

NOTICE: THIS MATERIAL MAY BE PROTECTED
BY COPYRIGHT LAW (TITLE 17 U.S. CODE)

DISCRETE MATHEMATICS WITH ALGORITHMS

Michael O. Albertson

Joan P. Hutchinson

Smith College



WILEY

JOHN WILEY & SONS

New York Chichester Brisbane Toronto Singapore

Copyright © 1988, by John Wiley & Sons, Inc.

All rights reserved. Published simultaneously in Canada.

Reproduction or translation of any part of this work beyond that permitted by Section 107 or 108 of the 1976 United States Copyright Act without the permission of the copyright owner is unlawful. Requests for permission or further information should be addressed to the Permissions Department, John Wiley & Sons.

Library of Congress Cataloging in Publication Data:

Albertson, Michael O.

Discrete mathematics with algorithms.

Includes indexes.

1. Mathematics—1961— . 2. Electronic data processing—Mathematics. I. Hutchinson, Joan P.

II. Title.

QA39.2.A43 1988 510 88-235

ISBN 0-471-84902-2

Printed in the United States of America

10 9 8 7 6 5 4 3 2

CONTENTS

Chapter 1	SETS AND ALGORITHMS: AN INTRODUCTION	1
1:1	Introduction	1
1:2	Binary Arithmetic and the Magic Trick Revisited	5
1:3	Algorithms	8
1:4	Between Decimal and Binary	15
1:5	Set Theory and the Magic Trick	20
1:6	Pictures of Sets	25
1:7	Subsets	29
1:8	Set Cardinality and Counting	35
1:9	Functions	40
1:10	Boolean Functions and Boolean Algebra	51
1:11	A Look Back	59
Chapter 2	ARITHMETIC	65
2:1	Introduction	65
2:2	Exponentiation, A First Look	68
2:3	Induction	71
2:4	Three Inductive Proofs	80
2:5	Exponentiation Revisited	88
2:6	How Good Is Fast Exponentiation?	91

CONTENTS

2:7	How Logarithms Grow	97
2:8	The “Big Oh” Notation	102
2:9	$2^n \neq O(p(n))$: Proof by Contradiction	110
2:10	Good and Bad Algorithms	118
2:11	Another Look Back	122
 <i>Chapter 3 ARITHMETIC OF SETS</i>		127
3:1	Introduction	127
3:2	Binomial Coefficients	131
3:3	Subsets of Sets	141
3:4	Permutations	153
3:5	An Application of Permutations: The Game of Mastermind	161
3:6	The Binomial Theorem	168
3:7	Important Subsets	176
 <i>Chapter 4 NUMBER THEORY</i>		181
4:1	Greatest Common Divisors	181
4:2	Another Look at Complexities	186
4:3	The Euclidean Algorithm	190
4:4	Fibonacci Numbers	197
4:5	The Complexity of the Euclidean Algorithm	206
4:6	Congruences and Equivalence Relations	211
4:7	An Application: Public Key Encryption Schemes	222
4:8	The Dividends	234
 <i>Chapter 5 GRAPH THEORY</i>		239
5:1	Building the LAN	239
5:2	Graphs	241
5:3	Trees and the LAN	251
5:4	A Good Minimum-Weight Spanning Tree Algorithm	263
5:5	An Ode to Greed	272
5:6	Graphical Highlights	279

CONTENTS

Chapter 6	<i>SEARCHING AND SORTING</i>	283
6:1	Introduction: Record Keeping	283
6:2	Searching a Sorted File	290
6:3	Sorting a File	295
6:4	Search Trees	302
6:5	Lower Bounds on Sorting	310
6:6	Recursion	317
6:7	MERGESORT	325
6:8	Sorting It All Out	331
Chapter 7	<i>RECURRENCE RELATIONS</i>	339
7:1	Beginnings of Sequences	339
7:2	Iteration and Induction	346
7:3	Linear Homogeneous Recurrence Relations with Constant Coefficients	353
7:4	LHRRWCCs with Multiple Roots: More About Rabbits	364
7:5	Divide-and-Conquer Recurrence Relations	372
7:6	Recurring Thoughts	381
Chapter 8	<i>MORE GRAPH THEORY</i>	389
8:1	Minimum-Distance Trees	389
8:2	Eulerian Cycles	399
8:3	Hamiltonian Cycles	410
8:4	Minimum-Weight Hamiltonian Cycles	425
8:5	Graph Coloring and an Application to Storage Allocation	431
	<i>SOLUTIONS TO QUESTIONS</i>	451
	<i>INDEX</i>	538
	<i>ALGORITHMS AND PROCEDURES</i>	543
	<i>NOTATIONS</i>	545

SETS AND ALGORITHMS: AN INTRODUCTION

1:1 INTRODUCTION

The four cards labeled *A*, *B*, *C*, and *D* in Figure 1.1 are part of a magic trick played by Player 1 upon Player 2. The trick is played as follows:

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>																																
<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 10px;">8</td><td style="padding: 2px 10px;">9</td></tr> <tr><td style="padding: 2px 10px;">10</td><td style="padding: 2px 10px;">11</td></tr> <tr><td style="padding: 2px 10px;">12</td><td style="padding: 2px 10px;">13</td></tr> <tr><td style="padding: 2px 10px;">14</td><td style="padding: 2px 10px;">15</td></tr> </table>	8	9	10	11	12	13	14	15	<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 10px;">4</td><td style="padding: 2px 10px;">5</td></tr> <tr><td style="padding: 2px 10px;">6</td><td style="padding: 2px 10px;">7</td></tr> <tr><td style="padding: 2px 10px;">12</td><td style="padding: 2px 10px;">13</td></tr> <tr><td style="padding: 2px 10px;">14</td><td style="padding: 2px 10px;">15</td></tr> </table>	4	5	6	7	12	13	14	15	<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 10px;">2</td><td style="padding: 2px 10px;">3</td></tr> <tr><td style="padding: 2px 10px;">6</td><td style="padding: 2px 10px;">7</td></tr> <tr><td style="padding: 2px 10px;">10</td><td style="padding: 2px 10px;">11</td></tr> <tr><td style="padding: 2px 10px;">14</td><td style="padding: 2px 10px;">15</td></tr> </table>	2	3	6	7	10	11	14	15	<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">3</td></tr> <tr><td style="padding: 2px 10px;">5</td><td style="padding: 2px 10px;">7</td></tr> <tr><td style="padding: 2px 10px;">9</td><td style="padding: 2px 10px;">11</td></tr> <tr><td style="padding: 2px 10px;">13</td><td style="padding: 2px 10px;">15</td></tr> </table>	1	3	5	7	9	11	13	15
8	9																																		
10	11																																		
12	13																																		
14	15																																		
4	5																																		
6	7																																		
12	13																																		
14	15																																		
2	3																																		
6	7																																		
10	11																																		
14	15																																		
1	3																																		
5	7																																		
9	11																																		
13	15																																		

Figure 1.1

<i>Player 1</i>	<i>Player 2</i>
Pick a whole number between 0 and 15.	
Got it?	Yes.
Is it on card <i>A</i> ?	Yes.
Is it on card <i>B</i> ?	No.
Is it on card <i>C</i> ?	No.
Is it on card <i>D</i> ?	Yes.
The number you picked is 9.	That's amazing! How did you do that? And so fast!

Now let's play again only this time you'll be player 1. I have a whole number between 0 and 15. It appears on cards *A*, *C*, and *D* and does not appear on card *B*. What number am I thinking of?

Question 1.1. (Figure it out before you read any further.) If you are at a loss for what to do, ask yourself the following questions. Can it be 0? Can it be 1? . . . Can it be 15?

Now it can't be 0 because 0 doesn't appear on any of the cards and the number I'm thinking of appears on three cards. It can't be 1 because even though 1 does appear on card *D*, it does not appear on cards *A* or *C*, and the number I'm thinking of appears on both cards *A* and *C*. If this magic trick is well designed, meaning that it is always possible for player 1 to guess player 2's number correctly, then there must be a unique number that corresponds with any possible sequence of answers provided by player 2. In this case the number I am thinking of is 11. It is easy to check that 11 appears on the cards *A*, *C*, and *D* but does not appear on the card *B*. It seems less obvious that 11 is the only such number.

Question 1.2. What would you need to do to check that this trick will always work?

Question 1.3. Design a pair of cards that will serve to distinguish the numbers 0, 1, 2, and 3. Is there more than one way to do this? Why can't two cards distinguish the numbers 0, 1, 2, 3, and 4?

Understanding why two cards can distinguish four numbers and why four cards can distinguish 16 numbers is fundamental to seeing how to design this game as well as how to play it well. Each card that player 1 shows to player 2 elicits one of two responses, either a "yes" or a "no." A game with two cards has four possible responses from player 2. These are "no, no," "no, yes," "yes, no," and "yes, yes." How many responses has a game with four cards? Justice seems to suggest that you respond 16. That is correct. Now let's see why.

Multiplication Principle. Suppose that a counting procedure can be divided into two successive stages. If there are r outcomes for the first stage, and if for each of these outcomes for the first stage, there are s outcomes for the second stage (where r and s are positive integers), then the total number of possible outcomes equals the product of r and s , rs .

Example 1.1. At tea one afternoon you are offered your choice of a bagel, a corn muffin, or a croissant with either cream cheese or lightly salted butter. How many different choices do you have? (Reread the multiplication principle.) At the first stage you can choose whether to have a bagel, muffin, or croissant. There are three

different outcomes ($r = 3$). At the second stage you can choose cheese or butter. There are two different outcomes ($s = 2$). By the multiplication principle as well as by a direct count you have 6 ($=rs$) choices.

Example 1.2. In the magic trick, how many different responses are there to the four cards? (Reread the multiplication principle.) First consider cards A and B . As we've already seen, there are four distinct responses to these two cards ($r = 4$). Next look at cards C and D . It doesn't matter what the responses to the A and B cards were. There are four distinct responses to these two cards ($s = 4$). Thus there are 16 ($=rs$) distinct responses in all to the four cards. Note that these 16 responses could have been counted in four stages with two responses at each of these stages. The multiplication principle works analogously for any number of stages. (See Exercises 7 and 8.)

Question 1.4. How many different seven-digit telephone numbers are there beginning with the digits 584?

Now returning to the magic trick, you see that player 1 could perform the trick by memorizing the 16 different responses that player 2 might give in order to successfully "guess" player 2's number. The possible responses are listed in Table 1.1.

Table 1.1

<i>Player 2's Number</i>	<i>Responses</i>			
	<i>Card A</i>	<i>Card B</i>	<i>Card C</i>	<i>Card D</i>
0	no	no	no	no
1	no	no	no	yes
2	no	no	yes	no
3	no	no	yes	yes
4	no	yes	no	no
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				

Question 1.5. Complete Table 1.1.

