CHAPTER

1

Introduction

It is a truth universally acknowledged, that a single man in possession of a good fortune must be in want of a wife.

—Jane Austen — Pride and Prejudice

A typical student just beginning a course in discrete mathematics has no clear idea about the content of the course. This introductory chapter will provide some initial ideas. You will find some of the content of this course to be mathematical ideas that are unfamiliar to you. You will have seen other topics as early as middle school or junior high school.

This chapter contains an introductory section titled What Is Discrete Mathematics? and several interesting problems, one covered in some detail. The others will be covered in more depth in later chapters. The problems represent a sample of the topics covered in discrete mathematics courses; they certainly do not represent all the topics.

1.1 What Is Discrete Mathematics?

The word *discrete* has the following definitions[1] that are relevant to our topic:

1: Constituting a separate entity: individually distinct.
2a: Consisting of distinct or unconnected elements: NONCONTINUOUS.
b: Taking on or having a finite or countably infinite number of values: not mathematically continuous.

Sometimes it is easier to understand a concept by first seeking to comprehend an opposite or a complementary concept. For example, it is easier to grasp the notion of an *irrational number* if we first understand the definition of a *rational number*.

> **DEFINITION 1.1** ***Rational Number***
> A real number, r, is called *rational* if it can be expressed as a ratio of two integers: $r = \frac{p}{q}$, where p and q are integers and $q \neq 0$.

Thus, the rational numbers are just the familiar fractions. In decimal form the fractions are the real numbers that have a finite or a repeating decimal expansion.[2] The irrational numbers are those real numbers that are not rational. They cannot be expressed as a ratio of integers; nor do their decimal expansions ever terminate or form a repeating pattern.

When first trying to understand what discrete mathematics consists of, it may be helpful first to consider what it is not about.[3]

[1] *Webster's Ninth New Collegiate Dictionary*, First Digital Edition.

[2] For example, $\frac{5}{4} = 1.25$ and $\frac{4}{7} = 0.\overline{571428}571428$.

[3] My first instinct is to think about what *indiscrete mathematics* would be. I immediately think about mathematics that "tells all". However, the word that fits such an imprudent approach is spelled *indiscreet*, not *indiscrete*. The word *indiscrete* means "not separated into distinct parts."

Mathematically continuous objects cannot be separated into pieces that are not joined together. You are familiar with many continuous mathematical objects: geometric lines and shapes, the graphs of functions used in Calculus [such as $\sin(x)$], and the real numbers. The real numbers are considered continuous because we can never find two real numbers that have a gap between them that does not contain other real numbers. That is, given any two distinct real numbers, r and t, we can always find another number, s, with $r < s < t$. We can then find another number between r and s, and then another real number in the new interval. The process never terminates.

In contrast, discrete mathematical objects consist of separate pieces. For example, the collection of all words on this page is a discrete set. The set of all integers is discrete (although infinite) since it is not possible to find another integer between 3 and 4 (or any other adjacent pair).

A thermometer is a good illustration of the difference between continuous and discrete. A mercury thermometer is a continuous device. It is capable of expressing every temperature in its range (although we will be unable to visually see the difference between 25.67° and 25.6698°). In contrast, a digital thermometer might only show temperatures to the nearest tenth of a degree. It has a discrete set of temperatures it can represent.

A digital thermometer has something in common with computers and calculators: They are only able to concurrently represent a finite (but very large) set of numbers. There will always be numbers that cannot be exactly represented. For example, it is impossible to represent the number one-tenth as a finite binary fraction. The reason is that its binary expansion is infinite, but a computer cannot store an infinite number of bits.[4] This means that computers and calculators are inherently discrete devices.[5]

One additional idea needs to be explored before an initial definition of discrete mathematics can be stated. The definition of *discrete* quoted previously used the phrase "countably infinite". What does that mean?

> **DEFINITION 1.2** ***Countably Infinite***
>
> A set is called *countably infinite* if its elements can be placed in one-to-one correspondence with the positive integers. That is, every element in the set can be labeled by exactly one positive integer and every positive integer is the label for exactly one element of the set.

You may be surprised to learn that the set of rational numbers which includes the integers as a proper subset, is a countably infinite set.[6] However, the set of all real numbers is not countably infinite (although it is certainly infinite, since it contains the rational numbers as a proper subset).[7]

It is now time for a working definition of discrete mathematics.

> **DEFINITION 1.3** ***Discrete Mathematics***
>
> *Discrete mathematics* is a collection of mathematical topics that examine and use finite or countably infinite mathematical objects.

Although this definition is fairly good, it does not completely capture the true nature of a typical discrete mathematics text or course. In reality, such a course usually

[4] In the decimal system, we can never write the number one-third exactly since the decimal expansion contains an infinite number of 3s. One-tenth works the same way in binary.

[5] Therefore you should not be surprised to see many connections with computer science in a discrete mathematics text.

[6] This assertion will be proved at the end of the chapter.

[7] This means that there is more than one size of *infinite*! *Proper subset* is formally defined on page 20.

consists of a number of mostly unrelated topics that often involve finite or countably infinite mathematical objects. However, probably because the topics are already only tenuously related, several other kinds of topics are usually included in a discrete mathematics course. For example, this text contains a chapter on logic and another on strategies for proving theorems. Neither topic fits neatly into the definition given previously. Nevertheless, they seem to work well as a portion of a discrete mathematics course.

