapter 4 depend on Chapter 3.

Possible Courses   A course emphasizing graph theory and its applications would cover most of Chapters 3–4, while a course with less graph theory would concentrate on Chapters 5–7.

Courses of various levels of sophistication can be taught from this book. For example, the topic of computational complexity is of great importance, and so attention is given to the complexity of many algorithms in this text. Yet it is a difficult topic, and the detail with which it is treated should correspond to the intended level of the course and the preparation of students.

Computer Projects   Each chapter ends with a set of computer projects related to its content, algorithmic and otherwise. These are purposely stated in general terms, so as to be appropriate to students using various computing systems and languages.

Supplements   A *Student's Solution Manual*, available for purchase by students, provides detailed, worked-out solutions to the odd-numbered exercises (ISBN 0-201-75483-5). An *Instructor's Answer Manual*, containing answers to all even-numbered computational exercises, is also available (ISBN 0-201-75482-7).

the country, as well as by the following list of colleague reviewers: Veena Chadha, University of Wisconsin-Eau Claire; Gary Chartrand, West Michigan University; Tilak de Alwis, Southeastern Louisiana University; Ron Dalla, Eastern Washington University; George Dimitroff, Evergreen State College; Gayla S. Domke, Georgia State University; Jerome Eisenfeld, University of Texas at Arlington; Kathleen Elder, Frostburg State University; Klaus Fischer, George Mason University; Donald A. Goral, Northern Virginia Community College; Natasa Jonoska, University of South Florida; Thomas Kiley, George Mason University; Theresa D. Magnus, St. Mary's College; Chris Mawata, University of Tennessee; Robert C. Mers, North Carolina A & T University; Catherine M. Murphy, Purdue University, Calumet; Anne Quinn, Edinboro University of Pennsylvania; Steen Pedersen, Wright State University; Dix H. Pettey, University of Missouri, Columbia; James L. Richards, Bemidji State University; A. Allan Riveland, Washburn University; Mohan Shrikhande, Central Michigan University; Allan Schwenk, Western Michigan University; and our colleague, Roger Eggleton.

We wish to offer special thanks to Michael Plantholt and Dean Sanders of Illinois State University, who independently checked all the algorithms in the third edition for both correctness and readability. Their suggestions led to substantial revisions and improvements of the algorithms.

Changes made for this fourth edition were guided by the comments of the following reviewers: Mark Ferris, Midwestern State University; Johanne Hattingh, Georgia State University; Colleen Hoover, St. Mary's College; Jason Miller, Truman State University; and Richard Rockwell, Pacific Union College.

We also appreciate the excellent editorial work of Cindy Cody and Kevin Bradley during the production process.

John A. Dossey
Albert D. Otto
Lawrence E. Spence
Charles Vanden Eynden

To the Student

This book is concerned with the *discrete*, that is, finite processes and sets of elements that can be listed. This contrasts with calculus, which has to do with infinite processes and intervals of real numbers.

Although discrete mathematics has been around for a long time, it has enjoyed a recent rapid expansion, paralleling the growth in the importance of computers. A digital computer is a complicated, but essentially finite, machine. At any given time it can be described by a large, but finite, sequence of 0s and 1s, corresponding to the internal states of its electronic components. Thus discrete mathematics is essential in understanding computers and how they can be applied.

An important part of discrete mathematics has to do with *algorithms*, which are explicit instructions for performing certain computations. You first learned algorithms in elementary school, for arithmetic is full of them. For example, there is the *long division algorithm*, which might cause an elementary school student to write down something like the following tableau.

$$
\begin{array}{r}
32 \\
13)\overline{425} \\
\underline{39} \\
35 \\
\underline{26} \\
9
\end{array}
$$

Internally, the student is applying certain memorized procedures: *There are three 13s in 42, 3 times 13 is 39, 42 minus 39 is 3, bring down the 5*, etc. These procedures comprise the algorithm.

Another example of an algorithm is a computer program. Suppose a small business wants to identify all customers who owe it more than $100 and have been delinquent in payments for at least 3 months. Even though the company's computer files contain this information, it constitutes only a small portion of their data. Thus a program must be written to sift out exactly what the company wants to know. This program consists of a precise set of instructions to the computer, covering all possibilities, that causes it to isolate the desired list of customers.

Our two examples of algorithms are similar in that the entity executing the algorithm does not have to understand why it works. Students in elementary school generally do not know why the long division algorithm gives the correct answer, only what the proper steps are. Of course, a computer doesn't understand anything; it just follows orders (and if its orders are incorrect, so that the program is wrong, the computer will dutifully produce the wrong answer).