EXERCISES FOR SECTION 1

1. Design a set of three cards that will distinguish the numbers $0, 1, 2, \dots, 7$. Suppose that we only wished to distinguish the numbers $0, 1, 2, \dots, 5$. Could your three cards be modified to play this game? Could your three cards be modified to play the game with the numbers $0, 1, 2, \dots, 8$?
2. Suppose that the local ice cream store offers 12 different flavors of ice cream and 5 different types of topping (chocolate, butterscotch, strawberry, blueberry, and raspberry). How many different dishes of ice cream plus topping are possible? Suppose that you can turn these dishes of ice cream plus topping into special sundaes by adding one kind of nuts (walnuts, almonds, or hazelnuts) and whipped cream if you like. How many different types of special sundaes can you order at this ice cream store?
3. A certain fast food chain offers a one-price meal consisting of a burger, an order of potatoes, a salad, a dessert, and a beverage. There are seven different kinds of burgers, three different kinds of potatoes, five different kinds of salads, and four different kinds of desserts. The restaurant advertises that you can eat one meal here every day for four years without ever having the same meal twice. What can you say about the number of beverage choices that the restaurant offers?
4. Often, when you sign onto a time-sharing computer, you are asked to specify the room you are in and the kind of terminal that you are using. Suppose that there are 13 different room categories and 16 different kinds of terminals. How many different pairs of answers is it possible to give as you sign on?
5. In the context of the preceding problem it is typically the case that not all answers are possible, since there are not 16 different kinds of terminals in every room. If every room contains four different kinds of terminals, how many different answers are possible?
6. Even the idea in the last problem might not be correct, since the kind and number of terminal types may vary from place to place. Suppose that we consider only five rooms and that they contain the following kinds of terminals: Every room contains a Digital VT terminal; Tektronix machines are located in the Social Science Room and in the Science Lab; IBM PCs are found in the Graphics Lab and in the Library Terminal Room; and Apple Macintoshes are available in the Library Terminal room and in the Humanities Computer Room. How many pairs of responses are now possible to send to the computer when you sign on?
7. Here is an extension of the multiplication principle: Suppose that a counting procedure can be divided into four successive stages. If there are p outcomes for the first stage, if for each of these outcomes for the first stage, there are r outcomes for the second stage, if for each pair of these first two outcomes,

there are s outcomes for the third stage, and finally if for each of these first three outcomes, there are t outcomes for the fourth stage (where p , r , s , and t are positive integers), then the total number of possible outcomes equals the product $prst$. Explain why this is valid, using the original form (two-stage) of the multiplication principle.

8. State and explain a multiplication principle that is valid for three stages, and then do the same for five stages.
9. Suppose that we have a rather primitive computer that can receive only strings of zeros and ones as input. Furthermore, these strings must contain exactly eight digits. How many different input strings are there?
10. Suppose that the machine in the preceding problem can receive strings with one to eight digits, and suppose that the machine disregards initial zeros. Thus, for instance, 1001 is the same input as the string of eight digits, 00001001. Now how many different input strings are possible?
11. How many different seven-digit phone numbers are there that begin 584 and contain no zero? How many phone numbers are there that begin 584 and contain at least one zero?
12. How many different seven-digit phone numbers are there that begin 58_-____ and contain seven different digits? How many of these contain no zero? How many do contain a zero? How many different phone numbers are there that begin with 58_, but contain no two identical consecutive digits?
13. Recently, a new telephone area code was introduced for the area of New York that contains Brooklyn and Queens because all seven-digit phone numbers had been used up. Assuming that none of the first three digits in a phone number can be either a 0 or a 1, what can you say about the number of phone lines in this area?
14. In the lottery game called Megabucks a player selects six different numbers between 0 and 35. How many different such selections are there? Before answering, specify when two selections are the same and when they are different.

1:2 BINARY ARITHMETIC AND THE MAGIC TRICK REVISITED

The magic trick of Section 1 was based on each of four questions receiving either a “yes” or a “no” answer. Thus a seemingly complex task, in this case deciding which number player 2 had chosen, could be broken down into a sequence of smaller tasks associated with each of the cards. This fundamental yes-no, true-false, or on-off dichotomy pervades most of the mathematics associated with computers. It even is fundamental to how computers “think” about numbers. We now model how a computer stores an integer using binary numbers.

1 SETS AND ALGORITHMS: AN INTRODUCTION

A number in **binary notation** is just a finite list (sequence or string) of zeros and ones. For example, 1, 101, 111001, and 1011 are all binary numbers. For the moment don't be concerned about which numbers these sequences are. We'll get to that shortly. Rather, think in the familiar decimal system. The number 37, for instance, can be thought of as 3 tens together with 7 ones,

$$37 = 3 \cdot 10 + 7.$$

The number 234 is 2 hundreds plus 3 tens plus 4 ones,

$$234 = 2 \cdot 100 + 3 \cdot 10 + 4.$$

The decimal number system is so named because successive columns when reading from the right represent consecutive powers of ten. We'll use the same device for the binary system. Specifically, let the rightmost column of a number written in binary represent the ones. We call this the 0th column or the one's column. The next column to the left represents the twos. This is called the 1st column or the two's column. The third column from the right represents the fours, the fourth column from the right represents the eights, and so on. So the p th column from the right (starting with $p = 0$) in the binary representation of a number represents the p th power of 2.

Example 2.1. The binary number 1 is the same number as the decimal number 1. The binary number 101 consists of 1 one, no twos, and 1 four. It is equivalent to the (decimal) number 5. Similarly, the binary number 111001 is equivalent to the decimal number 57, since $57 = 1$ one + 0 twos + 0 fours + 1 eight + 1 sixteen + 1 thirty-two (see Figure 1.2).

BINARY NUMBER		←→		DECIMAL NUMBER
101	=	$1 \cdot 4 + 0 \cdot 2 + 1$	=	5
111001	=	$1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 1 \cdot 1$	=	57

Figure 1.2

Question 2.1. Find the decimal equivalents of the following binary numbers: (a) 10101, (b) 100101, and (c) 11010. Given a binary number, how would you decide whether it is an even number or an odd number?

Question 2.2. Construct a table with all the numbers from 0 to 15 written in binary. Compare your results with the table that you completed for Question 1.5.

Question 2.3. List all the numbers from 0 to 15 that have a 1 in the four's column of their binary representation.

After doing the previous question, you should note that the numbers you obtain appear familiar. In fact they are just the numbers that appear on card *B* in the magic trick. You should check that the numbers that appear on card *A* are just those numbers between 0 and 15 whose binary representation has a 1 in the eight's column, those on card *C* are just those with a 1 in the two's column, and those on card *D* are just those with a 1 in the one's column. This suggests a quick way for player 1 to perform the magic trick. Player 1 should remember the number in the upper left-hand corner of every card to which player 2 says "yes" and add these numbers up. Thus in our original play the yes to card *A* produces an 8, and the yes to card *D* produces a 1 for a total of 9.

At the moment we have used binary representations of numbers to produce a simple procedure for player 1 to perform the magic trick. It is easy to proceed from the binary representation of a number to its decimal equivalent. What about the other direction, that is, given a number in decimal form how should we arrive at its binary equivalent?

Question 2.4. Write the following (decimal) numbers in binary: (a) 6, (b) 19, (c) 52, (d) 84, and (e) 232.

You have already done part (a) in previous questions. Part (b) you probably see. To write 52 and 84 in binary, you might require pencil and paper. By the time you get to 232, you will be glad the question stops. Surely you realize that given any positive integer of moderate size, you could find its binary representation with enough time, motivation, and paper. Still if the method that you have used is, basically trial and error, you might wish for an alternative. What you need in the jargon of discrete mathematics is a "good algorithm." We shall introduce you to this language in the next section.

EXERCISES FOR SECTION 2

1. List all numbers from 0 to 15 that have a 1 in the eight's column. Then list all of these numbers that have a 1 in the two's column. Why is it precisely the odd numbers that have a 1 in the zero's column?
2. Without listing all the numbers, give a characterization of the numbers from 0 to 31 that have a 1 in the sixteen's column. Then characterize those numbers from 0 to 31 that have a 1 in the eight's column. Finally, describe all numbers from 0 to 31 whose binary expansion ends with the digits 01.
3. What decimal numbers do the following binary numbers represent? (**a**) 11011, (**b**) 101011, (**c**) 10001, and (**d**) 11000.

4. Find the binary representation of the following decimal numbers: (a) 28, (b) 43, (c) 100, and (d) 81.
5. Suppose that you are given a decimal number m that has the property that when m is divided by 8 there is a remainder of 2. What can you say about the binary representation of m ?
6. Suppose that you are given a decimal number m that has the property that when m is divided by 4 there is a remainder of 3. What can you say about the binary representation of m ?
7. Suppose that you are given a decimal number m that has the property that when m is divided by 16 there is a remainder of 6. What can you say about the binary representation of m ?
8. What is the maximum number of integers that a six-card magic trick could distinguish? If the six cards were designed as in the original trick from Section 1 and the responses were yes, no, yes, yes, no, no; what number was selected?
9. Given two binary numbers, how could you tell (without converting them into decimal) which is bigger?
10. A **binary fraction** is a finite sequence of zeros and ones that follows what is called the binary point. For example 0.101 is a binary fraction. The column immediately to the right of the binary point represents the halves, the next column the quarters, the third column the eighths, and so on. Thus $0.101 = \frac{1}{2} + \frac{1}{8} = \frac{5}{8}$. Make a table of all four-bit binary fractions.
11. Express each of the following in binary: (a) $\frac{27}{32}$, (b) $\frac{36}{64}$, (c) $\frac{31}{8}$, (d) $\frac{53}{4}$, and (e) $\frac{127}{16}$.

1:3 ALGORITHMS

Definition. An **algorithm** is a finite sequence of well-described instructions with the following properties.

1. There is no ambiguity in any instruction.
2. After performing a particular instruction, there is no ambiguity about which instruction is to be performed next.
3. The instruction to stop is always reached after the execution of a finite number of instructions.

Example 3.1. Here are the instructions inscribed on a metal plate attached to the front of a video game:

STEP 1. Insert quarter into slot on side of machine.

STEP 2. Press green button on top of machine when ready to begin.

This is a finite sequence of well-described instructions. There is no ambiguity in any instruction. It is always clear what to do next. There is no explicit instruction to stop, but we overlook this, since it is clear that each instruction is executed just once for each play. Thus we may call this an algorithm: It is an algorithm to begin playing a video game.

Example 3.2. Suppose that we add an instruction to the previous example.

STEP 3. When each game is over, type your initials and press red button on top of machine to record score for posterity.

Step 3 fulfills the role of a stop instruction provided games don't go on forever and provided you cannot play an additional game on your initial quarter.

Example 3.3. Suppose we insert the following instruction.

STEP 4. For each 10,000 points you accumulate you will win a free game. When current game is over, if you have won a free game, go to step 2.

This has changed the nature of the instructions. The first two examples are known as sequential algorithms. By that we mean that each step (or instruction) is executed exactly once and that the next step on the list is the next instruction to be executed. Step 4 adds the possibility of executing an instruction many times. Indeed since there is no reason to suppose that the winning of free games couldn't go on forever the list of instructions in Example 3.3 is not an algorithm. You might think of this by reinterpreting instruction 4 as saying,

"If score \geq 10,000, then go to instruction 2."

This logical structure is known as a loop. Although this particular loop has made the sequence of instructions fail to be an algorithm, a loop need not force the execution of an infinite number of steps.