One other point bears notice: Continuous mathematical objects *do* appear in a discrete mathematics course. For example, when linear homogeneous recurrence relations are examined in Chapter 7, the real and complex roots of polynomial equations are quite important. Both number systems are continuous. The original problem, however, is certainly motivated by discrete ideas.

1.1.1 A Break from the Past

Many of the topics in this text may cause some students to feel a bit disoriented. Perhaps it is because this course breaks from the kinds of mathematics you have been studying in the past. Most of your previous mathematics has been centered on the continuous real numbers and the continuous geometry in the Euclidean tradition. Even calculus is an extension of previous work that introduces the continuous notion of a limit.

In this course, you will get a chance to expand your mathematical horizons. You will be exposed to mathematics that does not use any numbers, equations, or functions. You will expand your notion of such seemingly simple topics as counting and graphs.

Another difference from most of your previous mathematical learning is the inclusion of some topics that have become prominent as a result of the computer revolution. Topics, such as the study and analysis of algorithms, recursion, and finite-state machines, all fit nicely under the working definition of discrete mathematics.

Those students who have been successful in previous mathematics classes may initially feel unsettled because their old study strategies may not be completely successful with this new material. However, with a few adjustments (in particular the need to *read the text thoroughly*), they should continue to do well.

Students who have struggled in previous mathematics classes should consider this course as a fresh chance to succeed. Read and discuss the text with friends, get help from the instructor as soon as you need it, and promise yourself to never let the homework slide. With proper effort and course management, this could be a class you enjoy.

1.2 The Stable Marriage Problem

In a remote valley there is a village with some rather strange customs. The community is very close knit and the people desire to preserve their traditional customs. One of the customs that helps preserve community is their unique marriage custom. Rather than relying on the modern Western custom of dating, or the older custom of matchmakers or arranged marriages, the village council decides on marriage partners. Their custom differs from traditional arranged marriages because the arrangements are made for an entire group of young people. The village council waits until all young people of the current generation are old enough to marry. They then assign marriage partners.

The young people certainly have preferences about who they would like to marry; the village council clearly must pay attention to their wishes. However, it is unlikely that everyone will get their first (or even second) choice. The village council also has to consider the larger community. If they make assignments that the young people do not like, it is possible that one or more pairs may elope and marry someone other than the partner chosen by the council. In that case, the new couple will have disobeyed the decision of the community leaders. If they stay in the valley, the unity of the community will be broken. If they leave the valley, the future vitality of the community will be diminished.

How should the council make assignments so that no pair can successfully elope while still respecting the preferences of the young people (as much as possible)? Is such an assignment always possible?

Before starting to find a solution, it will be helpful to create a useful definition.

DEFINITION 1.4 ***Stable Assignment***

Given a collection of n men, $m_1, \ldots, m_n$, and n women, $w_1, \ldots, w_n$, we wish to associate every person with a mate. Suppose that each person ranks the people of the opposite gender by preference with no ties. An *assignment* is one of the possible collections of n couples. An assignment is *stable* if there does not exist a man, m_i, and a woman, w_j, who are not partners but m_i prefers w_j to his assigned bride and w_j prefers m_i to her assigned groom.

1.2.1 Seeking a Solution

Where do we start to obtain a solution to this problem? One helpful approach is to look at some small examples to help guide our intuition. One example will be presented here.

EXAMPLE 1.1

A Set of Preferences with Three Stable Assignments

Consider the set of preferences (Table 1.1) for a group of three women (A, B, C) and three men (X, Y, Z). The first table indicates that A would prefer to marry Y, but if that is not possible Z would be her second choice. The second table indicates that Y prefers B, with C as second choice and A as his least desirable mate.

Let R $\longleftrightarrow$ S represent the statement: "R and S have been assigned to be married". There are six distinct marriage assignments. Three of them are stable (Table 1.2); the other three are not (Table 1.3).

TABLE 1.1 Preferences for a Small Stable Marriage Problem

Female preferences

	1	2	3
A	Y	Z	X
B	Z	X	Y
C	X	Y	Z

Male preferences

	1	2	3
X	A	B	C
Y	B	C	A
Z	C	A	B

TABLE 1.2 The Stable Assignments

Female 1st choice

A	⟷	Y
B	⟷	Z
C	⟷	X

All 2nd choice

A	⟷	Z
B	⟷	X
C	⟷	Y

Male 1st choice

A	⟷	X
B	⟷	Y
C	⟷	Z

TABLE 1.3 The Unstable Assignments

A and Z elope

A	⟷	X
B	⟷	Z
C	⟷	Y

C and Y elope

A	⟷	Y
B	⟷	X
C	⟷	Z

B and X elope

A	⟷	Z
B	⟷	Y
C	⟷	X

Note that in the middle stable assignment (all 2nd choice), A would like to elope with Y, but Y prefers C to A. Similarly, none of the others (in that assignment) can find a willing partner for an elopement. Everyone must stay with the assigned partner, so the assignment is stable. The "all 2nd choice" assignment demonstrates that an assignment can be stable even though every person would prefer to marry someone else.

In the first of the unstable assignments, A prefers Z to X and Z prefers A to B. Thus, A and Z can successfully elope, causing the assignment to be unstable. A would really like to elope with Y, but Y likes C better than A, so Y will not cooperate. The six distinct assignments were found by exhaustive enumeration (i.e., listing all possibilities).[8] ■

[8]This approach is not practical if there are too many more candidates for marriage. A better approach is to develop an algorithm that will always lead to one of the stable assignments.

