Chapter 1

Counting

1.1 Basic Counting

The Sum Principle

We begin with an example that illustrates a fundamental principle.

Exercise 1.1-1 The loop below is part of an implementation of selection sort, which sorts a list of items chosen from an ordered set (numbers, alphabet characters, words, etc.) into non-decreasing order.

```
(1)  for i = 1 to n − 1
(2)       for j = i + 1 to n
(3)            if (A[i] > A[j])
(4)                 exchange A[i] and A[j]
```

How many times is the comparison $A[i] > A[j]$ made in Line 3?

In Exercise 1.1-1, the segment of code from lines 2 through 4 is executed $n - 1$ times, once for each value of i between 1 and $n - 1$ inclusive. The first time, it makes $n - 1$ comparisons. The second time, it makes $n - 2$ comparisons. The ith time, it makes $n - i$ comparisons. Thus the total number of comparisons is

$$(n-1) + (n-2) + \cdots + 1 \; . \tag{1.1}$$

This formula is not as important as the reasoning that lead us to it. In order to put the reasoning into a broadly applicable format, we will describe what we were doing in the language of sets. Think about the set S containing all comparisons the algorithm in Exercise 1.1-1 makes. We divided set S into $n-1$ pieces (i.e. smaller sets), the set S_1 of comparisons made when $i = 1$, the set S_2 of comparisons made when $i = 2$, and so on through the set S_{n-1} of comparisons made when $i = n - 1$. We were able to figure out the number of comparisons in each of these pieces by observation, and added together the sizes of all the pieces in order to get the size of the set of all comparisons.

in order to describe a general version of the process we used, we introduce some set-theoretic terminology. Two sets are called *disjoint* when they have no elements in common. Each of the sets S_i we described above is disjoint from each of the others, because the comparisons we make for one value of i are different from those we make with another value of i. We say the set of sets $\{S_1, \ldots, S_m\}$ (above, m was $n-1$) is a family of *mutually disjoint sets*, meaning that it is a family (set) of sets, any two of which are disjoint. With this language, we can state a general principle that explains what we were doing without making any specific reference to the problem we were solving.

Principle 1.1 (Sum Principle) *The size of a union of a family of mutually disjoint finite sets is the sum of the sizes of the sets.*

Thus we were, in effect, using the sum principle to solve Exercise 1.1-1. We can describe the sum principle using an algebraic notation. Let $|S|$ denote the size of the set S. For example, $|\{a, b, c\}| = 3$ and $|\{a, b, a\}| = 2$.[1] Using this notation, we can state the sum principle as: if S_1, S_2, $\ldots S_m$ are disjoint sets, then

$$|S_1 \cup S_2 \cup \cdots \cup S_m| = |S_1| + |S_2| + \cdots + |S_m| \,. \tag{1.2}$$

To write this without the "dots" that indicate left-out material, we write

$$|\bigcup_{i=1}^{m} S_i| = \sum_{i=1}^{m} |S_i|.$$

When we can write a set S as a union of disjoint sets S_1, S_2, $\ldots$, S_k we say that we have *partitioned* S into the sets S_1, S_2, $\ldots$, S_k, and we say that the sets S_1, S_2, $\ldots$, S_k form a *partition* of S. Thus $\{\{1\}, \{3, 5\}, \{2, 4\}\}$ is a partition of the set $\{1, 2, 3, 4, 5\}$ and the set $\{1, 2, 3, 4, 5\}$ can be partitioned into the sets $\{1\}$, $\{3, 5\}$, $\{2, 4\}$. It is clumsy to say we are partitioning a set into sets, so instead we call the sets S_i into which we partition a set S the *blocks* of the partition. Thus the sets $\{1\}$, $\{3, 5\}$, $\{2, 4\}$ are the blocks of a partition of $\{1, 2, 3, 4, 5\}$. In this language, we can restate the sum principle as follows.

