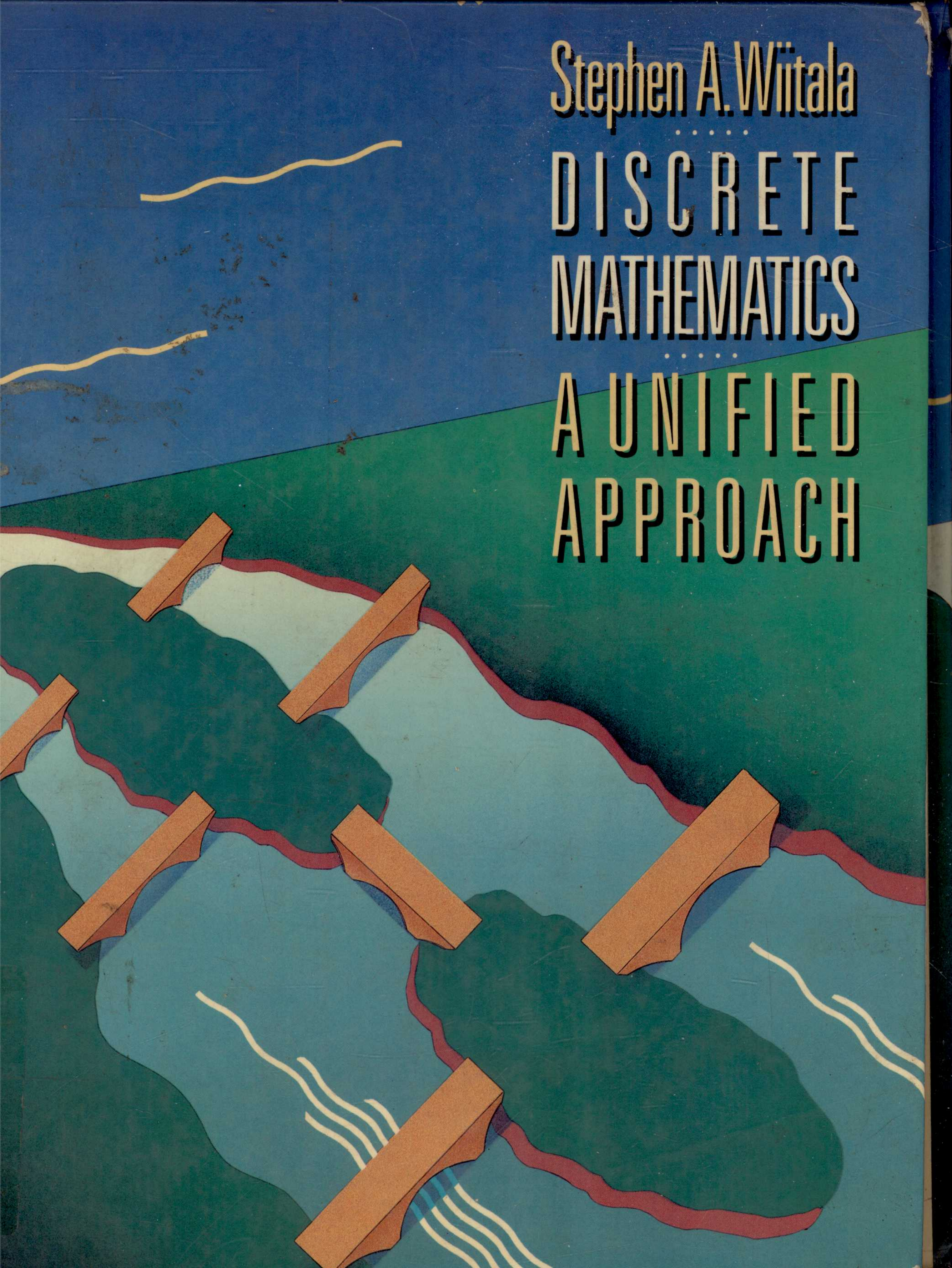
Stephen A. Wiitala

DISCRETE MATHEMATICS

A UNIFIED APPROACH



DISCRETE MATHEMATICS
A Unified Approach

Stephen A. Wiitala
Associate Professor of Mathematics
Norwich University

McGraw-Hill Book Company
New York St. Louis San Francisco Auckland Bogotá Hamburg
Johannesburg London Madrid Mexico Milan Montreal New Delhi
Panama Paris São Paulo Singapore Sydney Tokyo Toronto

DISCRETE MATHEMATICS: A Unified Approach

1234567890 HALHAL 89432109876

ISBN 0-07-070169-5

This book was set in Times Roman.
The editor was Kaye Pace;
the designer was Charles A. Carson;
the cover illustrator was John Hite;
the production supervisor was Diane Renda;
the drawings were done by Volt Information Sciences, Inc.
Arcata Graphics/Halliday was printer and binder.