The example shows that multiple stable assignments may exist. Additional examples would demonstrate that the number of stable and unstable assignments need not be the same. We would also discover that sometimes only one stable assignment is possible (with all other assignments being unstable). No matter how many examples we look at, we would not find any that did not have at least one stable assignment. This suggests that we may be able to prove that a stable assignment always must exist. Let us leave that proof until later and concentrate on finding an algorithm that will produce a stable assignment.

Although mathematics contains a great store of formulas to memorize, most of them were arrived at by people in the past who saw the important relationships. They were often guided by intuition and common sense (coupled with an understanding of the mathematical discoveries that occurred previously). Where might we look for intuition?

One good source of ideas would be current courtship practices. In a Western society, a young man typically must ask a young woman if she is willing to marry him. Most young men ask the young women they find more desirable before asking others they prefer less. The young women accept or reject proposals based on their own preferences. At times, they may ask for additional time to consider the proposal. During that additional time, they need to decide whether to settle with someone who is not their top candidate, to hope that a better proposal arrives real soon, or to simply reject the proposal.

The algorithm that will be presented next draws on these ideas. It was developed in the 1960s by Gale and Shapley [39], who were considering a slightly more general version of the problem.[9]

1.2.2 The Deferred Acceptance Algorithm

The algorithm operates in a series of rounds. The members of one gender become the suitors and the members of the other gender are the suitees. Each suitor proposes to his or her highest-ranked suitee. The suitees wait for all proposals during that round, then reject all but the proposal from the highest ranking suitor among their current string of suitors. That suitor is told to wait for an answer (hence the name *the Deferred Acceptance Algorithm*). The suitors who are waiting for an answer may still be rejected in a later round in favor of a better proposal.

At the start of subsequent rounds, the suitors who were rejected in the previous round each propose to their next highest-ranking suitee. Suitees then reject all but the current favorite, telling that suitor to wait until later for an answer. This concludes the current round; the next round is ready to begin. The process ends when every suitee has exactly one proposal pending. The village council then steps in and declares that the current pairings are final.[10]

> **DEFINITION 1.5** ***Unattached, Viable***
>
> A suitor will be called *unattached* if he or she is not currently waiting for a suitee to respond to a proposal. A suitee is *viable* for a suitor if that suitee has not already rejected a proposal from that suitor.

The algorithm (which, for now, assumes equal numbers of suitees and suitors) can be restated in pseudocode:[11]

[9]The notation [39] is a bibliographic reference.

[10]You should make a mental distinction between the stable marriage (or stable assignment) *problem* and the deferred acceptance *algorithm*. The algorithm is just one procedure for solving the general problem.

[11]Pseudocode is described in Section 4.1. You do not need the details right now.

The Deferred Acceptance Algorithm

round = 1
All suitors are initially considered unattached.
while at least one suitee has no pending proposal, **repeat** the following:
  Each unattached suitor proposes to his or her highest ranked viable suitee.
  Each suitee examines her or his string of suitors, rejecting all except the highest ranked suitor. That suitor is told to wait while the proposal is considered. (The waiting suitor therefore becomes attached for the next round.)
  Add 1 to round.
end while
All suitees accept the proposal of the single suitor waiting for an answer.

Two issues need to be addressed before we can confidently use this algorithm. We must show that the algorithm always terminates, and we must show that it always leads to a stable assignment.

THEOREM 1.6 ***Termination***

The Deferred Acceptance Algorithm always terminates.

Proof: Once a suitee has at least one proposal, that suitee will never have an empty string of suitors (since at each round, the best of the available suitors is kept). At the start of each round, the rejected suitors move to the next suitee on their preference lists. Since there is only a finite number of suitees available, the proposals cannot continue indefinitely.[12] Since no suitee ever keeps more than one suitor, and since the numbers of suitors and suitees are equal, eventually every suitee will have a proposal. ∎

THEOREM 1.7 ***Stability***

The Deferred Acceptance Algorithm always produces a stable assignment.

Proof 1—Suitor's Perspective: Suppose a suitor, X, is unhappy with the mate, A, assigned by the Deferred Acceptance Algorithm. There may be a suitee, B, that X would prefer. The key relationships in the current assignment are

$$X \longleftrightarrow A \qquad\qquad Y \longleftrightarrow B.$$

However, B will already have rejected X at some round of the algorithm (since X would have proposed to every higher-ranking suitee before proposing to A). The rejection was not made foolishly: B rejected X in favor of a valid proposal from a suitor, Y, whom B prefers to X. Thus, any suitee that X prefers to A will already have a mate that ranks higher than X. Consequently, no suitor will find a higher ranking, willing partner with whom to elope.

Proof 2—Suitee's Perspective: Suppose a suitee, A, is unhappy with the mate, X, assigned by the Deferred Acceptance Algorithm. There may be a suitor, Y, that A likes better than X. However, since suitors propose in the order of their preferences, Y must never have proposed to A (otherwise, A would have rejected X in favor of Y). If Y never proposed to A, then Y must have been accepted by a suitee, B, that Y prefers to A. Thus, A will be unable to find a suitor (ranked higher than X) who is willing to elope. The conclusion is that no suitee will be able to find a higher-ranking suitor who is willing to elope. ∎

[12] We will prove in Example 5.1 that at most $n^2 - 2n + 2$ rounds are possible.

1.2.3 Some Concluding Comments

The Stable Marriage Problem and the Deferred Acceptance Algorithm fit nicely into the notion of discrete mathematics presented earlier. The problem involves a finite collection of people, and the algorithm consists of a finite number of well-defined steps. Intuition and careful thinking were used to develop the algorithm. In addition, the presentation has used some other important elements of mathematics: definitions, theorems, and proofs. An interesting aspect of the problem (so far at least) is that no formulas or equations were used, yet the solution process clearly was of a mathematical nature.