Example 3.4. Suppose that we modify instruction 4 so that our instructions now say:

STEP 1. Insert quarter into slot on side of machine.

STEP 2. Press green button on top of machine when ready to begin.

STEP 3. When each game is over, type your initials and press red button on top of machine to record score for posterity.

STEP 4. For each 10,000 points you accumulate, you will win a free game up to a maximum of 10 free games for each paid game. When current play is over, if you have won a free game, go to step 2.

1 SETS AND ALGORITHMS: AN INTRODUCTION

The modification in step 4 can be described as adding a counter to the loop in order to insure that the loop is executed a finite number of times.

The logical structure of Example 3.3 illustrates one of the most common mistakes made by beginning programmers, that of an infinite loop. Example 3.4 shows a typical quick fix.

Reread the definition of algorithm! Given a sequence of instructions, how could it fail to be an algorithm? First, it might be the case that at least one of the instructions is ambiguous. In other words, some instruction might be poorly specified so that you, the reader, do not know how to carry out the instruction or so that a computer programmer does not know how to translate the instruction into a suitable computer language. An instruction that is clear to one person may be full of ambiguities for another. For example, an instruction like, “Start the airplane engine and take off on runway 4,” might be unambiguous to trained pilots.

A sequence of instructions might fail to be an algorithm because after executing a particular instruction, it might not be clear which instruction is to be executed next. Some instructions will clearly indicate the next instruction, such as “. . . and then go to step—.” If no such direction is given, we always move to the next step as given in the sequence of instructions. Finally, we would not have an algorithm if the execution of the sequence of instructions did not terminate in all instances.

If a sequence of instructions does satisfy the definition of an algorithm, it still might not be a correct algorithm to perform the desired task. To illustrate this possibility, we introduce an example from the kitchen. (It seems that most expostions about algorithms revert to cooking recipes at some time.)

Example 3.5. Consider the following sequence of instructions.

- STEP 1. Place one cup of water in the top of a double boiler.
- STEP 2. Place one cup of quick oatmeal in the bottom of a double boiler.
- STEP 3. Turn on a stove burner to medium.
- STEP 4. Place double boiler on burner and heat for 10 minutes.
- STEP 5. Remove pot.
- STEP 6. Turn off burner.

You may verify that the instructions satisfy the definition of algorithm. However, you would be unlikely to enjoy eating the results of this recipe. This example illustrates that a particular algorithm is designed in response to a particular problem. In this case the problem (which while not explicitly stated) can reasonably be inferred to be to make oatmeal. This sequence of instructions is not a correct algorithm for making oatmeal.

Question 3.1. Rewrite the steps in Example 3.5 so that the resulting algorithm correctly instructs us to make oatmeal.

Notice that writing a correct algorithm may be significantly more difficult than checking whether a given sequence of instructions is a valid algorithm. Creating an algorithm requires expertise with the subject matter. To give a correct algorithm for making quick oatmeal, you need to know (or read on the box) the proper proportions of water and oatmeal, the cooking time, and so on. To play a video game requires knowledge of the rules and object of the game. For the more mathematical algorithms of this book we need to develop the language and techniques of the subject before we can use, let alone create, new algorithms.

Sometimes it may be difficult to decide if a given sequence of instructions will necessarily terminate in a finite number of steps. The obvious cases might always stop, but how can we know if we have tested all possibilities?

Example 3.6. Consider the following algorithm.

Algorithm(?) COLLATZ

- STEP 1. Input z a positive integer
- STEP 2. If z is even, replace z by $z/2$
- STEP 3. If $z = 1$, then output z and stop.
- STEP 4. If z is odd, replace z by $3z + 1$
- STEP 5. Go to step 2

This will be an algorithm if it stops. This will happen if z eventually equals 1. Whether or not this will happen for every positive integer z is a famous unsolved problem known as the Collatz Problem.

Question 3.2. Run COLLATZ for the following initial values of z : (a) $z = 1$, (b) $z = 20$, and (c) $z = 7$.

An algorithm will typically need input to begin and will produce output at the end. Input is the data or material needed to start the algorithm, like a quarter, water and oatmeal, or a positive integer z . The output is the result of the algorithm, derived from the particular input, like a score stored on a video game, burned oatmeal, or the number $z = 1$. General-purpose algorithms, which are the most useful, will draw input values from a set of possibilities, like all positive integers or all quick-cooking hot cereals. For each input there will be exactly one set of output. (The word *set* will be explained more fully in Section 5.)

Given a finite sequence of instructions that is in fact an algorithm, we now address the question of what qualities make this algorithm “good.” Algorithms

1 SETS AND ALGORITHMS: AN INTRODUCTION

are created to solve problems. The term *correct* is used to label algorithms that produce correct solutions to a particular problem. Thus the sine qua non of a good algorithm is that the output must be correct for all possible input data. A great deal of effort within computer science is expended proving that programs (implementations of algorithms) are correct.

What properties might distinguish two correct algorithmic solutions to a particular problem? One might be easier to understand. One might provide internal consistency checks to assure that the algorithm was being carried out correctly. One might be easier to implement in your favorite programming language. Most commonly, one algorithm is said to be better than another if it requires fewer resources to implement. These resources may consist either of time (the number of steps required until the algorithm terminates) or space (the amount of memory required to implement the algorithm). In the next section we illustrate these notions with several different responses to the problem of taking a number written in decimal and translating it into binary.

EXERCISES FOR SECTION 3

1. Here are two approaches to making whipped cream. Discuss whether these are or are not algorithms.

Approach 1

- STEP 1. Buy a pint of whipping cream.
- STEP 2. Chill cream and beaters until cold.
- STEP 3. Add a small amount of sugar and vanilla extract to cream.
- STEP 4. Whip cream until stiff but not dry.
- STEP 5. Wash dishes and stop.

Approach 2

- STEP 1. Buy a can of Readi-Whip.
- STEP 2. Shake 30 times.
- STEP 3. Invert can.
- STEP 4. Press nozzle.
- STEP 5. Stop.

2. Here are two algorithms that calculate the sum of the integers from 1 to 100. Comment on the relative efficiency of the two responses.

Response 1

- STEP 1. Sum the integers from 1 to 100 and store the result in the variable Answer.
- STEP 2. Print out the value of Answer.
- STEP 3. Stop.

Response 2

- STEP 1. Store each of the integers from 1 to 200 in different memory locations.
- STEP 2. Sum the integers from 1 to 200 and store the result in the variable Answer.
- STEP 3. Sum the integers from 101 to 200, subtract the sum from the number in Answer, and retain the difference in Answer.
- STEP 4. Print out the values of all 200 integers and also the value of Answer.
- STEP 5. Stop.
3. Each of the following fails to be an algorithm; what rule or rules do they violate?

Attempt 1

- STEP 1. Set Sum equal to 1
- STEP 2. Set X equal to 2
- STEP 3. Give Sum the value $\text{Sum} + X$
- STEP 4. If Sum is even, then go to Step 5;
if Sum is odd, go to Step 3
- STEP 5. Print out the value of Sum and stop.

Attempt 2

- STEP 1. Set Sum equal to 1
- STEP 2. Set X equal to 2
- STEP 3. Give Sum the value $\text{Sum} + X$
- STEP 4. If Sum is even, then stop.
4. Give a finite sequence of well-described instructions that never stops; that is, give an example of a pseudo-algorithm that fails to meet the third property needed to be an algorithm.

1 SETS AND ALGORITHMS: AN INTRODUCTION

5. In each of the following decide whether or not the sequence of instructions is an algorithm. If not, explain why not. If yes, figure out what the algorithm produces as an answer.

Algorithm A

- STEP 1. Set $j = 0$
- STEP 2. Set Answer = 0
- STEP 3. Give Answer the value Answer + 2^j
- STEP 4. Add 1 to j
- STEP 5. If j is less than 5, go to step 3
- STEP 6. Output Answer
- STEP 7. Stop.

Algorithm B

- STEP 1. Set $j = 0$
- STEP 2. Set Answer = 0
- STEP 3. Give Answer the value Answer + 2^j
- STEP 4. Add 1 to j
- STEP 5. If j is not zero, go to step 3
- STEP 6. Output Answer
- STEP 7. Stop.

Algorithm C

- STEP 1. Set $j = 0$
- STEP 2. Set Answer = 0
- STEP 3. Give Answer the value (Answer) (2^j)
- STEP 4. Add 1 to j
- STEP 5. If j is less than 5, go to step 3
- STEP 6. Output Answer
- STEP 7. Stop.

6. Find a more efficient algorithm that produces the same output as the one given here.

Algorithm FUN

- STEP 1. Input z , an integer between 100 and 300
 STEP 2. Let $u = 3z$
 STEP 3. Let w be u written twice {so 345 becomes 345,345}
 STEP 4. Let y be w with an extra zero on the end
 STEP 5. Let a be y divided by 2
 STEP 6. Let b be a divided by 3
 STEP 7. Let c be b divided by 5
 STEP 8. Let d be c divided by 7
 STEP 9. Let e be d divided by 11
 STEP 10. Let f be e divided by 13
 STEP 11. Let g be f plus 1
 STEP 12. Output g and stop.

1:4 BETWEEN DECIMAL AND BINARY

In Section 2 we discussed the problem of changing numbers from decimal to binary, and vice versa. Example 2.1 outlines a procedure for changing a binary number into its decimal equivalent. We shall soon make this procedure precise. First we digress to introduce some convenient notation that we shall use to present various algorithms.

The symbol “:=” is used for assignment. Specifically, the statement $a = b$ means that the current value of the variable b is assigned to the variable a . We can use this to form instructions that do not represent equalities in the normal arithmetic sense. For instance, the statement

$$a := a + 1$$

does not mean that $a = a + 1$, an assertion that is never true. Rather it means that the value assigned to the variable a should have 1 added to it, and the resulting sum should be reassigned to the variable a .

The symbol “*” is used for multiplication. Specifically, the expression $a * b$ means the product of the numbers a and b . The symbol “/” is used for division. Specifically, a/b means that a is divided by b . Thus the statement

$$a := a/b + 2^3$$

will divide the value of a by the value of b , add 8 to this quotient, and store the result in the variable a . For this instruction to make sense, the variables a and b must have been previously assigned values.

Problem. Given a binary number s as a string of zeros and ones, convert this into its decimal equivalent.

(To add clarity to our algorithmic instructions, we shall often insert comments inside braces as in {COMMENT: ...}).

Algorithm BtoD

- STEP 1. Set $j := 0$
{ j will stand for the binary column with which we are currently working. We label the columns from right to left beginning with 0. }
- STEP 2. Set $m := 0$
{ m will contain the final decimal number. }
- STEP 3. If there is no j th entry of s , then stop.
- STEP 4. If the j th entry of s is a 1, add 2^j to m
{ We write this as $m := m + 2^j$. }
- STEP 5. Increase the value of j by 1, that is, set $j := j + 1$
- STEP 6. Go to step 3

Table 1.2

Step	Value Assigned to the Given Variable		
	s	j	m
1	1011	0	
2	1011	0	0
3	1011	0	0
4	1011	0	1
5	1011	1	1
6	1011	1	1
4	1011	1	3
5	1011	2	3
4	1011	2	3
5	1011	3	3
4	1011	3	11
5	1011	4	11
3	STOP		

Example 4.1. Table 1.2 is a detailed look at what happens when this algorithm is applied to the binary number 1011. Such a tabulation is called a **trace** of the algorithm. We record the values of the variables at the end of each step.