If you are taking a course using this book, however, you are no longer in elementary school and you are a human being, not a computer. Thus you will be expected to know not only *how* our algorithms work, but *why*.

We will investigate some algorithms you probably have never seen before. For example, suppose you are planning to drive from Miami, Florida, to Seattle, Washington. Even if you stick to the interstate highways, there are hundreds of ways to go. Which way is the shortest? You might get out a map and, after playing around, find a route you *thought* was shortest, but could you be sure?

There is an algorithm that you could apply to this problem that would give you the correct answer. Better yet, you could program the algorithm into a computer, and let it find the shortest route. That algorithm is explained in this book.

We will be interested not only in the how and why of algorithms, but also in the *how long*. Computer time can be expensive, so before we give a computer a job to do we may want an estimate of how long it will take. Sometimes the surprising answer is that the computation will take so long as to make a computer solution impractical, even if we use the largest and fastest existing machines. It is a popular but incorrect idea that computers can do any computation. No computer can take the data from the world's weather stations and use it to predict future weather accurately more than a few days in advance. The fact that no one knows how to do certain computations efficiently can actually be useful. For example, if n is the product of two primes of about 150 decimal digits, then to factor n takes hundreds of years (even using the best methods and computers known), and this is the basis of an important system of cryptography.

You have probably already heard a number of times that mathematics is not a spectator sport, and that the only way to learn mathematics is by doing it. There is an important reason we are repeating this advice here. IT'S TRUE! Moreover, it's the best thing we know to tell you. You can't learn to play the guitar or shoot free throws just by watching someone else do these things, and you can't learn discrete mathematics just by reading this book or attending lectures. The mind must be in gear and active. When reading a mathematics book, you should always have paper and pencil handy to work out examples and the details of computations. When attending a mathematics lecture, it is best if you have read the material already. Then you can concentrate on seeing if your understanding of the content agrees with that of the professor, and you can ask questions about any difficult points.

Of course, one of the best ways to be active in learning mathematics is by doing exercises. There are many of these in this book. Some are purely computational, others test understanding of concepts, and some require constructing proofs. Answers to odd-numbered computational exercises are in the back of the book, but *don't look before you have determined your own answer.* If your work consistently gives the

same answer as in the back of the book, then you can have confidence that you are on the right track.

Some exercises are harder than others. The more time you spend on such exercises, the more you learn. There is a common notion (reinforced by some courses) that if you can't figure out how to do a problem in five minutes you should go on to the next problem. This attitude becomes less and less relevant the more skillful you become. Very few accomplishments of any importance can be done in five minutes.

Many students do not realize the importance of learning the technical language of what they are studying. It is traditional in mathematics to assign special meaning to short, common words such as *set, function, relation, graph, tree, network*. These words have precise definitions that you must learn. Otherwise, how can you understand what you read in this book, or what your professor is saying? These technical words are necessary for efficient communication. How would you like to explain a baseball game to someone if you were not allowed to use the particular language of that sport? Every time you wanted to say that a pitch was a *ball*, you would have to say that it was a pitch that was not in the strike zone and that the batter didn't swing at it. For that matter, *strike zone* is a technical term that would also need an explanation in each instance. Communication on such a basis would be almost impossible.

Finally, proper terminology is necessary to share information in a useful way with others. Mathematics is a human endeavor, and human cooperation depends on communication. In the real world, it is seldom sufficient simply to figure something out. You must be able to explain it to other people, and to convince them that your solution is correct.

We hope your study of discrete mathematics is successful, and that you get from it techniques and attitudes that you will find useful in many contexts.