The front and back covers of the book are meant to represent the Königsberg bridge problem.

Library of Congress Cataloging-in-Publication Data

Wiitala, Stephen A. (date).
Discrete mathematics.

1. Mathematics—1961– . 2. Electronic data processing—Mathematics. I. Title.
QA39.2.W53 1987 510 86-21031
ISBN 0-07-070169-5

ABOUT THE AUTHOR

Stephen A. Wiitala was born in Vancouver, Washington, in 1946. He attended Western Washington University, receiving a B.A. degree in 1968. Following a brief period as a high school mathematics teacher, he entered graduate school in 1970, receiving an M.A. in mathematics from Western in 1971. He continued his graduate work at Dartmouth College, receiving a Ph.D. in mathematics in 1975. His dissertation dealt with geometric algebras defined over finite fields; and in studying these topics, he first became involved with combinatorics and the interaction between computer science and mathematics.

His first college teaching position was at Nebraska Wesleyan University, where he taught mathematics and computer science. In 1980 he moved to Norwich University, in Northfield, Vermont, where he became responsible for the development of a mathematics-based computer science program and for a discrete mathematics course designed for sophomore mathematics and computer science majors. This book is the result of the time spent in developing that course.

Professor Wiitala currently serves as the program director for the computer science mathematics program at Norwich University and has also served as the director of academic computing.

PREFACE

Discrete mathematics has become a popular introductory course for students contemplating the study of either computer science or mathematics.

As a topic in mathematics, discrete mathematics presents a convenient introduction to the techniques of proof and the process of logical reasoning but allows the student to focus attention on concepts which are relatively concrete, as opposed to some of the ideas in the calculus. As a topic for computer science, it provides much of the mathematical foundation for the later study of theoretical topics as well as a foundation for many of the more concrete ideas. Thus, the topics in discrete mathematics present the opportunity for many illustrative examples.

Students of mathematics should see some of the applications of mathematics in other disciplines, and computer science students need to see the mathematical foundations of their discipline from the beginnings of their study.

This book stresses two themes. The first is the application of the topics to computer science. In particular, whenever feasible, algorithms are included to accomplish the processes that are being described. The second is an introduction to the methods of theoretical mathematics. The mathematical topics covered stress the development of concepts and the process of proof.

There is a reason for my choice of this particular title, *Discrete Mathematics: A Unified Approach*. Although a strong relationship exists between most of the topics in discrete mathematics, in attempting to make their texts as flexible as possible, too many authors present the ideas as a compendium of separate ideas which have no relationship to each other. The selection of topics and order of presentation in this book is designed to allow for a more unified presentation of the material. It is possible to use the material in this book in a different order, but I feel that the present order provides the student with the best sense of unity.

Ample material is given here for a two-semester or three-quarter course in discrete mathematics. At Norwich we have covered Chapters 1, 2, 4, 5, and parts of 6 in the first semester, and the remainder of Chapter 6 and Chapters 7, 8, and 9 in the second semester. Chapter 3 is covered in another course but could be used in either semester. Chapter 4 can be omitted if the students are already familiar with matrix arithmetic.

Suggested course organizations:

One semester (stressing Graph Theory): Chapters 1, 2, 4 (if needed), and 5, and Sections 6.1 to 6.5.

One semester (stressing Theory of Computation): Chapter 1; Sections 2.1 to 2.5, 5.1 to 5.4, 6.1, 6.2, 6.6, 6.9, and 7.1 to 7.4; and Chapters 8 and 9.

Two semesters: First semester, Chapters 1 to 5; second semester, Chapters 6 to 9.