Principle 1.2 (Sum Principle) *If a finite set S has been partitioned into blocks, then the size of S is the sum of the sizes of the blocks.*

Abstraction

The process of figuring out a general principle that explains why a certain computation makes sense is an example of the mathematical process of *abstraction*. We won't try to give a precise definition of abstraction but rather point out examples of the process as we proceed. In a course in set theory, we would further abstract our work and derive the sum principle from the axioms of

[1] It may look strange to have $|\{a, b, a\}| = 2$, but an element either is or is not in a set. It cannot be in a set multiple times. (This situation leads to the idea of multisets that will be introduced later on in this section.) We gave this example to emphasize that the notation $\{a, b, a\}$ means the same thing as $\{a, b\}$. Why would someone even contemplate the notation $\{a, b, a\}$. Suppose we wrote $S = \{x | x \text{ is the first letter of Ann, Bob, or Alice}\}$. Explicitly following this description of S would lead us to first write down $\{a, b, a\}$ and the realize it equals $\{a, b\}$.

set theory. In a course in discrete mathematics, this level of abstraction is unnecessary, so we will simply use the sum principle as the basis of computations when it is convenient to do so. If our goal were only to solve this one exercise, then our abstraction would have been almost a mindless exercise that complicated what was an "obvious" solution to Exercise 1.1-1. However the sum principle will prove to be useful in a wide variety of problems. Thus we observe the value of abstraction—when you can recognize the abstract elements of a problem, then abstraction often helps you solve subsequent problems as well.

Summing Consecutive Integers

Returning to the problem in Exercise 1.1-1, it would be nice to find a simpler form for the sum given in Equation 1.1. We may also write this sum as

$$\sum_{i=1}^{n-1} n-i.$$

Now, if we don't like to deal with summing the values of $(n-i)$, we can observe that the values we are summing are $n-1, n-2, \ldots, 1$, so we may write that

$$\sum_{i=1}^{n-1} n-i = \sum_{i=1}^{n-1} i.$$

A clever trick, usually attributed to Gauss, gives us a shorter formula for this sum.

We write

$$\begin{array}{ccccccccccc} & 1 & + & 2 & + & \cdots & + & n-2 & + & n-1 \\ + & n-1 & + & n-2 & + & \cdots & + & 2 & + & 1 \\ \hline & n & + & n & + & \cdots & + & n & + & n \end{array}$$

The sum below the horizontal line has $n-1$ terms each equal to n, and thus it is $n(n-1)$. It is the sum of the two sums above the line, and since these sums are equal (being identical except for being in reverse order), the sum below the line must be twice either sum above, so either of the sums above must be $n(n-1)/2$. In other words, we may write

$$\sum_{i=1}^{n-1} n-i = \sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}.$$

This lovely trick gives us little or no real mathematical skill; learning how to think about things to discover answers ourselves is much more useful. After we analyze Exercise 1.1-2 and abstract the process we are using there, we will be able to come back to this problem at the end of this section and see a way that we could have discovered this formula for ourselves without any tricks.

The Product Principle

Exercise 1.1-2 The loop below is part of a program which computes the product of two matrices. (You don't need to know what the product of two matrices is to answer this question.)

```
(1)  for i = 1 to r
(2)      for j = 1 to m
(3)          S = 0
(4)          for k = 1 to n
(5)              S = S + A[i, k] * B[k, j]
(6)          C[i, j] = S
```

How many multiplications (expressed in terms of r, m, and n) does this code carry out in line 5?

Exercise 1.1-3 Consider the following longer piece of pseudocode that sorts a list of numbers and then counts "big gaps" in the list (for this problem, a big gap in the list is a place where a number in the list is more than twice the previous number:

```
(1)  for i = 1 to n - 1
(2)      minval = A[i]
(3)      minindex = i
(4)      for j = i to n
(5)          if (A[j] < minval)
(6)              minval = A[j]
(7)              minindex = j
(8)      exchange A[i] and A[minindex]
(9)
(10) for i = 2 to n
(11)     if (A[i] > 2 * A[i - 1])
(12)         bigjump = bigjump +1
```

How many comparisons does the above code make in lines 5 and 11 ?

In Exercise 1.1-2, the program segment in lines 4 through 5, which we call the "inner loop," takes exactly n steps, and thus makes n multiplications, regardless of what the variables i and j are. The program segment in lines 2 through 5 repeats the inner loop exactly m times, regardless of what i is. Thus this program segment makes n multiplications m times, so it makes nm multiplications.

Why did we add in Exercise 1.1-1, but multiply here? We can answer this question using the abstract point of view we adopted in discussing Exercise 1.1-1. Our algorithm performs a certain set of multiplications. For any given i, the set of multiplications performed in lines 2 through 5 can be divided into the set S_1 of multiplications performed when $j = 1$, the set S_2 of multiplications performed when $j = 2$, and, in general, the set S_j of multiplications performed for any given j value. Each set S_j consists of those multiplications the inner loop carries out for a particular value of j, and there are exactly n multiplications in this set. Let T_i be the set of multiplications that our program segment carries out for a certain i value. The set T_i is the union of the sets S_j; restating this as an equation, we get

$$T_i = \bigcup_{j=1}^{m} S_j.$$

Then, by the sum principle, the size of the set T_i is the sum of the sizes of the sets S_j, and a sum of m numbers, each equal to n, is mn. Stated as an equation,

$$|T_i| = |\bigcup_{j=1}^{m} S_j| = \sum_{j=1}^{m} |S_j| = \sum_{j=1}^{m} n = mn. \tag{1.3}$$

Thus we are multiplying because multiplication is repeated addition!