Question 4.1. Apply BtoD to the binary numbers 10101, 11010, and 100101. Do you get the same answers and are you carrying out the same procedure as in Question 2.1?

Next we return to the harder Question 2.4: Given a number written in decimal, how should we find its binary equivalent? Several sequences of instructions listed in order of increasing quality follow. First the problem is formalized.

Problem. Find the binary representation of a positive integer m .

Response 1

- STEP 1. Write down a finite sequence of zeros and ones.
- STEP 2. Take the binary number you wrote down in step 1 and translate it into decimal using Algorithm BtoD, given above.
- STEP 3. If the number you obtain in step 2 equals m , then stop. Otherwise go to step 1.

Question 4.2. Why is Response 1 not an algorithm?

Response 2

- STEP 1. Set $k := 1$
{ k will denote the number of binary digits in the number under consideration.}
- STEP 2. List all possible sequences of zeros and ones with k digits in increasing order
{For example, if $k = 1$, $0 < 1$, and if $k = 2$, $00 < 01 < 10 < 11$.}
- STEP 3. For each sequence from step 2, find its decimal equivalent using Algorithm BtoD
- STEP 4. If one of the numbers that you obtain in step 3 equals m , then stop.
- STEP 5. $k := k + 1$
- STEP 6. Go to step 2

The binary digits, zeros and ones, in a binary number are called **bits**.

Question 4.3. Why is the sequence of steps listed in Response 2 a correct algorithm for solving the problem? Use this algorithm to find the binary representation of 19. What makes this algorithm low quality?

Response 3

- STEP 1. Find the largest power of 2 that is less than or equal to m . If this is the r th power of 2, place a 1 in the r th column (reading from the right and beginning with 0)
- STEP 2. Subtract the power of 2 obtained in step 1 from m and set the result equal to m , or in symbols, set $m := m - 2^r$. If m equals zero, fill in the remaining columns with zeros and stop.
- STEP 3. Go to step 1

Question 4.4. Why does the sequence of steps listed in Response 3 necessarily stop? Use this algorithm to find the binary representation of 182.

Response 4 (ALGORITHM DtoB)

- STEP 1. Set $j := 0$
 $\{j \text{ will indicate the column in the binary representation of } m \text{ on which we are working.}\}$
- STEP 2. Divide m by 2 to obtain the quotient q and the remainder r $\{ \text{necessarily either 0 or 1} \}$; place r in the j th column of the answer $\{ \text{reading from the right} \}$
- STEP 3. If $q = 0$, then stop.
- STEP 4. Set $m := q$
- STEP 5. Set $j := j + 1$
- STEP 6. Go to step 2

Example 4.2. Table 1.3 is a trace of the algorithm DtoB, run on the decimal number 21.

Question 4.5. Why does the sequence of steps listed in Response 4 necessarily stop? Use this algorithm to find the binary representation of 395. Compare the algorithms in Responses 3 and 4.

The algorithm given in Response 4 is one that we shall use again and so we have named it Algorithm DtoB. In the exercises you are asked to work with the algorithms and ideas of this section. Here and throughout the book you will be asked to write algorithms. How should you create an algorithm from scratch? Here are some ideas, but there is no all-purpose algorithm to create an algorithm! First

Table 1.3 Values Assigned to the Variable After the Execution of the Given Step.

<i>Step</i>	<i>j</i>	<i>m</i>	<i>q</i>	<i>r</i>	<i>Answer</i>
1	0	21			
2	0	21	10	1	1
4	0	10	10	1	
5	1	10	10	1	
2	1	10	5	0	01
4	1	5	5	0	
5	2	5	5	0	
2	2	5	2	1	101
4	2	2	2	1	
5	3	2	2	1	
2	3	2	1	0	0101
4	3	1	1	0	
5	4	1	1	0	
2	4	1	0	1	10101
3	STOP				

figure out how to solve the problem at hand. Then ask yourself what your steps were and try to write them down so that another person or a computer could understand and follow them. Then analyze these steps, as we have in this section, to see whether your steps are a correct algorithm. This task is always challenging. Sometimes you will have seen algorithms in the text and exercises that you can modify and build upon; other times you need to jump in and follow your own logical path to a solution.

EXERCISES FOR SECTION 4

1. Apply BtoD to the following binary numbers: **(a)** 11, **(b)** 101, **(c)** 1101, **(d)** 1011, **(e)** 1111, and **(f)** 10101010.
2. Suppose that the decimal number D is expressed in binary as a sequence S of zeros and ones. If a zero is placed at the right end of S , how does the decimal value of the resulting number compare with D ? If a one were placed at the right end of S , how would the decimal value change?
3. Suppose that S is a string of zeros and ones that corresponds with the even decimal number D . If the last entry of S on the right is erased, express the value of the new decimal number in terms of D . Repeat if D is odd.
4. Apply DtoB to the decimal numbers 17 and 59. Then apply BtoD to the resulting binary numbers. Next apply BtoD to the binary numbers 10001 and 110110. Then apply DtoB to the resulting decimal numbers.

5. A 16-bit computer allocates 16 spaces or bits to store an integer. The first bit designates whether the number is positive or negative and the remaining bits contain either a zero or a one, expressing the integer in binary. How many different integers can you store in 16 bits? What is the decimal value of the largest and of the smallest integer that can be stored using 16 bits?
6. Numbers written in base 3 can use only the digits 0, 1, and 2. Thus the decimal numbers 0, 1, and 2 are expressed in base 3 in the same way, but to write 3 in base 3 we must write $10 = 1 \cdot 3^1 + 0 \cdot 3^0$. Write the (decimal) numbers from 4 to 12 in base 3. The base 3 representation of a number is also called its **ternary representation**.
7. For each of the following numbers written in base 3, determine its decimal equivalent: **(a)** 22, **(b)** 20102, **(c)** 12121, **(d)** 20010, and **(e)** 1121.
8. Translate each of the following decimal numbers into a base 3 representation: **(a)** 13, **(b)** 15, **(c)** 21, **(d)** 27, **(e)** 30, and **(f)** 80.
9. Find the base 3 representation of the number 20 and then translate that base 3 number back into decimal notation. Similarly, begin with the base 3 number 111, find the decimal number that it represents, and change that decimal number back into base 3.
10. Given a number s expressed in base 3 as a string of 0s, 1s and 2s, write down an algorithm that will convert s into its decimal equivalent. (*Hint*: Look at Algorithm BtoD.)
11. How can you tell if a given number written in base 3 is even?
12. Given a decimal number n , write down an algorithm that will express n in base 3 as a string of 0s, 1s, and 2s.
13. What can you say about the base 3 representation of a decimal number that when divided by 9 leaves a remainder of 7?
14. Create an algorithm that will input a binary fraction (see Exercise 2.10) and output the fraction in standard form. Run your algorithm with input **(a)** 0.1101, **(b)** 0.00101, and **(c)** 0.101010.
15. Create an algorithm that will input a positive integer n and a rational number z with $0 < z < 1$ and output n bits of z 's binary representation. Run your algorithm for $n = 6$ and **(a)** $z = \frac{5}{16}$, **(b)** $z = \frac{7}{32}$, **(c)** $z = \frac{1}{3}$, and **(d)** $z = \frac{1}{10}$.
16. Why can't there be an algorithm to input a fraction z with $0 < z < 1$ and output z 's binary representation?

1:5 SET THEORY AND THE MAGIC TRICK

When we think about the magic trick presented in Section 1, the numbers $0, 1, 2, \dots, 15$ are all of the objects with which we are concerned. The totality of these objects we call our **universe** or underlying set. The eight numbers that appear

on card A are called a set of objects. More generally, given any universe of objects we use the word **set** to denote any well-specified collection of objects from the universe. We have included the descriptive “well-specified” in order to make it clear that there can be no ambiguity as to what is in and what is not in the set.

Given two sets, say A and B , of objects from the same universe, A is said to be a **subset** of B , denoted by $A \subseteq B$, provided every object that is contained in A is also contained in B . Every set is necessarily a subset of the universal set, and the **empty set** or null set, the set with no objects, is a subset of every set. The empty set is often denoted by \emptyset . The objects in a set are also called **elements** of the set. Two sets A and B are said to be **equal**, written $A = B$, if they contain precisely the same objects (or elements). Note that $A = B$ if and only if $A \subseteq B$ and $B \subseteq A$.

Example 5.1. Let Z denote the integers. We can list the objects in Z by writing $0, 1, -1, 2, -2, \dots$. For this example Z will be the underlying set or universe.

Let N denote the natural numbers, that is, N can be listed as $0, 1, 2, 3, \dots$. The natural numbers are objects in Z and there is no ambiguity about which objects are in N , so the natural numbers are a set in the universe of the integers.

Let P denote the set of prime numbers. An integer greater than one is called **prime** if it cannot be factored into the product of two smaller positive integers. Thus 5 is prime while $6 = 2 \cdot 3$ is not prime. P can be listed as $2, 3, 5, 7, 11, \dots$. P is a subset of N . Note that for a large integer it might be computationally difficult to decide whether the integer is prime or not. Nevertheless, the elements in P are well specified.

Example 5.2. Let the universe consist of the students enrolled in mathematics courses this semester. If S denotes the set of students who are enrolled in Discrete Mathematics and A denotes the set of students who will earn As in Discrete Mathematics, then A is (we hope) a nonempty subset of S .

Typically, we shall use capital letters to denote sets and small letters to denote the objects, when this is possible. If A is a set and s an element of A , we write $s \in A$; read “ s is an element of A .” If s is not an element of A , we write $s \notin A$. When specifying a set, we shall occasionally list all objects within a pair of curly braces; however, the order in which the objects are so listed is immaterial. The set $\{1, 2, 3\}$ is the same set as $\{3, 1, 2\}$ and $\{1, 3, 2\}$, since each contains precisely the same objects. More frequently, we shall place within the curly braces the property or properties that specify the set.

Example 5.3. The numbers that appear on card A in the magic trick form a set that we could denote by $A = \{8, 9, 10, 11, 12, 13, 14, 15\}$ or by

$$A = \{x: 0 \leq x \leq 15 \text{ and the binary form of } x \\ \text{contains a 1 in the third column, reading from} \\ \text{the right and beginning with 0}\}.$$

Read the above line as “ A equals the set of x such that zero is less than or equal to x , which is less than or equal to 15, and the binary form of x contains a 1 in the third column” It is often the case that some contextual information is left out of the specification of the set. Here, we did not note that the objects in the universe are all integers. Such information may be omitted provided that it does not lead to any confusion on your part.

Question 5.1. Let the universe consist of the positive integers less than 30. Below we list several sets that are well specified by the properties that their elements must satisfy. For each such set list the elements in the set.