Brief Description of Chapters

Chapter 1: An introduction to the ideas of mathematical logic. This chapter lays the foundation for the techniques of proof used in the later chapters. It provides an introduction to predicate logic and mathematical induction. The presentation of mathematical induction is based on predicate logic.

Chapter 2: An introduction to mathematical set theory and combinatorial analysis. The combinatorial analysis is presented at a relatively low level and is used in the discussion of algorithm analysis later in the text.

Chapter 3: A brief introduction to boolean algebra. On the basis of similarities already observed between logic and set theory, the concept of a boolean algebra as a mathematical abstraction is introduced. The chapter begins with a discussion of logic circuits. This chapter is independent of those that follow, and can be used at any time after Chapter 2, or it may be omitted altogether.

Chapter 4: A brief introduction to matrix arithmetic. The barest essentials of matrix addition, scalar multiplication, and matrix multiplication are described for the student who has not encountered these topics before. They are needed in order to make use of the matrix algorithms in graph theory in the following chapter. Students who have already been introduced to matrices can skip this chapter.

Chapter 5: The theory of undirected graphs. This chapter and Chapter 6 form the heart of my discrete mathematics course. Chapter 5 presents the terminology and algorithms needed for the applications of graph theory to computer science. Several of the classic ideas, such as connectivity, chromatic number, and planarity are discussed.

Chapter 6: The theory of directed graphs. The ideas of undirected graphs are extended to that of graphs in which the edges have a specified direction. Discussion of applications of graphs to computer science and other areas is included, as well as the important algorithms of Dijkstra and Warshall. Directed (rooted) trees are presented in this chapter as a special case of the directed graph.

Chapter 7: An introduction to finite automatons and formal languages as described by them. The state diagrams of finite automatons are used to illustrate another example of the use of directed graphs. The famous syntax diagrams of Pascal are discussed as an example of the finite automaton.

Chapter 8: An introduction to Turing machines. Pushdown automata are briefly discussed as an extension of the finite automaton, and these ideas are extended to the Turing machine. A brief discussion of Church's thesis is included.

Chapter 9: The culmination of the book is a discussion of the pumping lemmas and Turing's halting problem. These theorems enable us to put most of the ideas in the book together to illustrate their use in producing some significant results.

The last three chapters are written at a level which is accessible to freshmen and sophomores, and as a consequence some of the mathematical details have been left out in some cases.

Acknowledgments

I would like to thank the many people who made the process of writing this book a lot easier than I thought it might be, particularly the staff at McGraw-Hill who provided an incredible amount of assistance.

I must thank those generations of Norwich University students who used my manuscript as a textbook and provided useful feedback to me in the process; and Prof. Gerard LaVarnway, who used an early draft of Chapters 5 and 6 in his section of discrete mathematics. I also thank Prof. Bruce Edwards of the University of Florida (and a graduate school colleague) for encouraging me to start this project when it was merely the glimmer of an idea.

I appreciate the help and direction given me by those who read my manuscript and provided extremely useful suggestions for revision: Lynne Doty, Marist College; Joe Hemmeter, University of Delaware; Burton Kleinman, Queensborough Community College; Robert McGuigan, Westfield State College; Kay Smith, St. Olaf College; J. W. Smith, University of Georgia; and Jerry Waxman, Queens College.

Finally, I thank my family for putting up with me through the process of writing and producing this book. They had to endure a number of late nights with the printer running on the computer, and then cope with a grouch the next morning on more than one occasion.

Stephen A. Wiitala

CONTENTS

CHAPTER

1 Mathematical Logic and Proofs

1.1 WHAT IS A PROOF?

The Basic Problem

In mathematics and computer science, as well as in many places in everyday life, we face the problem of determining whether something is true. Often the decision is easy. If we were to say that $2 + 2 = 5$, most people, no doubt, would immediately say that the statement was not true. (Actually, the more likely response would be, "What kind of dummy would think that?" or something even more insulting.) If we were to say $2 + 2 = 4$, then undoubtedly the response would be, "Of course" or "Everybody knows that." However, many statements are not so clear. A statement such as "The sum of the first n odd integers is equal to n^2," in addition to meeting with a good deal of consternation, might be greeted with a response of "Is that really true?" or "Why?" This natural response lies behind one of the most important concepts of mathematics, that of proof. In this chapter we explore the ideas of mathematical logic which lie behind the concept of proof.