From our solution we can extract a second principle that simply shortcuts the use of the sum principle.

Principle 1.3 (Product Principle) *The size of a union of m disjoint sets, each of size n, is mn.*

We now complete our discussion of Exercise 1.1-2. Lines 2 through 5 are executed once for each value of i from 1 to r. Each time those lines are executed, they are executed with a different i value, so the set of multiplications in one execution is disjoint from the set of multiplications in any other execution. Thus the set of all multiplications our program carries out is a union of r disjoint sets T_i of mn multiplications each. Then by the product principle, the set of all multiplications has size rmn, so our program carries out rmn multiplications.

Exercise 1.1-3 demonstrates that thinking about whether the sum or product principle is appropriate for a problem can help to decompose the problem into easily-solvable pieces. If you can decompose the problem into smaller pieces and solve the smaller pieces, then you either add or multiply solutions to solve the larger problem. In this exercise, it is clear that the number of comparisons in the program fragment is the sum of the number of comparisons in the first loop in lines 1 through 8 with the number of comparisons in the second loop in lines 10 through 12 (what two disjoint sets are we talking about here?). Further, the first loop makes $n(n+1)/2 - 1$ comparisons[2], and that the second loop has $n-1$ comparisons, so the fragment makes $n(n+1)/2 - 1 + n - 1 = n(n+1)/2 + n - 2$ comparisons.

Two element subsets

Often, there are several ways to solve a problem. We originally solved Exercise 1.1-1 by using the sum principal, but it is also possible to solve it using the product principal. Solving a problem two ways not only increases our confidence that we have found the correct solution, but it also allows us to make new connections and can yield valuable insight.

Consider the set of comparisons made by the entire execution of the code in this exercise. When $i = 1$, j takes on every value from 2 to n. When $i = 2$, j takes on every value from 3 to n. Thus, for each two numbers i and j, we compare $A[i]$ and $A[j]$ exactly once in our loop. (The order in which we compare them depends on whether i or j is smaller.) Thus the number of comparisons we make is the same as the number of two element subsets of the set $\{1, 2, \ldots, n\}$[3]. In how many ways can we choose two elements from this set? If we choose a first and second element, there are n ways to choose a first element, and for each choice of the first element, there are $n-1$ ways to choose a second element. Thus the set of all such choices is the union of n sets

[2] To see why this is true, ask yourself first where the $n(n+1)/2$ comes from, and then why we subtracted one.

[3] The relationship between the set of comparisons and the set of two-element subsets of $\{1, 2, \ldots, n\}$ is an example of a bijection, an idea which will be examined more in Section 1.2.

of size $n-1$, one set for each first element. Thus it might appear that, by the product principle, there are $n(n-1)$ ways to choose two elements from our set. However, what we have chosen is an *ordered pair*, namely a pair of elements in which one comes first and the other comes second. For example, we could choose 2 first and 5 second to get the ordered pair $(2,5)$, or we could choose 5 first and 2 second to get the ordered pair $(5,2)$. Since each pair of distinct elements of $\{1,2,\ldots,n\}$ can be ordered in two ways, we get twice as many ordered pairs as two element sets. Thus, since the number of ordered pairs is $n(n-1)$, the number of two element subsets of $\{1,2,\ldots,n\}$ is $n(n-1)/2$. Therefore the answer to Exercise 1.1-1 is $n(n-1)/2$. This number comes up so often that it has its own name and notation. We call this number "n choose 2" and denote it by $\binom{n}{2}$. To summarize, $\binom{n}{2}$ stands for the number of two element subsets of an n element set and equals $n(n-1)/2$. Since one answer to Exercise 1.1-1 is $1+2+\cdots+n-1$ and a second answer to Exercise 1.1-1 is $\binom{n}{2}$, this shows that

$$1+2+\cdots+n-1=\binom{n}{2}=\frac{n(n-1)}{2}.$$

Important Concepts, Formulas, and Theorems

1. *Set.* A *set* is a collection of objects. In a set order is not important. Thus the set $\{A,B,C\}$ is the same as the set $\{A,C,B\}$. An element either is or is not in a set; it cannot be in a set more than once, even if we have a description of a set which names that element more than once.