(a) $A = \{x: x \text{ is not prime}\}$.

(b) $B = \{x: x \text{ is a square, that is, for some integer } y, x = y^2\}$.

(c) $C = \{x: x \text{ is divisible by a square greater than one}\}$.

Given a set A consisting of some objects from the universe U , the **complement** of A , denoted by A^c , is the set of objects from the universe that are not elements of A . In the curly brace notation

$$\begin{aligned} A^c &= \{x \text{ is in } U: x \text{ is not in } A\} \\ &= \{x \in U: x \notin A\}. \end{aligned}$$

Question 5.2. Find the complements of each of the sets from the preceding question.

Question 5.3. For the six sets you found in the preceding two questions determine which are subsets of each other.

From two sets of elements in the same universe, say A and B , we can derive two new sets, the union of A and B and the intersection of A and B . The **union** of A and B , denoted by $A \cup B$, consists of all the elements of U that are either in A or in B (or in both). Note that in English “or” often means exclusive or: For lunch I shall eat a pizza or a grinder (but not both). Here we use “or” in the inclusive sense: mathematics majors usually study statistics or computer science (or both). The **intersection** of A and B , denoted by $A \cap B$, consists of all the elements of U that are in both A and B . In curly brace notation

$$A \cup B = \{x: x \text{ is in } A \text{ or } x \text{ is in } B \text{ or } x \text{ is in both } A \text{ and } B\}$$

and

$$\begin{aligned} A \cap B &= \{x: x \text{ is in } A \text{ and } x \text{ is in } B\} \\ &= \{x: x \in A \text{ and } x \in B\}. \end{aligned}$$

Question 5.4. Find the pairwise unions and intersections of the sets you found in Question 5.1.

Example 5.4. Let A , B , C , and D denote the sets of numbers on the cards of the magic trick. For instance, suppose that player 2 is thinking of the number 6. This number appears on cards B and C and does not appear on cards A and D . When player 2 says yes to card B and yes to card C , player 1 knows that player 2's number is in the set labeled B and in the set labeled C . In the language just introduced player 2's number is in $B \cap C$. Now

$$B \cap C = \{4, 5, 6, 7, 12, 13, 14, 15\} \cap \{2, 3, 6, 7, 10, 11, 14, 15\} = \{6, 7, 14, 15\}.$$

Let us do the analogous set theory for the no responses. When player 2 says no to card A , player 1 knows that the number is not on card A . If a number is not in A but is in the universe, then it must be in the complement of A . Similarly, player 2's number must be in the complement of D . In the language of set theory, $A^c = \{0, 1, 2, 3, 4, 5, 6, 7\}$, $D^c = \{0, 2, 4, 6, 8, 10, 12, 14\}$, so

$$A^c \cap D^c = \{0, 2, 4, 6\}.$$

Player 2's responses mean that the number chosen must be in A^c , B , C , and D^c . What number is it?

$$A^c \cap B \cap C \cap D^c = (B \cap C) \cap (A^c \cap D^c) = \{6, 7, 14, 15\} \cap \{0, 2, 4, 6\} = \{6\}.$$

In general, one way to explain why the magic trick always works is to notice that if the set S is either A or A^c , the set T either B or B^c , V either C or C^c , and W either D or D^c , then $S \cap T \cap V \cap W$ contains exactly one number.

EXERCISES FOR SECTION 5

- Let the universe consist of all two-letter "words," that is, all sequences of two alphabetic characters (which don't have to form a real English word). Let A consist of all of these words that begin with an "a," and B those that end with a "b." Let C be those that contain no "c," D those that contain no vowel, and E those that contain only vowels.
 - Describe the complementary set in each case.
 - List or describe $A \cap B$, $A \cup B$, $A \cap C$, $A \cup C$, $A \cap D$, $A \cap E$, $B \cap D$, $B \cup D$, $C \cap D$, $C \cup E$, and $D \cup E$.
- Let the universe consist of all two-digit numbers:

$$\{x: 10 \leq x \leq 99\}.$$

1 SETS AND ALGORITHMS: AN INTRODUCTION

Let A be the set of two-digit numbers that begin with a 1, B those that end with a 9, C those that are multiples of 3, D those in which both digits are even, and E those that are even.

- (a) Among these sets, find two such that one is contained in the other.
 - (b) Are there three sets whose intersection is empty, but such that the intersection of any two is not empty?
 - (c) Construct a set with half as many elements as A using A, \dots, E , their complements, unions, intersections, and so on.
 - (d) List the following sets: $A \cap (B \cup C)$, $A \cup (D \cap E)$, $(A \cap B) \cup (D \cap E)$, $C \cap D \cap E$, $A \cup C \cup D$, $A \cap B \cap C \cap D \cap E$, $A \cap C \cap E$, and $C \cap (D \cup E)$.
3. Using the universe and sets A, B, \dots, E of the previous problem, identify the following sets and then show that the indicated identity is valid:
- (a) $(D \cup E)$, $(C \cap D)$, $(C \cap E)$: $C \cap (D \cup E) = (C \cap D) \cup (C \cap E)$.
 - (b) $(A \cup C)$, $(A \cap E)$, $(C \cap E)$: $(A \cup C) \cap E = (A \cap E) \cup (C \cap E)$.
 - (c) $(B \cup D)$, B^c , D^c : $(B \cup D)^c = B^c \cap D^c$.
 - (d) $(A \cap E)$, A^c , E^c : $(A \cap E)^c = A^c \cup E^c$.
 - (e) $(B \cap E)$, $(A \cup B)$, $(A \cup E)$: $A \cup (B \cap E) = (A \cup B) \cap (A \cup E)$.
4. Suppose that you are designing a version of the magic trick with the numbers 0, 1, . . . , 7. Furthermore, you have already constructed two cards labeled E and F where in the notation of set theory

$$E = \{1, 2, 3, 4\} \quad \text{and} \quad F = \{3, 4, 5, 6\}.$$

Construct a single card G that will enable you to successfully perform the trick. Is G uniquely determined, that is, is there choice about what numbers can be put on G ?

5. Suppose that you are again designing a version of the magic trick with the numbers 0, 1, . . . , 7. What set theoretic properties must two cards E and F satisfy so that it is possible to construct a third card G with which the magic trick can be played?
6. From a universe of seven objects, find seven sets each containing three objects such that each object is contained in three sets and the intersection of any pair of sets consists of one object.
7. Let the universe consist of all five-bit binary fractions. Suppose that $A = \{x: x \geq \frac{1}{2}\}$, $B = \{x: x \text{ has a 1 in its } \frac{1}{4} \text{ column}\}$, $C = \{x: x \text{ has an odd number of 1s in its representation}\}$, $D = \{x: \text{the last two bits of } x \text{ are } 0\}$, and $E = \{x: \frac{1}{4} < x < \frac{3}{4}\}$.
 - (a) Describe the complementary set in each case.
 - (b) List or describe $A \cap B$, $A \cup B$, $A \cap C$, $A \cup C$, $A \cap D$, $A \cap E$, $B \cap D$, $B \cup D$, $C \cap D$, $C \cup E$, and $D \cup E$.
8. Using the universe and sets A, B, \dots, E of the previous problem, identify the following sets and then show that the indicated identity is valid.

- (a) $(D \cup E), (C \cap D), (C \cap E)$: $C \cap (D \cup E) = (C \cap D) \cup (C \cap E)$.
 (b) $(A \cup C), (A \cap E), (C \cap E)$: $(A \cup C) \cap E = (A \cap E) \cup (C \cap E)$.
 (c) $(B \cup D), B^c, D^c$: $(B \cup D)^c = B^c \cap D^c$.
 (d) $(A \cap E), A^c, E^c$: $(A \cap E)^c = A^c \cup E^c$.
 (e) $(B \cap E), (A \cup E), (A \cup B)$: $A \cup (B \cap E) = (A \cup B) \cap (A \cup E)$.

1:6 PICTURES OF SETS

Given two sets, A and B , in the same universe we can form their union and intersection as indicated in Section 5. We can also form their **difference**, which is defined as follows:

$$A - B = \{x: x \in A \text{ and } x \notin B\}.$$

Thus $A - B$ is a subset of A and is sometimes called the **relative complement** of B with respect to A . This is because if you narrow your viewpoint to think of A as the whole universe, then it is natural to restrict B to $A \cap B$. Now, in the A -universe, $B^c = A - B$. Notice also that this provides us with an alternative way to represent A^c as $U - A$, where U is the universe under consideration.

Example 6.1. Let A and B be defined as in the original card trick.

$$A - B = \{x: x \in A \text{ and } x \notin B\} = \{8, 9, 10, 11\}.$$

Note that

$$B - A = \{x: x \in B \text{ and } x \notin A\} = \{4, 5, 6, 7\}.$$

We shall study relations between sets and shall want to establish the validity of certain assertions. For example, we claim that it is always true that $A - B$ and $B - A$ are **disjoint**, that is, they have no element in common; in symbols

$$(A - B) \cap (B - A) = \emptyset.$$

How can we be sure that this statement is always true? It was true in the case of Example 6.1, but we need a general proof like the following. Suppose that x is an element of $(A - B)$. Then x is in A , but not in B . Consequently, x is not in B and cannot be an element of $(B - A)$, which is a subset of B . Thus the set $(A - B)$ has no element in common with the set $(B - A)$, and so the intersection is empty.

Writing a correct proof is more complicated than working out a specific example. The reason for this complexity is that it is necessary to chase down all the definitions and all possible cases. English is a somewhat clumsy vehicle with which to do this. These sorts of logical arguments are made easier both to construct and to read when accompanied by a picture.

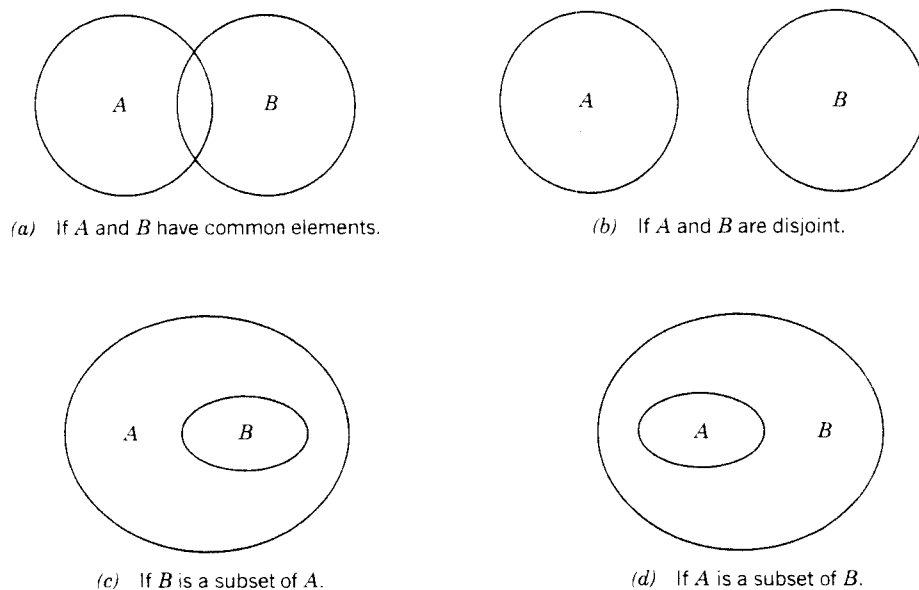


Figure 1.3

To obtain a picture of a particular statement concerning sets, we represent the sets in question as regions of the plane. For example, the set A can be conveniently thought of as all points that are inside of or on the boundary of a circular region. Thinking of B in the same way, we can picture these two sets at the same time with one of the diagrams in Figure 1.3.