If a statement is obviously true or false, we usually don't worry too much about proof or disproof (although some of the most difficult problems in mathematics are those whose truth may seem obvious). But for those statements whose truth is unknown, it is useful to be able to determine exactly what the situation is. Many people find that the process of discovering a proof is more interesting than the statement being proved. The well-known pythagorean theorem of geometry has been known to be true for thousands of years, but hundreds of proofs have been found for it, and even today people enjoy attempting to find new proofs of this theorem. The people who find these proofs are not concerned about whether the statement is true; they are really interested in why it is true.

This interest in the reasons why statements are true or false is one of the things that makes mathematics fascinating. The mathematician is often concerned not only with *what* is true but also with *why* things are true.

As we proceed through this book, we will encounter many statements about mathematics and computer science which will require proof or disproof. With a solid grounding in mathematical logic, we should be able to proceed with some degree of confidence. One warning should be given though. Some things which are "obviously true" are not so obvious when we try to prove them—in fact, some turn out to be false! One of the more bizarre facts of logic tells us that it is even possible that some statements could be true but impossible to prove from the current axioms of mathematics.

In computer science, as in mathematics, it is crucial to *know* what we are doing is correct. In algorithm analysis, for example, we are concerned with knowing that the algorithm does what it is supposed to do and that it does its job efficiently. To correctly address those issues, it is often necessary to use the formalism that mathematics can provide in order to verify that the statements we wish to make about the algorithms in question are correct.

Many mathematical structures are used in computer science. These structures are often best described in terms of the formal theory of mathematics, and in this context proofs play a very important role. Graph theory, for example, provides the foundation for many ideas in data structures, and it forms the basis for some of the theoretical models of a computer which are useful in understanding how computers are able or unable to perform certain tasks.

Some Examples

To begin considering the ideas of mathematical logic and proofs, we look at a few examples of reasoning, some of which represent valid reasoning and others which do not. As we proceed through the chapter, we will be able to discern for certain which arguments are valid and which are not.

EXAMPLE 1.1

Consider the following argument:

If computers are to really find their way into the home of the average person, then the price of some complete systems should be less than $1000. The local computer store has complete systems priced under $1000, so it must follow that computers have found their way into the average home. ■

We need to ask whether the argument convinces us that the conclusion is justified by the evidence. If the conclusion does "follow logically," then we say that the argument *proves* that the conclusion is indeed true. Sometimes we find that the conclusion is true even when the argument is not "logical."

EXAMPLE 1.2

If computers are to find their way into the home of the average person, then the price of some complete systems should be less than $1000. Computers are found in the homes of average people, thus we must conclude that there are complete computer systems which are priced under $1000. ■

EXAMPLE 1.3

All computers have input devices. The Macintosh is a computer. Thus the Macintosh has an input device. ■

EXAMPLE 1.4

All Apple computers can be connected to printing devices. All minicomputers can be connected to printing devices. Thus, some Apple computers are minicomputers. ■

In an argument, the statements used to build the conclusion are called the *premises*, and the final statement is called the *conclusion*. We regard an argument as being valid provided that whenever all the premises are true, the conclusion is guaranteed to be true.

A couple of important facts need to be recognized. First, a valid argument in which some of the premises are false may or may *not* produce a valid conclusion, second an invalid argument (one that is not valid) *can* produce a true conclusion. In the second case, the arriving at a true conclusion is more a chance happening than anything else.

In the problems for this section, we ask you to consider some arguments and decide whether you think they are valid. There obviously cannot be right or wrong answers at this point, since we have no techniques for determining whether an argument is valid. The point is to get you thinking about the problems involved.

Problem Set 1

The following problems should be answered based on your intuitive knowledge of logic. Much of the rest of this chapter is devoted to providing techniques to answer these questions. These problems should get you started thinking about the ideas.

1. Consider the argument in Example 1.1. Do you believe that it is a convincing argument? Why or why not? Do you believe the conclusion?
2. Answer the questions in Problem 1 for Example 1.2.
3. Answer the questions in Problem 1 for Example 1.3.
4. Answer the questions in Problem 1 for Example 1.4.
5. Explain why it is reasonable that a valid argument with false premises produces conclusions which may or may not be true.
6. We said that an invalid argument could lead to a true conclusion. How could this happen? What does this tell us about the process of proof?
7. Analyze the following argument, which is sometimes called the "apple press" argument:

   Apple presses eat apples.