The Stable Marriage Problem has not been exhausted; it will appear again in this text.[13] In the exercises for this chapter, you will be asked to extend the algorithm to solve (or attempt to solve) more general versions of the problem. This feature is also common in mathematical investigations. Once a problem has been solved, it is interesting (and often useful) to see if the solution can be generalized. It is also important to abstract the essential features of the problem and solution so that they can be recognized and applied in other situations that are essentially the same.

For example, the problem that Gale and Shapley originally sought to solve did not involve producing stable marriage assignments. Their original interest was in matching graduating high school seniors with colleges. Their problem involved graduates ranking various colleges by preference. Colleges would have quotas for the number of new freshmen to admit. After reading applications, the colleges would rank the graduates. The problem was then to assure matchings of students and colleges so that a student would not receive a scholarship from one school only to accept an offer from a rival college at a later date.

The Deferred Acceptance Algorithm has actually been used in a third setting: matching medical students who are ready to start an internship or residency with cooperating hospitals. The key features of the problem clearly hold: Two groups to match, with each group having clear preferences toward members of the other group.

1.3 Other Examples

The examples in this section will be discussed in more detail in later chapters. They are presented here to indicate some of the diverse areas that this text will cover. The six subsections provide a very small sample of the topics that lie ahead. It might be helpful to pause at this point and make a quick scan of the table of contents to get a more complete idea of the available topics.

1.3.1 A Simple Counting and Probability Example

The simplest way to connect a dvd player to a television is to use an audio video composite cable. The cable has three wires: a yellow video line and a white and a red audio line for stereo audio (see Figure 1.1). Suppose I buy a cable, but am completely color blind, so I cannot properly match the colored cables to the colored connectors on the dvd player and television.

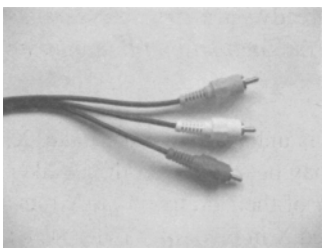


Figure 1.1. A composite cable.

There are two ways I can correctly plug in the cable (since the cable is symmetric on both ends, it does not matter which end is plugged into the television). In how many ways can I incorrectly plug in the cable? What is the probability that I randomly plug it in correctly?

To help think about the question, assume that one end of the cable is labeled A and the other end is labeled B. In how many ways can I plug the A end into the television? I have three choices of wire to plug into the yellow receptacle on the television. Once I have made a choice, there are two wires left that can be plugged into the white receptacle. After I plug one of them in, there is only one choice left for the wire to plug into

[13] See pages 144 and 225.

the red receptacle.

For example, suppose I plug the red wire into the yellow receptacle. Then I can either plug the yellow wire into the white receptacle and the white wire into the red receptacle, or else I can plug the white wire into the white receptacle and the yellow wire into the red receptacle. So once I make the first choice (red wire into yellow receptacle), there are two ways to plug the rest of the cable into the television. I have three choices for the first wire, and two options for each of those choices. After reading Chapter 5, it will be easy to conclude that there are $3 \cdot 2 \cdot 1 = 6$ ways to plug the cable into the television. Table 1.4 lists all of them (the columns represent the six ways to plug in the cable, with the first being the correct arrangement).

TABLE 1.4 Plugging the Cable into the Television

	Cable Wire					
Yellow Receptacle	Y	Y	R	R	W	W
White Receptacle	W	R	Y	W	Y	R
Red Receptacle	R	W	W	Y	R	Y

Now the B side of the cable needs to be plugged into the dvd player. It should be clear that there are six ways to do that. However, each of those 6 ways can be coupled with each of the 6 ways to plug in the A side of the cable. That results in 36 distinct ways to plug in the cable if the A end is connected to the television. If the B side were to be connected to the television, 36 more possibilities would result.

In all, there are 72 ways to plug in the cable Only two of those arrangements result in a correct connection. That is, only $\frac{2}{72} \simeq 3\%$ of the time would I randomly connect the cable properly. It would be better to have a friend plug it in for me.

1.3.2 Sierpinski Curves

During the late 1800s, Georg Cantor was attempting to find an infinite set that was larger[14] than the open interval $(0, 1)$ on the real number line [26]. The obvious set to look at was the open unit square, having corners at the points $(0, 0)$, $(0, 1)$, $(1, 1)$, and $(1, 0)$ in the plane. After extensive effort trying to show that the unit square was indeed larger than the unit interval, Cantor started to suspect that they might be the same size! He eventually proved that they were indeed the same size.

Other mathematicians were able to provide a very elegant kind of proof of this counterintuitive result. They constructed functions that map the points on the interval $[0, 1]$ onto points in the closed unit square in such a manner that the function establishes a one-to-one correspondence. However, such functions must be discontinuous [79, Chapter 1]. It is possible to construct continuous functions that map the interval $[0, 1]$ onto the unit square. (There will be points in the unit square that are mapped onto by multiple points in the unit interval.) Such functions are called space-filling curves. Space-filling curves can be more formally defined by requiring the image of the function to have a nonzero area.[15]

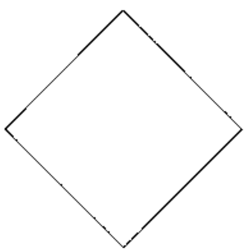


Figure 1.2. S_0.