We concentrate on the first picture, since in some sense it represents the most general situation for two sets. We label the various regions of the plane with the sets they represent (Figure 1.4). These pictures are called **Venn diagrams**. Frequently, they are made more useful by appropriate shading of the basic regions.

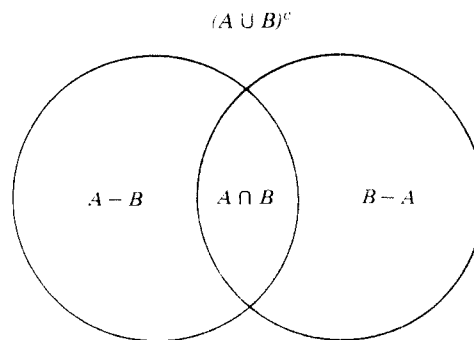


Figure 1.4

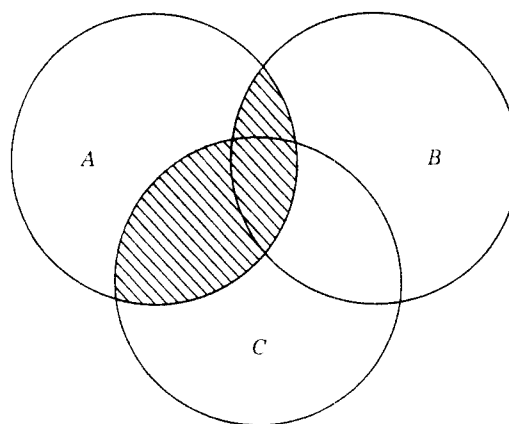


Figure 1.5

Example 6.2. Given sets A , B , and C , Figure 1.5 shows a shaded Venn diagram that highlights the set $A \cap (B \cup C)$.

Question 6.1. Draw shaded Venn diagrams for the following: (a) $(A \cup B)^c$, (b) $A^c \cup B^c$, (c) $(A \cap B)^c$, (d) $A^c \cap B^c$, (e) $A \cup ((B \cap C)^c)$, and (f) $(A \cup B)^c \cap (A \cup C)^c$.

Again we choose to draw the circles as mutually and partially overlapping. If we knew more about specific properties of the sets, for example, that $A \subseteq B$ or that $B \cap C$ is empty, then we could incorporate these properties in the picture, but when the sets are unspecified the drawing in Figure 1.5 is most useful.

From the Venn diagram in Figure 1.5 you might notice, for example, that

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$$

The Venn diagram convinces us of this equality but does not prove the result. For example, does the equality still hold when $A \cap B$ is empty or $B \subseteq C$? To be certain of this statement, we construct an abstract and fully general proof.

There is a straightforward strategy to prove that two apparently different sets are equal. If the two sets are called V and W , first take an element in V and show that it must be in W ; thus $V \subseteq W$. Then take an element in W and show that it must be in V . Then $W \subseteq V$, and we conclude that $V = W$.

We follow this strategy here. Let x be in $A \cap (B \cup C)$. Then x is in A and in addition x is in either B or C (or both). Thus either x is in A and B or x is in A and C ; that is, x is in $(A \cap B) \cup (A \cap C)$. Conversely, if x is in $(A \cap B) \cup (A \cap C)$, then x is in $A \cap B$ or x is in $A \cap C$. Since both B and C are subsets of $B \cup C$, we have x is in $A \cap B \subseteq A \cap (B \cup C)$ or x is in $A \cap C \subseteq A \cap (B \cup C)$ as desired. \square

We have completed a proof of the following theorem. We mark the end of a proof with a square box. It often helps to announce that what was supposed to be proved has been proved.

Theorem 6.1. If A , B , and C are sets in a universe U , then

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$$

Question 6.2. From your Venn diagrams of Question 6.1 what other pairs of seemingly different sets are in fact the same? Prove that these sets are equal.

EXERCISES FOR SECTION 6

1. Draw a Venn diagram of your three-card magic trick from Exercise 5.4. Label points with the integers $0, 1, \dots, 7$ and show in which region each of these points lies.
2. Copy the Venn diagram in Figure 1.4 and explain why

$$(A \cup B) = (A - B) \cup (B - A) \cup (A \cap B).$$

Explain why each pair of the sets $(A - B)$, $(B - A)$, and $(A \cap B)$ is disjoint. We say that $A \cup B$ is the **disjoint union** of $(A - B)$, $(B - A)$, and $(A \cap B)$, and that these three sets **partition** $A \cup B$.

3. Draw a Venn diagram that portrays arbitrary sets A , B , and C , and shade the region that represents the set $A \cup (B \cap C)$. Then on a separate Venn diagram shade in the sets $A \cup B$ and $A \cup C$, and show that

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C).$$

Finally, give a proof that the last equality holds for all sets A , B , and C .

4. Draw shaded Venn diagrams for each of the following:

$$(a) (A \cap B \cap C)^c \quad (b) (A \cup B \cup C)^c$$

$$(c) A^c \cup B^c \cup C^c \quad (d) A^c \cup (B^c \cap C^c)$$

$$(e) A^c \cap (B^c \cup C^c) \quad (f) A^c \cap B^c \cap C^c$$

From these diagrams find pairs of sets that are equal.

5. Draw shaded Venn diagrams for each of the following:

$$(a) (A - B) \cap C \quad (b) A \cup (B - C)$$

$$(c) A \cap (B - C) \quad (d) (A - B) \cup C$$

$$(e) (A \cap C) - (B \cap C) \quad (f) (A \cup C) - (B \cup C)$$

$$(g) (A \cup B) - (A \cup C) \quad (h) (A \cap B) - (A \cap C)$$

6. Suppose that A , B , and C are three sets such that $A \subseteq B$ and $B \cap C$ is the empty set. Draw a Venn diagram that illustrates this situation. Next draw a Venn diagram that illustrates the case when $A \subseteq B$ and $A \cap C$ is the empty set.
7. Suppose that A , B , C , and D are four sets in the same universe with the property that A and D do not intersect while both B and C intersect both A and D as well as each other. Draw a Venn diagram that pictorially represents this situation; label it and shade in the region $(A \cup D) \cap (B \cap C)$.
8. Given sets A , B , and C in the same universe, draw a Venn diagram to indicate the set D , where

$$D = (B \cap (A \cup C)) \cup (C \cap A).$$

Give a proof that $A \cap B \cap C \subseteq D$.

1:7 SUBSETS

In the previous section we saw how two or three sets, say A , B , and C , could be combined to form new sets like $A - B$, $A \cap (B \cup C)$, and $(A \cap B)^c$. Beginning with only one set A , we can also derive a variety of different sets, for example, A^c and the subsets of A .

If $A = Z$, the integers, there are infinitely many subsets of Z . For instance, consider all sets of the form $\{i, i + 1\}$ for i in Z . But if the set A contains a finite number of objects, then there can only be a finite number of different subsets of A . If A contains n objects, A is called an **n -set**. We investigate now how many subsets an n -set has and how we can go about finding all of them.

Example 7.1. Suppose that F is the set of fruit in my refrigerator at this moment; F consists of one apple, one banana, and one cantaloupe. Expressed more briefly,

$$F = \{a, b, c\}.$$

Then the subsets of F consist of the choices I have in selecting fruit for dessert, and these choices vary from the extremes of the empty set (i.e., no fruit for dessert) to the whole set F (i.e., eating all three pieces of fruit.)

Question 7.1. Show that there are eight possible fruit desserts in Example 7.1 by listing all subsets of F .

A set A with only one element has exactly two subsets: itself and the empty set. But if $A = \{a, b\}$, then we find four subsets: A , $\{a\}$, $\{b\}$, and the empty set \emptyset .

1 SETS AND ALGORITHMS: AN INTRODUCTION

From your work in Question 7.1 it should seem plausible that if A is any set containing three objects, then there are eight subsets of A . In general if A is an n -set, then A has 2^n subsets, a result we shall demonstrate later in this section and prove rigorously in Chapter 2. We want to study such a general n -set A , where n is a positive integer. Typically, we describe the set by

$$A = \{a_1, a_2, \dots, a_n\},$$

where a_1, a_2, \dots, a_n stand for the n elements in A . These may be the numbers $\{1, 2, \dots, n\}$, or they may stand for a different set of numbers like $\{1, 5, 17, 25, \dots, 94\}$, or they may stand for names of pieces of fruit as in $\{\text{apple, banana, } \dots, \text{quince}\}$. Here are two methods to list all subsets of a general set like A .

Problem. Given an n -set $A = \{a_1, a_2, \dots, a_n\}$, list all 2^n subsets of A .

Response 1

STEP 1. List the empty set

STEP 2. Set $j := 1$

STEP 3. List all subsets of A that contain j elements

STEP 4. If $j < n$, set $j := j + 1$ and go to step 3; otherwise, go to step 5

STEP 5. Stop.

Is it clear how to carry out step 3? For example, if A contains 17 elements and $j = 9$ we must list all subsets that contain exactly 9 elements. But how? (A subset containing j elements is known as a **j-subset**. In Chapter 3 we'll study the problem of counting and constructing j -subsets of an n -set, for j an arbitrary integer between 0 and n .) For now Response 1 is too imprecise for us to call it an algorithm or to use it effectively.

The idea of the next response is to begin by listing all subsets of $\{a_1\}$; we've seen above how to do this. Then we add to the list those subsets of $\{a_1, a_2\}$ that have not been listed so far, namely those subsets that contain a_2 . Then we list additional subsets of $\{a_1, a_2, a_3\}$ that contain a_3 and therefore have not been listed so far. More generally, once we've listed all subsets of $\{a_1, a_2, \dots, a_j\}$, then we just need to add in the subsets of $\{a_1, \dots, a_j, a_{j+1}\}$ that contain the last element a_{j+1} .

Response 2 (Algorithm SUBSET)

STEP 1. List the empty set

STEP 2. Set $j := 1$

- STEP 3. For each subset B listed so far, create and list the subset $B \cup \{a_j\}$
 {At this point we've listed all subsets of $\{a_1, \dots, a_j\}$ }
 STEP 4. If $j < n$, set $j := j + 1$ and go to step 3; otherwise go to step 5
 STEP 5. Stop.

Example 7.2. We apply the algorithm SUBSET to the set $A = \{a_1, a_2\}$ with $n = 2$. In Table 1.4 we trace for each step the value of the variable j and show the subsets as they are produced.