算法名词

ALGEBRAIC AND COMBINATORIAL ALGORITHMS
代数与组合算法

Algorithm for Evaluating x^n

求 x^n 的值的算法

Algorithm for Evaluating $n!$

求 $n!$ 的值的算法

Algorithm for Evaluating $P(n, r)$

求 $P(n, r)$ 的算法

Algorithm for the Sum of an Arithmetic Progression

算术级数和的算法

Algorithm for the Sum of An Geometric Progression

几何级数和的算法

Algorithm for Evaluating the nth Fibonacci Number

求第 n 个斐波那契数的算法

Horner's Polynomial Evaluation Algorithm

Horner 多项式计数算法

Revised Polynomial Evaluation Algorithm

修正多项式计数算法

GRAPH-THEORETIC ALGORITHMS
图论算法

Breadth-First Search Algorithm

广度优先搜索算法

Depth-First Search Algorithm

深度优先搜索算法

Dijkstra's Algorithm

Dijkstra 最短路算法

Euler Circuit Algorithm

欧拉回路算法

Huffman's Optimal Binary Tree Algorithm

Huffman 最优二分树算法

Inorder Traversal Algorithm

向内横越算法

Kruskal's Algorithm

Kruskal 最小生成树算法

Postorder Traversal Algorithm

向后横越算法

Preorder Traversal Algorithm

向前横越算法

Prim's Algorithm

Prim 算法

Prufer's Algorithm

Prufer 算法

SEARCHING AND SORTING ALGORITHMS
检索与分类算法

Binary Search Tree Construction Algorithm

二分搜索树构造算法

Binary Search Tree Search Algorithm

二分搜索树搜索算法

Bubble Sort Algorithm

冒泡排序算法

郑 重 声 明

高等教育出版社依法对本书享有专有出版权。任何未经许可的复制、销售行为均违反《中华人民共和国著作权法》，其行为人将承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。为了维护市场秩序，保护读者的合法权益，避免读者误用盗版书造成不良后果，我社将配合行政执法部门和司法机关对违法犯罪的单位和个人给予严厉打击。社会各界人士如发现上述侵权行为，希望及时举报，本社将奖励举报有功人员。

反盗版举报电话：(010) 58581897/58581896/58581879

传　　真：(010) 82086060

E - mail：dd@hep.com.cn

通信地址：北京市西城区德外大街 4 号
　　　　　　高等教育出版社打击盗版办公室

邮　　编：100011

购书请拨打电话：(010)58581118

数学类引进版教材目录(已出版)

书　号	书　名	著译者
7040063948	微积分(翻译版)	胡乃闶、邵勇等译
7040118750	多元微积分(翻译版)	董达英、高孟宁等译
7040063964	数学的原理与实践(翻译版)	申大维、方丽萍等译
7040078910	统计学：基本概念和方法(翻译版)	吴喜之、程博等译
7040087014	时间序列的理论与方法(第 2 版) (翻译版)	田铮译
7040142538	数学史通论(第 2 版)(翻译版)	李文林、邹建成等译
7040108232	托马斯微积分(第 10 版) (附光盘 1 片) (翻译版)	叶其孝、王耀东等译
7040144247	托马斯微积分(第 10 版)(上册)(附光盘 1 片)(影印版)	Finney Weir Giordano
7040144255	托马斯微积分(第 10 版)(下册)(影印版)	Finney Weir Giordano
7040140039	微积分(第 5 版)(上册)(附光盘 2 片)(影印版)	Stewart
7040140047	微积分(第 5 版)(下册)(影印版)	Stewart
7040162318	微积分及其在商业、经济、生命科学及社会科学中的应用(第 9 版)(影印版)	Raymond A. Barnett
7040140063	高等工程数学(第 5 版)(上册)(影印版)	Peter V O'Neil
7040140055	高等工程数学(第 5 版)(下册)(影印版)	Peter V O'Neil
7040155583	简明微积分及其应用(第 2 版)(影印版)	Bill Amstrong
7040154862	微积分(第 2 版)(上册)(影印版)	Robert T. Smith
7040154870	微积分(第 2 版)(下册)(影印版)	Robert T. Smith
7040155567	抽象代数(影印版)	Joseph J. Rotman
7040155575	数理统计学导论(第 5 版)(影印版)	Robert V. Hogg
7040155591	统计学导论(第 6 版)(附光盘 1 片)(影印版)	Neil A. Weiss
7040155605	概率论与数理统计(第 5 版)(附光盘 1 片)(影印版)	Jay L. Devore
7040165457	应用多元统计分析方法(附光盘 1 片)(影印版)	Dallas E. Johnson
7040163802	应用线性回归模型(第 4 版)(附光盘 1 片)(影印版)	Kutner
7040163233	经典和现代回归分析及其应用(第 2 版)(影印版)	Raymond H. Myers
704016230X	离散数学及其应用(第 3 版)(影印版)	Susanna S. Epp
7040167336	线性代数(第 4 版)(改编版)	Stephen H. Friedberg 清华大学 王殿军等改编
7040163810	线性代数及其应用(第 8 版)(改编版)	Bernard Kolman 清华大学 王殿军等改编
7040166321	离散数学(第 4 版)(改编版)	John A. Dossey 清华大学 俞正光等改编
7040167328	离散数学和组合数学(第 2 版)(改编版)	James A. Anderson 清华大学 俞正光等改编

目　录