One very attractive space-filling curve is the Sierpinski curve, which was defined by Waclaw Sierpinski. The Sierpinski curve is actually the limit of a sequence of curves. The curves will be defined in more detail in Section 7.1.5. For now, it is sufficient to consider the first few members of the sequence.

The curves will be labeled $S_0, S_1, S_2, S_3, \ldots$.

Consider the curve S_0 in Figure 1.2. We can think of the diamond shape of the curve as depicting a function that successively maps the subintervals $[0, 0.25]$, $[0.25, 0.5]$,

[14] A more precise definition of *larger* and *the same size* can be made using Definition 12.2 on page 732.

[15] See Sagan, Chapter 1 for more details.

[0.5, 0.75], and [0.75, 1] onto the points in the plane that lie on the line segments joining (0.5, 1) to (1, 0.5), (1, 0.5) to (0.5, 0), (0.5, 0) to (0, 0.5), and (0, 0.5) to (0.5, 1) in the unit square.

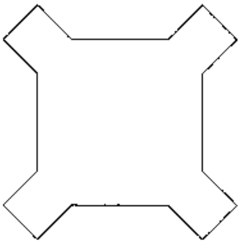


Figure 1.3. S_1.

Similarly, we can view S_1 (Figure 1.3) as the image of a function that maps [0, 1], subdivided into 20 equal-length intervals, onto the 20 line segments that constitute the curve S_1. (Notice that the horizontal and vertical line segments in S_1 are double length and hence count as two segments each.)

The curves in the sequence are sets that are the same size as the interval [0, 1] on the real number line. In addition, each curve in the sequence contains more of the points in the unit square than do the previous curves. Also, none of the curves cross themselves. It can be shown that as n approaches infinity, S_n uses more and more of the points in the unit square. The limit, S, of the sequence uses all the points and establishes a mapping from the interval [0, 1] onto the unit square (proving that the interval cannot be smaller than the square). The graph of the curve would appear to be a solid black square in the plane, but the function would match each point in [0, 1] with exactly one point in the square. The curve S_4 in Figure 1.4 demonstrates this limit to some extent.

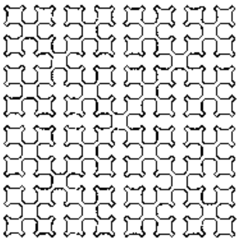


Figure 1.4. S_4.

An alert student might wonder why this topic is in this text, since it clearly is motivated by continuous ideas (rather than discrete ideas). The connection is the mechanism used to draw the curves. That mechanism is a technique called *recursion*. You might consider how to write a program to draw S_4 (or S_6 or, in general, S_n). It does not seem to be an easy task. Using standard methods, it is not. However, using a set of recursive functions, it is quite simple.[16] Recursion is a major topic in typical discrete mathematics courses.

1.3.3 The Bridges of Königsberg

An important branch of mathematics had its origin in a clever puzzle whose solution occurred in 1736 in the town of Königsberg (now named Kaliningrad). The town is built on both sides of the Pregel River and includes an island and a section of land where the river forks. The crude sketch in Figure 1.5 shows the seven bridges that connected the various land masses. The landmasses have been labeled A, B, C, and D.

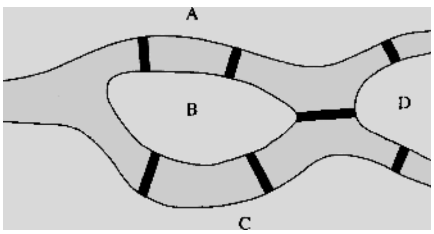


Figure 1.5 The bridges of Königsberg.

A popular puzzle in the town was to produce a tour of the town that crossed each of the bridges exactly once. A really nice solution would provide a tour that began and ended at the same place. Many people tried (and failed) to find a solution. The problem was waiting for Leonhard Euler's masterful solution, which incidentally was the origin of *graph theory*. The details will be presented in Chapter 10.

1.3.4 Kirkman's Schoolgirls

Imagine a schoolmistress with 15 young girls at her boarding school. Each day the schoolmistress lines the girls up in 5 rows of 3 girls each to go for a walk. Is it possible to group the girls so that in the course of 7 walks, each girl will have been in a row

[16] The details appear in Section 7.1.5.

with every other girl exactly once? This problem was proposed in 1850 by the Reverend Thomas P. Kirkman (predating Madeline and Miss Clavel by 90 years).

A solution (there are many) that lists the 5 rows for each of the 7 days is an example of a *balanced incomplete block design*, which is in turn an example of a *combinatorial design*. The techniques for producing combinatorial designs range from guess-and-check through some very sophisticated uses of other mathematical structures (such as vector spaces and finite geometries).

A solution for the Kirkman schoolgirl problem will be presented as Example 8.28 in Section 8.3. A solution to a smaller version of the problem is presented here.

Suppose that there are only 9 schoolgirls and they only walk 4 days each week (perhaps they have a field trip on Wednesdays). It is possible to group them into lines of size 3 so that every girl is with every other girl exactly once per week. If the girls are labeled 1–9, Table 1.5 shows one solution.