Table 1.4

Step	Value Assigned to j	List of Subsets
1		\emptyset
2	1	
3		$\emptyset \cup \{a_1\} = \{a_1\}$
4	2	
3		$\emptyset \cup \{a_2\} = \{a_2\}$ $\{a_1\} \cup \{a_2\} = \{a_1, a_2\}$
4	2	
5	STOP	

Question 7.2. With $A = \{a_1, a_2, a_3\}$ use SUBSET to list all of A 's subsets.

How do we know SUBSET is a correct solution? We don't. We only have evidence from the examples we've done, $n = 2$ and 3. Perhaps we have confidence that this approach will continue to work for larger n . To be certain of the validity of SUBSET, we need a proof of its correctness; however, we must defer the proof until Chapter 2 where we'll develop more proof techniques.

One additional important set that is constructed from a single set is known as the Cartesian product. We have been stressing that order in sets and subsets does not matter, but there are many instances when order does matter. For example, in the magic trick an answer "yes, yes, no, yes" gives us information about a number on cards A , B , C , and D (in that order), and it is important to the game that the responses are given in the correct order. Similarly, in the binary number 1101 and the decimal number 29 the order in which the digits are presented is crucial to our representation of the numbers. The Cartesian product is exactly the idea that we need.

If A is any set, we define the **Cartesian product** $A \times A$ to be the set of all ordered pairs (s, t) such that s and t are elements of A . Symbolically,

$$A \times A = \{(s, t) : s \text{ and } t \in A\}.$$

Notice that we have called these **ordered pairs**; that is, we consider (s, t) to be different from the pair (t, s) because the order of the elements matters in this setting. Also notice that we may have a pair of the form (s, s) as well as (s, t) with $s \neq t$.

Similarly, we define

$$A \times A \times A = \{(s, t, u): s, t, \text{ and } u \in A\},$$

and, in general, for any positive integer n we define the n -fold Cartesian product

$$\begin{aligned} A \times A \times \cdots \times A &= A^n \\ &= \{(a_1, a_2, \dots, a_n): a_i \in A \text{ for } i = 1, \dots, n\}. \end{aligned}$$

An element of $A \times A \times A$ is often called an **ordered triple** and an element of A^n is known as an (ordered) **n -tuple**.

Example 7.3. Let $A = \{0, 1\}$. Then $A \times A$ consists of the four ordered pairs $(0, 0)$, $(0, 1)$, $(1, 0)$, and $(1, 1)$. We may associate with each of these a binary number with exactly two digits. A^n consists of all n -tuples of zeros and ones and thus corresponds with all binary numbers of length n .

Without listing elements, we could have seen that $A \times A$ contains four pairs by the multiplication principle: In each pair (s, t) there are two choices for s and, independent of these choices, there are two choices for t . The multiplication principle also tells us $A \times A \times A$ contains $2 \cdot 2 \cdot 2 = 8$ triples and that $A \times A \times A \times A$ contains 16 4-tuples. Generalizing this process to A^n , A^n contains 2^n n -tuples. From this we conclude also that there are 2^n binary numbers of length n .

Question 7.3. If $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, describe $A \times A$, $A \times A \times A$, and A^n for n an arbitrary positive integer.

Question 7.4. If $A = \{a, b, c\}$, list all elements of $A \times A$. If A contains r objects, how many objects does $A \times A$ contain? How many does $A \times A \times A$ contain? A^n ?

Example 7.4. Let $A = \mathbb{Z}$. Then $\mathbb{Z} \times \mathbb{Z}$ is the set of ordered pairs (i, j) , where i and j are integers. We often think of these as points in the coordinate plane with both coordinates being integers. If $A = \mathbb{R}$, the real numbers, then $\mathbb{R} \times \mathbb{R}$ gives us the real, 2-dimensional coordinate plane and $\mathbb{R} \times \mathbb{R} \times \mathbb{R}$ or \mathbb{R}^3 is (real) 3-dimensional coordinate space.

When $A = \{0, 1\}$, the Cartesian products $A \times A$, $A \times A \times A$, \dots and A^n are, perhaps surprisingly, related to subsets of a 2-set, subsets of a 3-set, \dots and subsets of an n -set. Suppose that S is an n -set, and that its elements are listed in

a fixed order, say x, y, z, \dots . For each subset T of S we may construct an n -tuple of 0s and 1s (an element of A^n) whose first entry is a 1 if and only if x , the first element of S , is contained in T . Next, the second entry in the n -tuple is a 1 if and only if y , the second element of S , is in the subset T , and so on. Thus each subset is described by the n -tuple or the string of binary digits as derived above. (The n -tuple is also called the **bit vector** or the **characteristic function** of the subset.) A subset containing exactly i elements corresponds to an n -tuple containing i ones and $(n - i)$ zeros. Conversely, every n -tuple of A^n corresponds to a unique subset of S : For $j = 1, 2, \dots, n$ the subset contains the j th element of S if and only if the j th entry of the n -tuple equals 1.

Question 7.5. Given the universal set $S = \{u, v, w, x, y, z\}$, explain why the bit vector of $T = \{u, v, y\}$ is 110010. Then find the bit vectors of the subsets $\{u, w, z\}$ and $\{x, y, z\}$. Explain why the subset corresponding to the string 000001 is $\{z\}$. Then find the subsets corresponding to the strings 000010 and 111100.

Now we see why there are 2^n subsets of an n -set. We know from Example 7.3 that there are 2^n elements in A^n . Each n -tuple of A^n corresponds to one subset of an n -set, and every subset of an n -set corresponds to exactly one n -tuple. Thus there are as many subsets of an n -set as there are n -tuples in $A^n = \{0, 1\}^n$.

Cartesian products also generalize to different sets. Specifically, we define

$$A \times B = \{(s, t): s \in A \text{ and } t \in B\}$$

and

$$A \times B \times C = \{(s, t, u): s \in A, t \in B, \text{ and } u \in C\}.$$

An ordered list of elements taken from one or several sets is called by a variety of names. We have previously called a binary number a string and a sequence of zeros and ones. We shall also call an n -tuple a **vector** and an **array**, depending upon the convention in the particular example.

The Cartesian product is familiar from coordinate geometry of real 2- and 3-dimensional space. This product is important in computer science and arises, for example, in presentations of abstract conceptions of computers, called Turing machines.

EXERCISES FOR SECTION 7

1. Let $A = \{a, b, c, d, e\}$. List all subsets that contain one element, then all subsets with two elements, next all subsets containing three elements, and finally all subsets that contain four elements. How many subsets have you listed?

1 SETS AND ALGORITHMS: AN INTRODUCTION

2. Given a set with n elements, $A = \{a_1, a_2, \dots, a_n\}$, explain why A has exactly n subsets that contain one element. Then write an algorithm that will list all subsets of A containing just one element.
3. A subset of an n -set A that contains $n - 1$ elements can be formed by omitting just one element of A . Explain why A contains n subsets with $n - 1$ elements and then explain how to list them.
4. List and count all subsets of A that contain two elements when $A = \{a_1, a_2, a_3\}$, when $A = \{a_1, a_2, a_3, a_4\}$, and when $A = \{a_1, a_2, a_3, a_4, a_5\}$.
5. Explain why, for a positive integer n , the set $A = \{a_1, a_2, \dots, a_n\}$ contains $n(n - 1)/2$ subsets with two elements.
6. Given the set A as in Exercise 5, design an algorithm that will list all subsets of A that contain exactly two elements.
7. If $A = \{a_1, a_2, \dots, a_n\}$, determine a formula for the number of subsets of A that contain $n - 2$ elements.
8. If A is an n -set, determine a formula for the number of subsets of A that contain exactly three elements.
9. In Exercise 5.1 we considered the universe of all two-letter “words.” Find a set S such that this universe can be described also as $S \times S$. Then find a set T such that the subset D of that problem can be described as $T \times T$. Is there a set V such that the subset A can be described as $V \times V$?
10. List all four-digit binary numbers. Then associate each with a subset of $A = \{a, b, c, d\}$ and check that all subsets of A have been listed once and only once.
11. Suppose that $A = \{d, e, f\}$ and $B = \{g, h\}$. List all elements of $A \times B$. If A contains r elements and B contains s elements, how many elements are contained in $A \times B$? Justify your answer. If, in addition, C contains t elements, determine the number of elements in $A \times B \times C$.
12. Canadian zip codes are always six symbols long, alternating between letters and digits, beginning with a letter; for example, H4V-2M9 is a valid zip code. Find sets A and B and express the universe of all possible Canadian zip codes as the Cartesian product of A s and B s. How many zip codes are possible?
13. Massachusetts license plates are six symbols long and each symbol may be either a letter or a digit. Express the universe of all possible license plates as a Cartesian product of an appropriate set or sets. How big is this universe?
14. Suppose that my refrigerator contains four pieces of fruit: two apples, one banana, and one cantaloupe. List all possible fruit desserts. By a fruit dessert I mean a collection of fruit (possibly empty). Next suppose that I have five pieces of fruit: two apples, one banana, one cantaloupe, and one damson plum.

List and count all possible fruit desserts. In general, given the so-called **multiset**

$$A = \{a_1, a_1, a_2, a_3, \dots, a_n\}$$

with the first element repeated, find a formula for the number of different subsets of A . The two copies of a_1 are considered to be indistinguishable.

1:8 SET CARDINALITY AND COUNTING

A set A is said to be **finite** if it consists of a finite number of objects. An m -set A is a finite set containing m objects for m , a positive integer. We also say that A has **cardinality** m and write $|A| = m$. In our original magic trick the universe has cardinality 16 while each of the sets A , B , C , and D has cardinality 8.

Example 8.1. Let the universe consist of the positive integers less than 30. If $A = \{x: x \text{ is even}\}$, $B = \{x: x \text{ is divisible by } 3\}$, and $C = \{x: x \text{ is divisible by } 5\}$, then $|A| = 14$, $|B| = 9$, and $|C| = 5$.

Example 8.2. Suppose that we wanted to know the cardinality of $A \cup B$ from the previous example. $A \cup B = \{x: x \text{ is divisible by } 2 \text{ or } 3\}$. We list the elements of $A \cup B$.

$$A \cup B = \{2, 3, 4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 22, 24, 26, 27, 28\}.$$

Note that the cardinality of $A \cup B$ equals 19. This, at first, may seem somewhat strange. We took the union of a 14-element set and a 9-element set and came up with a 19-element set.

Question 8.1. In the context of Example 8.1, find $|A \cup C|$ and $|B \cup C|$.

Question 8.2. In the original magic trick what is the cardinality of each union of two cards?

We shall frequently have occasion to count the number of elements in particular sets, and often these sets can be written as the union (or intersection) of simpler sets. The counting question becomes: Given the cardinalities of two sets A and B , what is the cardinality of their union (or intersection)?

Question 8.3. If possible, find a 5-set A and a 3-set B whose union is a set of cardinality (a) 4, (b) 5, (c) 6, (d) 7, (e) 8, and (f) 9.

In trying to answer the above question, you probably came up with some conclusions about the possible cardinalities of the union of two sets. If you look back at Example 8.2 and Questions 8.1 and 8.2, the following result seems plausible.

Theorem 8.1. Given sets A and B in the same universe.