   Johnny eats apples.

   Thus Johnny must be an apple press.

   Obviously this is not a valid argument. Explain why.

In Problems 8 to 15, determine whether the argument given justifies the conclusion stated and explain why you think that this is the case.

8. If I get the job and work hard, then I will be promoted. I was promoted. Thus I got the job.
9. If I get the job and work hard, then I will be promoted. I was not promoted. Thus either I did not get the job or I did not work hard.
10. I will either get an A in this course or I will not graduate. If I don't graduate, I will go into the army. I got an A. Thus, I won't go into the army.
11. I will either get an A in this course or I will not graduate. If I don't graduate, I will go into the army. I got a B. Thus, I will go into the army.
12. If the football game runs late, then *60 Minutes* will be delayed. If *60 Minutes* is delayed, then the local news will not start until after 11:00. The local news started at 11:15, so the football game ran late.
13. If I buy a new car, then I will not be able to go to Florida in December. Since I am going to Florida in December, I will not buy a new car.
14. Either the butler or the maid committed the crime. If the butler did it, he would not have been able to answer the phone at 11:00. Since he did answer the phone at 11:00, the maid must have done it.

15. If the temperature had gone down, Bill would not have gone to the parade. Since Bill did not go to the parade, the temperature must have gone down.

***16.** Consider the following pair of statements:
(a) The statement labeled (b) is false.
(b) The statement labeled (a) is true.
Can either statement above be true? Why or why not?

***17.** Consider the statement "I am telling a lie." Can this statement be true? Can it be false?

1.2 PROPOSITIONS

Mathematically Useful Statements

As we begin to study the logic used in mathematics, we should note first two commonsense ideas which are built into the system of logic that we use and the reasons why we use these ideas.

First, mathematically interesting statements are those which can be shown to be either true or false—or which at least could be shown to be true or false if some additional information were known regarding variables used in the statement. Statements like "If x is an odd number and y is an odd number, then $x + y$ is an odd number" and "The sum of the first n odd integers is n^2" are examples of statements which are mathematically interesting, because they can be shown to be either true or false with no concern about the values of the variables involved. The statement $x + y = 4$, on the other hand, would be mathematically interesting also, provided that some kind of additional information was given regarding the variables x and y. Do we mean to say this is true for all values of x and y (unlikely)? Or, are we asserting that for all values of x we can find a value of y such that the statement is true? Or, do we mean to assert that this statement is true when $x = 2$ and $y = 3$? In any case, once we make a decision about what we intend to do with x and y, we come up with a statement that will be either true or false.

However, such statements as "Three is a pretty number" or "Fred is tall," though certainly legitimate in English, leave much to be desired mathematically. We have no idea what a pretty number is, and the decision as to whether 3 is pretty is quite subjective. Similarly, unless Fred is 10 feet tall, or 3 feet tall, it will be pretty difficult to come to any general agreement as to whether that statement is true or not.

Second, a mathematical statement is either true or false but never both. It is interesting to consider a statement like "This statement is false," which is true if it is false and false if it is true. But such statements are mathematical curiosities and are not the kind of thing to which we want to devote a lot of time in this book.

There are some rather interesting consequences from these two concepts. The most obvious is usually called the *law of the excluded middle.* If a statement is not true, then it is false (or if it is not false, it is true). This seemingly obvious fact is used very conveniently in some proof techniques, because sometimes it is easier to prove that a statement is not false than it is to prove that the statement is true. The other consequence of this idea is the reverse of what we referred to above: If a statement is true, it is not false; and if a statement is false, it is not true.

Despite the fact that the statements we made above would seem reasonable to most of us, there are people who, for various philosophical reasons, do not accept all that we stated above. In particular, there are mathematicians (known as *intuitionists*) who do not accept the idea that a statement has been proved true if it has been demonstrated that the statement is not false. They feel that there is a difference between not false and true. Actually in everyday life we sometimes use that kind of reasoning as well. People sometimes say, for example, when asked how they feel about a decision that affects them, that they are "not unhappy." A little reflection might suggest that this is a weaker statement than saying that they are happy. The statement that they are happy *should* be true, but maybe it really is in some kind of intermediate state. The mathematical logic of the intuitionists is interesting, but for our purposes we stick with the law of the excluded middle for the things that we do in this book.