TABLE 1.5 An Arrangement of Nine Girls for Weekly Walks

Row	Monday	Tuesday	Thursday	Friday
1	1 2 3	1 4 7	1 5 9	1 6 8
2	4 5 6	2 5 8	2 6 7	2 4 9
3	7 8 9	3 6 9	3 4 8	3 5 7

This solution does not require any sophisticated ideas to generate. The girls can be arbitrarily grouped on Monday. So put 1, 2, and 3 together; then group 4, 5, and 6. This leaves 7, 8, and 9 as the final row. For Tuesday, we must make sure that no girl is with a previous companion. Starting with girl 1, the next available companion is 4. We cannot add either 5 or 6 to the row since girl 4 has already walked with them on Monday. Thus, 1, 4, and 7 become a row. The next available girl is 2. She is matched with 5 and then 8. Girl 3 is matched with 6 and 9. For Thursday, we match girl 1 with girl 5. The third girl cannot be 2, 3, 4, or 7 (since 1 has already been grouped with them), nor can it be 4, 6, 2, or 8 (since 5 has walked with them). That leaves only girl 9. In a similar fashion, 2 and 6 must be grouped with 7. Luckily, the remaining girls (3, 4, and 8) have all been unmatched so far. You should verify that the rows for Friday properly complete the solution.

1.3.5 Finite-State Machines

Researchers in computer science find it helpful to have mathematical models to help study and think about computation and the process of translating a high-level program (perhaps in Java or C) into native machine language. In Section 9.2, we will look at some of the models used. One of the simplest is called a *finite-state machine*. These models can also be used to help programmers think about how to solve a programming task and are used by software engineers to discuss, explore, and communicate critical portions of an algorithm with nonprogrammer clients.

EXAMPLE 1.2 **A Simple Finite-State Machine**

Suppose we want to write a program to tease a small child named Percival. The program will repeatedly ask for a name. If the name entered is "Percival", the program is to print "Percival, it is your bedtime." If any other name is entered, it will respond with "*the name*, you may stay up late tonight."

A finite-state machine uses a discrete set of *states* to reflect one of a finite number of possibilities. In this case, there are three states: "Percival", "any other name", and "waiting for a name". The finite-state diagram in Figure 1.6 uses bubbles to represent states and labeled arrows to represent the transition between states. The arrows are labeled with an input and an output: (*input, output*).

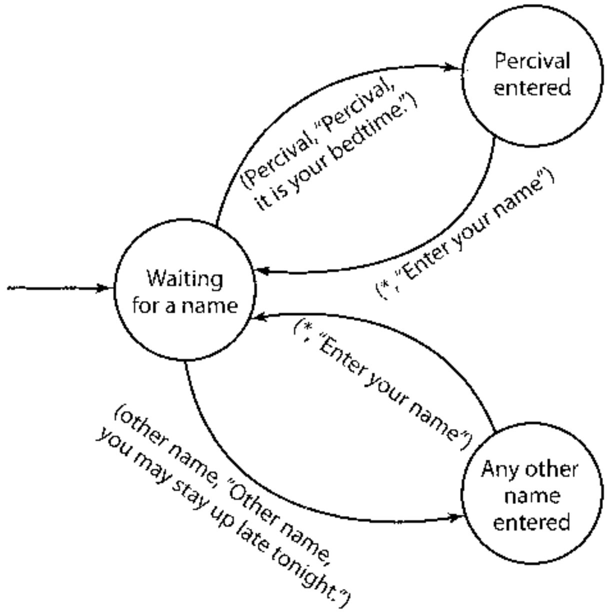


Figure 1.6 A finite-state diagram for the Percival program.

In the finite-state diagram, an input will move from the "waiting for a name" state to either the "Percival" state or the "any other name" state. In each case, an appropriate output is sent to the computer screen. From either state, any input (including no input for at least 30 seconds) causes a new prompt to be printed and moves back to the "waiting for a name" state. On the diagram, the symbol * has been used to indicate "any input". ■

The previous example is trivial, but it does introduce the notion of a finite-state machine without the need for much explaining. The diagram is fairly clear with little need for formal definitions (although formal definitions *will* be presented later). We will be able to model more complicated situations using the simple tool introduced here.

Many other applications to computer science will be discussed. Some examples included (but are not limited to): the analysis of the complexity of algorithms, pattern matching, pseudorandom number generators, encryption, XML, relational databases, and circuit minimization.

Discrete mathematics is a key component in the standard computer science curriculum. In fact, the GRE subject exam in computer science contains more questions from the discrete mathematics course than from almost any other single course in a typical computer science major.

A course in discrete mathematics is also an important component in a typical mathematics major. It is usually where students encounter their first careful examination of logic, axiomatic mathematics, and proof techniques. A more complete look at elementary set theory, elementary number theory, and counting are also presented. Additional topics of interest include (but again are not limited to) Boolean algebras, finite probability theory, recurrence relations, combinatorics, generating functions, and graph theory.

1.3.6 The Set of Rational Numbers Is Countably Infinite

As a final preview example, a proof will be given that the set of rational numbers is countably infinite. This example will illustrate the use of definitions in proofs and also may help you to understand more clearly this counterintuitive result.

The definition of countably infinite (repeated below) is inspired by the buddy system that is often used when swimming at a summer camp. Each swimmer is paired with a partner. If the lifeguard blows a whistle, the partners must find each other and raise their clasped hands. This allows the lifeguard to ensure that no swimmer is without a partner (and so no single swimmer is missing).

In the same way, the definition ensures that every element in the countably infinite set is paired with exactly one integer. This correspondence indicates that the two sets should be considered to be of the same size.

DEFINITION 1.2 ***Countably Infinite***

A set is called *countably infinite* if its elements can be placed in one-to-one correspondence with the positive integers. That is, every element in the set can be labeled by exactly one positive integer and every positive integer is the label for exactly one element of the set.