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

Example 8.2 (continued). We know that $|A| = 14$ and $|B| = 9$.

$$\begin{aligned} A \cap B &= \{x: x \text{ is divisible by both 2 and 3}\} \\ &= \{x: x \text{ is divisible by 6}\} = \{6, 12, 18, 24\}. \end{aligned}$$

Thus $|A \cap B| = 4$. In this particular instance we verify Theorem 8.1 by noting that $14 + 9 - 4 = 19$.

Proof of Theorem 8.1. We must show that each element of $A \cup B$ contributes exactly one to $|A| + |B| - |A \cap B|$ (see Figure 1.6).

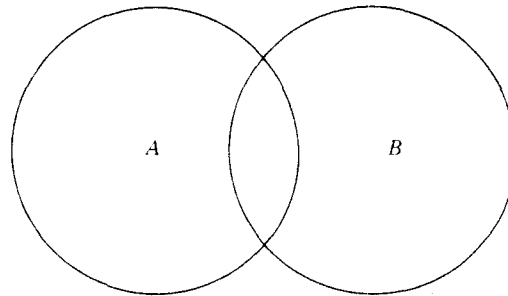


Figure 1.6

Case i. Suppose that x is a member of only one set, say A . Then x is counted in $|A|$, but not in $|B|$ nor in $|A \cap B|$, since $A \cap B \subseteq B$. Thus x contributes a count of one to $|A| + |B| - |A \cap B|$. If x is in B , but not in A , a similar argument suffices.

Case ii. Suppose that x is an element of both A and B . Then x is counted once in $|A|$, once in $|B|$, and once in $|A \cap B|$. Thus it contributes $1 + 1 - 1 = 1$ to $|A| + |B| - |A \cap B|$, just as we wanted. \square

Example 8.3. How many seven-digit telephone numbers are there that begin 584- and contain at least one 0 and at least one 1? Suppose that we let A be the

set of all such numbers that contain a 0 and B be the set of all numbers that contain a 1. The telephone numbers we're looking for are in both A and B . Thus we want $|A \cap B|$. By Theorem 8.1, $|A \cap B| = |A| + |B| - |A \cup B|$. First let's find $|A|$. (See Exercise 1.11.) There are 10^4 possible telephone numbers, since there are 10 choices for each of 4 numbers. If we exclude 0, then there are 9 choices for each of 4 numbers. Thus there are 9^4 numbers that contain no 0. Thus $|A| = 10^4 - 9^4$. Similarly, $|B| = 10^4 - 9^4$. What about $|A \cup B|$? There are 8^4 numbers that contain neither a 0 nor a 1 and so $|A \cup B| = 10^4 - 8^4$. Thus

$$|A \cap B| = (10^4 - 9^4) + (10^4 - 9^4) - (10^4 - 8^4) = 974.$$

Question 8.4. A joint meeting of Discrete Mathematics and Introductory Computer Science had 232 students. If 146 students are enrolled in the mathematics course and 205 students are enrolled in the computer science course, how many students are enrolled in both courses at once?

We present a formula that generalizes the pattern of Theorem 8.1 from two sets to three sets; the proof is similar. See Exercise 12.

Theorem 8.2. Given sets A , B , and C in the same universe, then

$$|A \cup B \cup C| = |A| + |B| + |C| - (|A \cap B| + |A \cap C| + |B \cap C|) + |A \cap B \cap C|.$$

(See Figure 1.7.)

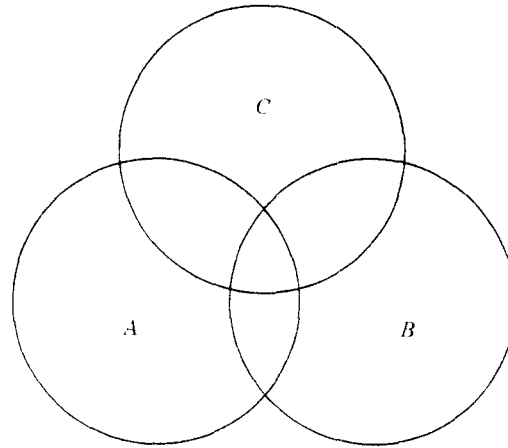


Figure 1.7

Example 8.4. Let A , B , and C be the sets from the original magic trick. Then $A \cup B \cup C = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$ contains 14 elements. As we have seen $|A| = |B| = |C| = 8$. We have also noted that the intersection of any two of these sets has cardinality equal to 4. You can check that $A \cap B \cap C = \{14, 15\}$. Then

$$|A \cup B \cup C| = 8 + 8 + 8 - (4 + 4 + 4) + 2 = 14.$$

The Principle of Inclusion and Exclusion. Theorems 8.1 and 8.2 are both instances of a general counting result known as the Principle of Inclusion and Exclusion or P.I.E. for short. The same type of counting formula applies for 4 or 5 or k , an arbitrary number, sets. See Exercise 13.

EXERCISES FOR SECTION 8

- Refer to the sets in Exercise 5.1. Determine the indicated set cardinalities and then check the validity of Theorem 8.1:
 - $|A \cup B| = |A| + |B| - |A \cap B|$.
 - $|A \cup C| = |A| + |C| - |A \cap C|$.
 - $|B \cup D| = |B| + |D| - |B \cap D|$.
 - $|D \cup E| = |D| + |E| - |D \cap E|$.
- Refer to the sets in Exercise 5.2. Determine the indicated set cardinalities and then check the validity of Theorem 8.2:
 - $|A \cup C \cup D| = |A| + |C| + |D| - |A \cap C| - |A \cap D| - |C \cap D| + |A \cap C \cap D|$.
 - $|C \cup D \cup E| = |C| + |D| + |E| - |C \cap D| - |C \cap E| - |D \cap E| + |C \cap D \cap E|$.
- Of the 876 students living in the Quadrangle, 530 have completed Introductory Computer Science, 364 have completed Calculus II, and 287 have completed Chemistry I. Of course, lots of students take more than one of these courses. In fact, 213 have completed both mathematics and computer science, 164 have completed both mathematics and chemistry, 116 have completed chemistry and computer science, and 103 have completed all three courses. How many students living in the Quad have completed none of these three courses?
- There were 184 students enrolled in Introductory Computer Science last fall. Of these 112 will take Data Structures, 84 will take Foundations of Computer Science, and 46 will take Assembly Language. Of the total, 66 will take both Foundations and Data Structures, 37 will take Assembly Language and Data Structures, and 30 will take Assembly Language and Foundations. If 45 of the original students take no additional computer science, how many students take all three of the intermediate courses?
- The registrar informs us that three years ago 119 students enrolled in Introductory Computer Science. In the following semester, of these 119 students, 96

took Data Structures, 53 took Foundations, and 39 took Assembly Language. Also 38 took both Data Structures and Foundations, 31 took both Foundations and Assembly Language, 32 took both Data Structures and Assembly Language, and 22 brave souls took all three courses. We claim that the registrar must have made an error. Why?

6. How many seven-digit telephone numbers are there that begin with 584- and that contain a 0, a 1, and a 2?
7. Suppose that A and B are sets in the universe U . Find a way to express the cardinality of the set $A^c \cup B^c$ in terms of the cardinality of U , A , B , and combinations of these sets.
8. Consider the universe of all possible strings of six letters made from the letters a, b, c, d, e, f with no repetitions of letters. How many such strings are there in total? How many of these are such that the first letter is neither “ a ” nor “ b ” and the last letter is neither “ e ” nor “ f ”? (*Hint*: Let A be the set of all these words that do begin with an a or b and B the set of those that end with an e or f . Then we are looking for the cardinality of $A^c \cap B^c$.)
9. A group of 100 students was surveyed to determine the students’ interest in winter sports. It was found that 69 liked downhill skiing, 38 liked cross-country skiing, 75 liked skating, and 15 liked none of these sports. On further questioning it was determined that 35 liked both kinds of skiing, 30 liked cross-country skiing and skating, and 42 liked downhill skiing and skating. How many like to ski, either downhill or cross-country? How many like all three sports?
10. In a sample of 100 students, 43 like avocados, 71 like radishes, and 36 like olives in their salad. Each student liked at least one vegetable. If 26 students like both avocados and radishes, 16 students like avocados and olives while 22 like radishes and olives, how many students like all the ingredients in an avocado, radish, and olive salad?
11. Here is an alternative proof of Theorem 8.1. Give reasons for each of the following steps:

$$\begin{aligned} \text{STEP 1. If } A \cap B = \emptyset, \text{ then } |A \cup B| &= |A| + |B| \\ &= |A| + |B| - |A \cap B|. \end{aligned}$$

$$\text{STEP 2. } B = (A \cap B) \cup (B - A).$$

$$\text{STEP 3. } |B| = |A \cap B| + |B - A|.$$

$$\text{STEP 4. } |B - A| = |B| - |A \cap B|.$$

$$\text{STEP 5. } A \cup B = A \cup (B - A).$$

$$\text{STEP 6. } |A \cup B| = |A| + |B - A|.$$

$$\text{STEP 7. } |A \cup B| = |A| + |B| - |A \cap B|.$$

12. Prove Theorem 8.2. (*Hint:* Let x be an element of $A \cup B \cup C$. Show that x 's contribution is exactly one. Divide the proof into three cases depending on how many of A , B , and C contain x .)
13. Find an expression for the cardinality of the union of four sets in terms of the cardinalities of the sets and various intersections.

1:9 FUNCTIONS

The concept of a function is fundamental to both mathematics and computer science and will be used throughout this book.

Definition. A **function** f is a mapping from a set D to a set T with the property that for every element d in D , f maps d to a unique element, denoted $f(d)$, of T . Here D is called the **domain** of f , and T is called the **target** of f . We write $f: D \rightarrow T$. We also say that $f(d)$ is the **image** of d under f , and we call the set of all images the **range** R of f . In set notation

$$R = \{f(d) : d \in D\}.$$

Note that $R \subseteq T$.

A mapping might fail to be a function if it is not defined at every element of the domain or if it maps an element of the domain to two or more elements in the range. Figure 1.8 illustrates these ideas.

To define a function f , we must specify its domain D and a rule for how it operates. If the domain is changed, we consider that a new function is formed.

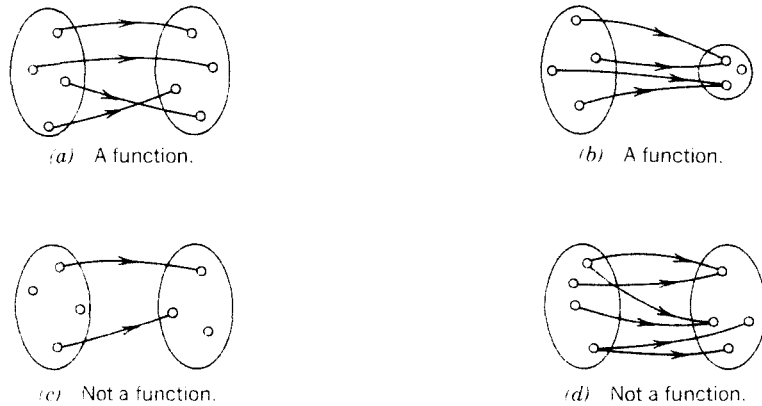


Figure 1.8