Propositional Logic

We begin our study of logic by formally defining the idea of a *proposition.*

DEFINITION

A **proposition** is a statement which, as given, is either true or false.

One of the goals of mathematics is to determine which propositions are true and which are false.

DEFINITION

A **theorem** is a proposition which has been proved to be true about a mathematical system.

Most, but not all, theorems are in the form "if something, then something else." In propositional logic, we learn how to work with statements of that form in order to better understand how to prove theorems. When claiming that a statement is a proposition, we need not *know* whether it is true or false; we only need to know that it is the kind of statement which falls into one or the other of those categories.

The statement "For any integer value n for which $n > 2$, the equation $a^n + b^n = c^n$ has no solutions in which a, b, and c are all nonzero integers" is obviously either true or false. The fact that years of mathematical effort have not resolved the issue as to which category that statement belongs does not alter the fact the statement is a proposition.[1]

EXAMPLE 1.5

The following are all propositions:
(a) There are two solutions to the equation $x^2 + 4 = 20$, and both solutions are integers.
(b) Either this program runs, or there was an error in keying in the data.
(c) It is not the case that 5 is a prime number.
(d) If x is any integer, then x^2 is a positive integer.
(e) Every integer is the sum of four perfect squares. ■

EXAMPLE 1.6

The following are examples of statements that are *not* propositions.
(a) $x^2 = 11$.
(b) This is a bad program.
(c) Go for it! ■

Our first approach to mathematics is a study of propositional logic. The basic building blocks of propositional logic are, naturally enough, propositions. In propositional logic, the goal is to study the ways of combining propositions to form new propositions and to determine under what circumstances these new "compound" propositions are true. Later we turn to a more inclusive form of logic, which will enable us to use statements like Example 1.6a and in effect turn them into propositions.

Notation

To describe the processes of mathematical logic, we need a way of symbolizing propositions. It has become a tradition of sorts in elementary propositional logic to use lowercase letters starting from p and continuing as needed (q, r, s, etc.) to represent propositions when we wish to study logic in a symbolic fashion. (The reason for the use of p is that p is the first letter in the word "proposition.")

In indicating that we will use a particular letter to stand for a proposition, we will use the following kind of notation:

$$p\text{: If } x \text{ is any integer, then } x^2 \text{ is positive or zero.}$$

[1] This proposition was stated as a theorem by the famous French mathematician Pierre de Fermat (1601–1665) and has been the subject of intense scrutiny ever since. But to this day it remains one of the most puzzling unsolved problems in mathematics.

This means that we would denote the proposition in Example 1.5a by p.

Often these symbols are used as *propositional variables*, that is, as symbols that could represent any proposition, rather than standing for a particular proposition.

In the next section, we begin to explore the process of combining several propositions into one, which is a first step to explaining the process of proof. Remember that our goal in this chapter is to develop the mathematical techniques which we need to prove that a particular proposition is true (or false).

Problem Set 2

1. Classify the following statements as propositions or nonpropositions, and explain your answers:
(a) The population of the United States is 185 million.
(b) July 4 occurs in the winter in the northern hemisphere.
(c) Elephants are smarter.
(d) X is greater than Y.

2. Classify the following statements as propositions or nonpropositions, and explain your answers:
(a) Buy bonds!
(b) The DEC Rainbow is an 8-bit computer.
(c) $A + B = 17$.
(d) There is a largest prime number.

3. Construct an example of a proposition which is (a) false, (b) true, and (c) of unknown truth.

4. Construct three statements that are not propositions, and explain why they are not propositions.

Suppose that we have the following:

p: George Washington was the first president of the United States.

q: Abraham Lincoln discovered America.

5. Write the proposition which combines the propositions p and q with the word "or." Is this proposition true or false?

6. Repeat Problem 5 with the word "and."

7. Write the proposition which is the "opposite" of p. Is this proposition true or false?

8. Write the proposition "If p, then q" in good English. Is this statement true or false?

Let p: Neil Armstrong walked on the moon.

q: IBM makes computers.