THEOREM 1.8

The set of rational numbers is countably infinite.

Proof: The proof will first demonstrate that the set of positive rationals is countably infinite.

An initial attempt to list the positive rationals in an orderly fashion is seen in Figure 1.7.

$1/1$	$2/1$	$3/1$	$4/1$	$5/1$	$\cdots$
$1/2$	$2/2$	$3/2$	$4/2$	$5/2$	$\cdots$
$1/3$	$2/3$	$3/3$	$4/3$	$5/3$	$\cdots$
$1/4$	$2/4$	$3/4$	$4/4$	$5/4$	$\cdots$
$1/5$	$2/5$	$3/5$	$4/5$	$5/5$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Figure 1.7 A first attempt to list the positive rationals.

The problem with this attempt is that each rational number will appear multiple times. For example, one half will be listed as $\frac{1}{2}, \frac{2}{4}, \frac{3}{6}, \ldots$, which can be corrected by traversing the listing diagonally as shown in Figure 1.8. Whenever a duplicate is encountered, it can be crossed out.

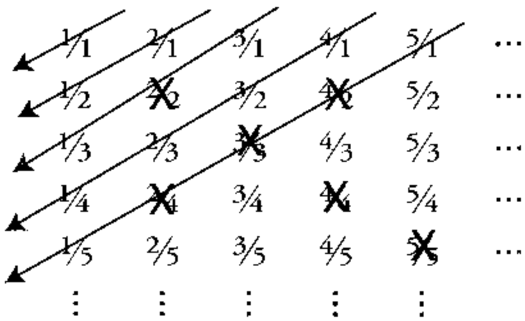


Figure 1.8 Listing the positive rationals.

TABLE 1.6

$\mathbb{Z}^+$	$\mathbb{Q}^+$
1	$\frac{1}{1}$
2	$\frac{2}{1}$
3	$\frac{1}{2}$
4	$\frac{3}{1}$
5	$\frac{1}{3}$
6	$\frac{4}{1}$
7	$\frac{3}{2}$
8	$\frac{2}{3}$
9	$\frac{1}{4}$
⋮	⋮

Now start traversing the diagonals in order, assigning the next available positive integer to each remaining entry. As can be seen in Figure 1.9, each integer will eventually be assigned a positive rational number and each positive rational number will have a unique positive integer attached.

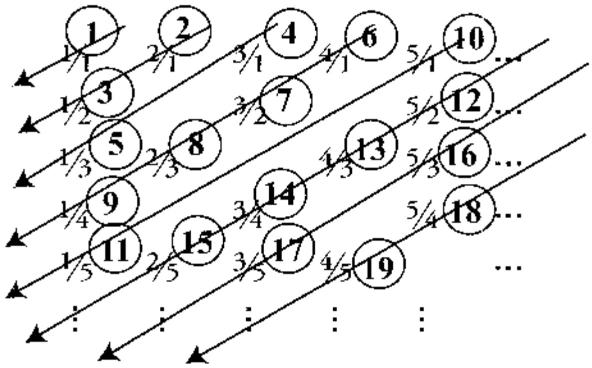


Figure 1.9 Pairing positive integers and positive rationals.

TABLE 1.7

$\mathbb{Z}^+$	$\mathbb{Q}$
1	$\frac{0}{1}$
2	$\frac{1}{1}$
3	$\frac{2}{1}$
4	$-\frac{2}{1}$
5	$\frac{1}{2}$
6	$-\frac{1}{2}$
7	$\frac{3}{1}$
8	$-\frac{3}{1}$
⋮	⋮

Thus, the set of positive rational numbers is countably infinite. The pairing begins as in Table 1.6.

We can form the one-to-one correspondence between the set of positive integers and the set of all rational numbers by adjusting the right-hand column. Insert the number $0 = \frac{0}{1}$ next to 1 and push all other rational numbers down one position. Then, after each positive rational, insert its negation, again pushing all other numbers down one position. The result looks like Table 1.7.

This pairing completes the proof. ∴

1.4 Exercises

The exercises marked with ★ have detailed solutions in Appendix H. Appendix G contains suggestions to help you submit properly written solutions to homework exercises.

1. Consider the preferences for the following group of four women (C, D, E, F) and four men (L, M, N, O).

Female Preferences

	1	2	3	4
C	O	L	M	N
D	O	M	L	N
E	N	M	O	L
F	O	N	M	L

Male Preferences

	1	2	3	4
L	C	D	E	F
M	D	F	E	C
N	C	E	D	F
O	D	C	E	F

Decide whether the following assignment is stable or unstable. Provide some justification for your answer.

C	⟷	O
D	⟷	M
E	⟷	N
F	⟷	L

2. Find a stable marriage assignment for the following group of four women (A, B C, D) and four men (W, X, Y, Z).

Female Preferences

	1	2	3	4
A	Z	X	Y	W
B	X	Z	W	Y
C	W	Y	Z	X
D	Y	W	X	Z

Male Preferences

	1	2	3	4
W	D	A	B	C
X	B	D	C	A
Y	A	C	D	B
Z	A	C	D	B

3. Suppose that there is a group of four men and four women who are eligible for marriage. Create a set of preferences such that there exists exactly one stable assignment among all the possible assignments that could be made. (Assume that there cannot be ties in the preference ratings.)

4. ★ Suppose that there is a group of two women and only one man who are eligible for marriage. Suppose that the man gets paired up with the woman who rated second on his preference chart. Could such an assignment be stable? Provide adequate justification for your answer. (Assume that there cannot be ties in the preference ratings.)