2. *Disjoint.* Two sets are called *disjoint* when they have no elements in common.

3. *Mutually disjoint sets.* A set of sets $\{S_1,\ldots,S_n\}$ is a family of *mutually disjoint sets*, if each two of the sets S_i are disjoint.

4. *Size of a set.* Given a set S, the size of S, denoted $|S|$, is the number of distinct elements in S.

5. *Sum Principle.* The size of a union of a family of mutually disjoint sets is the sum of the sizes of the sets. In other words, if S_1, S_2, $\ldots S_n$ are disjoint sets, then
$$|S_1\cup S_2\cup\cdots\cup S_n|=|S_1|+|S_2|+\cdots+|S_n|.$$
To write this without the "dots" that indicate left-out material, we write
$$|\bigcup_{i=1}^{n} S_i|=\sum_{i=1}^{n}|S_i|.$$

6. *Partition of a set.* A partition of a set S is a set of mutually disjoint subsets (sometimes called blocks) of S whose union is S.

7. *Sum of first $n-1$ numbers.*
$$\sum_{i=1}^{n} n-i=\sum_{i=1}^{n-1} i=\frac{n(n-1)}{2}.$$

8. *Product Principle.* The size of a union of m disjoint sets, each of size n, is mn.

9. *Two element subsets.* $\binom{n}{2}$ stands for the number of two element subsets of an n element set and equals $n(n-1)/2$. $\binom{n}{2}$ is read as "n choose 2."

Problems

1. The segment of code below is part of a program that uses insertion sort to sort a list A

```
for i = 2 to n
    j=i
    while j ≥ 2 and A[j] < A[j − 1]
        exchange A[j] and A[j − 1]
        j − −
```

   What is the maximum number of times (considering all lists of n items you could be asked to sort) the program makes the comparison $A[j] < A[j-1]$? Describe as succinctly as you can those lists that require this number of comparisons.

2. Five schools are going to send their baseball teams to a tournament, in which each team must play each other team exactly once. How many games are required?

3. Use notation similar to that in Equations 1.2 and 1.3 to rewrite the solution to Exercise 1.1-3 more algebraically.

4. In how many ways can you draw a first card and then a second card from a deck of 52 cards?

5. In how many ways can you draw two cards from a deck of 52 cards.

6. In how many ways may you draw a first, second, and third card from a deck of 52 cards?

7. In how many ways may a ten person club select a president and a secretary-treasurer from among its members?

8. In how many ways may a ten person club select a two person executive committee from among its members?

9. In how many ways may a ten person club select a president and a two person executive advisory board from among its members (assuming that the president is not on the advisory board)?

10. By using the formula for $\binom{n}{2}$ is is straightforward to show that

$$n\binom{n-1}{2} = \binom{n}{2}(n-2).$$

    However this proof just uses blind substitution and simplification. Find a more conceptual explanation of why this formula is true.

11. If M is an m element set and N is an n-element set, how many ordered pairs are there whose first member is in M and whose second member is in N?

12. In the local ice cream shop, there are 10 different flavors. How many different two-scoop cones are there? (Following your mother's rule that it all goes to the same stomach, a cone with a vanilla scoop on top of a chocolate scoop is considered the same as a cone with a a chocolate scoop on top of a vanilla scoop.)

13. Now suppose that you decide to disagree with your mother in Exercise 12 and say that the order of the scoops does matter. How many different possible two-scoop cones are there?

14. Suppose that on day 1 you receive 1 penny, and, for $i > 1$, on day i you receive twice as many pennies as you did on day $i - 1$. How many pennies will you have on day 20? How many will you have on day n? Did you use the sum or product principal?

15. The "Pile High Deli" offers a "simple sandwich" consisting of your choice of one of five different kinds of bread with your choice of butter or mayonnaise or no spread, one of three different kinds of meat, and one of three different kinds of cheese, with the meat and cheese "piled high" on the bread. In how many ways may you choose a simple sandwich?

16. Do you see any unnecessary steps in the pseudocode of Exercise 1.1-3?

1.2 Counting Lists, Permutations, and Subsets.

Using the Sum and Product Principles

Exercise 1.2-1 A password for a certain computer system is supposed to be between 4 and 8 characters long and composed of lower and/or upper case letters. How many passwords are possible? What counting principles did you use? Estimate the percentage of the possible passwords that have exactly four characters.