5. Suppose that there are n young men and m young women (with $n \neq m$) who are eligible for marriage. How can the Deferred Acceptance Algorithm be modified to produce a stable assignment with these conditions? Can you prove that the revised algorithm terminates and always produces a stable assignment?

6. The notion of stable assignments has relevance in other settings. One such setting is the stable roommate problem. In this problem, there are n students who need to be assigned roommates (i.e., two students per room). Each student has ranked the other students in preference order with no ties (that is, a student cannot assign the same preference rank to two potential roommates). Is there an algorithm that will produce a stable assignment of roommates?

To see that no such algorithm is possible, it is sufficient to produce an example where such an assignment is impossible. Your task is to create such an example. [*Hint:* Let $n = 4$ and create a set of preferences so that every possible assignment is unstable.]

7. I plan to make some oatmeal cookies, but I am considering three possible modifications to the recipe: I can cut the sugar in half, I could add some walnuts, or I can add some raisins. I can make all or just some of the changes (or I could decide to make no changes to the recipe). How many different kinds of oatmeal cookies can I make?

8. Provide a different solution for the 9-schoolgirl walking problem in Section 1.3.4. Leave the rows for Monday unchanged. Your solution should not be merely a rearrangement of the days. This means that on at least 1 day of the week, at least one row should contain 3 girls who are never together (as a group of 3) in the previous solution.

9. For each claim, determine whether it is always true or else false in some cases. Then give some justification for your answer. Read carefully.

(a) There is no place for continuous mathematical objects in a discrete mathematics course.

(b) Although not usually the case, some topics in mathematics can be studied (at least in part) without using any equations or functions.

(c) The Deferred Acceptance Algorithm, also called the Stable Marriage Problem, has been used in matching medical students with cooperating hospitals.

(d) ★ John proposed to Mary, but she rejected him in favor of Fred. With the Deferred Acceptance Algorithm, Mary is no longer viable for John.

10. For each claim, determine whether it is always true or else false in some cases. Then give some justification for your answer. Read carefully.

(a) An assignment (in the context of the Stable Marriage Problem) is unstable if there exists a man, m_i, and a woman, w_j, who are not partners but m_i prefers w_j to his assigned bride.

(b) The unit square, having corners at the points $(0, 0)$, $(0, 1)$, $(1, 1)$, and $(1, 0)$ in the plane, is no different in size than the open unit interval $(0, 1)$ on the real number line.

(c) ★ A stable assignment could exist in which each member of one of the groups is paired with the person that he or she ranked lowest in the opposite group. (Assume that there are equal numbers of people in both of the groups and that there are no ties in the preference ratings.)

(d) Suppose that there are two groups of people, and that each person is paired with the individual that he or she ranked lowest in the opposite group. Then this assignment could be stable. (Assume that there are equal numbers of people in both of the groups and that there are no ties in the preference ratings.)

11. Classify each of the following sets/objects as discrete or continuous mathematical objects.

(a) ★ The set of integers greater than 1000.

(b) ★ The set of y-coordinates on the graph of

$$f(x) = 4x^2 + 2x + 5$$

(c) The collection of fans at the baseball game.

(d) The set of angles (in radians, relative to 12 o'clock) attained by the second hand of an analog clock.

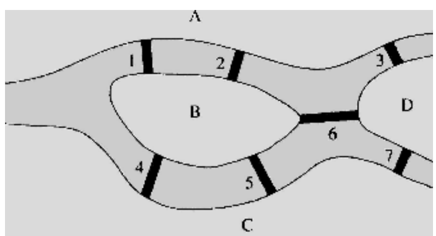
(e) The set of real numbers greater than 0 but less than 50.

12. Show that the set of integers, $\mathbb{Z}$, and the set of even integers (a proper subset of $\mathbb{Z}$), are the same size. Your proof should exhibit a function that matches every element of the integers with exactly one even integer, and every even integer with exactly one integer.

13. ★ Suppose that Percival (Example 1.2) notices the inequity of the program. He wants everyone to be given the same message. Modify the finite-state machine to accommodate this change.

14. Suppose that we wish to design a program that repeatedly asks for the professions of people. Any teacher should be told that he or she will receive an apple, while an individual in any other profession should be told that he or she will receive a new car. Thus, if the profession "teacher" is entered, the program will respond with "you will receive an apple". If any other profession is entered, it will respond with "you will receive a new car". Create a finite-state diagram to accommodate this new idea for a computer program.

15. Looking at the town of Königsberg, find 10 different tours that cross any six of the seven bridges exactly once per bridge. List the starting and ending landmass in each case. The bridges have been labeled for the purpose of specifying which bridge is being visited. For this problem, consider a tour and its reverse ordering to be two representations of the same tour.



16. Prove that the following sets are countably infinite.

(a) The set of positive even integers.

(b) The set of integers.

17. Locate a definition of *discrete mathematics* in some source other than this book. Write the definition and clearly cite the source.

1.5 CHAPTER REVIEW

1.5.1 Summary

This chapter exists mainly to provide an overview for the rest of the course. Most of the examples in this chapter will be presented in greater detail in future chapters. You may wait until that time to worry about the details.

The exception is the discussion about the Stable Marriage Problem and the Deferred Acceptance Algorithm. That material will be supplemented elsewhere in the book, but the initial discussion presented here will not be repeated.

1.5.2 Notation

The following notation has been introduced in this chapter.

Notation	Page	Brief Description
[39]	5	A bibliographic reference