A good way to attack a counting problem is to ask if we could use either the sum principle or the product principle to simplify or completely solve it. Here that question might lead us to think about the fact that a password can have 4, 5, 6, 7 or 8 characters. The set of all passwords is the union of those with 4, 5, 6, 7, and 8 letters so the sum principle might help us. To write the problem algebraically, let P_i be the set of i-letter passwords and P be the set of all possible passwords. Clearly,

$$P = P_4 \cup P_5 \cup P_6 \cup P_7 \cup P_8 \ .$$

The P_i are mutually disjoint, and thus we can apply the sum principal to obtain

$$|P| = \sum_{i=4}^{8} |P_i| \ .$$

We still need to compute $|P_i|$. For an i-letter password, there are 52 choices for the first letter, 52 choices for the second and so on. Thus by the product principle, $|P_i|$, the number of passwords with i letters is 52^i. Therefore the total number of passwords is

$$52^4 + 52^5 + 52^6 + 52^7 + 52^8.$$

Of these, 52^4 have four letters, so the percentage with 54 letters is

$$\frac{100 \cdot 52^4}{52^4 + 52^5 + 52^6 + 52^7 + 52^8.}$$

Although this is a nasty formula to evaluate by hand, we can get a quite good estimate as follows. Notice that 52^8 is 52 times as big as 52^7, and even more dramatically larger than any other term in the sum in the denominator. Thus the ratio thus just a bit less than

$$\frac{100 \cdot 52^4}{52^8,}$$

which is $100/52^4$, or approximately .000014. Thus to five decimal places, only .00001% of the passwords have four letters. It is therefore much easier guess a password that we know has four letters than it is to guess one that has between 4 and 8 letters—roughly 7 million times easier!

In our solution to Exercise 1.2-1, we casually referred to the use of the product principle in computing the number of passwords with i letters. We didn't write any set as a union of sets of equal size. We could have, but it would have been clumsy and repetitive. For this reason we will state a second version of the product principle that we can derive from the version for unions of sets by using the idea of mathematical induction that we study in Chapter 4.

Version 2 of the *product principle* states:

Principle 1.4 (Product Principle, Version 2) *If a set S of lists of length m has the properties that*

1. *There are i_1 different first elements of lists in S, and*
2. *For each $j > 1$ and each choice of the first $j-1$ elements of a list in S there are i_j choices of elements in position j of those lists,*

then there are $i_1 i_2 \cdots i_m = \prod_{k=1}^{m} i_k$ lists in S.

Let's apply this version of the product principle to compute the number of m-letter passwords. Since an m-letter password is just a list of m letters, and since there are 52 different first elements of the password and 52 choices for each other position of the password, we have that $i_1 = 52$, $i_2 = 52, \ldots, i_m = 52$. Thus, this version of the product principle tells us immediately that the number of passwords of length m is $i_1 i_2 \cdots i_m = 52^m$.

In our statement of version 2 of the Product Principle, we have introduced a new notation, the use of Π to stand for product. This notation is called the *product notation*, and it is used just like summation notation. In particular, $\prod_{k=1}^{m} i_k$ is read as "The product from $k = 1$ to m of i_k." Thus $\prod_{k=1}^{m} i_k$ means the same thing as $i_1 \cdot i_2 \cdots i_m$.

Lists and functions

We have left a term undefined in our discussion of version 2 of the product principle, namely the word "list." A *list* of 3 things chosen from a set T consists of a first member t_1 of T, a second member t_2 of T, and a third member t_3 of T. If we rewrite the list in a different order, we get a different list. A list of k things chosen from T consists of a first member of T through a kth member of T. We can use the word "function," which you probably recall from algebra or calculus, to be more precise.

Recall that a function from a set S (called the *domain* of the function) to a set T (called the *range* of the function) is a relationship between the elements of S and the elements of T that relates exactly one element of T to each element of S. We use a letter like f to stand for a function and use $f(x)$ to stand for the one and only one element of T that the function relates to the element x of S. You are probably used to thinking of functions in terms of formulas like $f(x) = x^2$. We need to use formulas like this in algebra and calculus because the functions that you study in algebra and calculus have infinite sets of numbers as their domains and ranges. In discrete mathematics, functions often have finite sets as their domains and ranges, and so it is possible to describe a function by saying exactly what it is. For example

$$f(1) = \text{Sam}, \; f(2) = \text{Mary}, \; f(3) = \text{Sarah}$$

is a function that describes a list of three people. This suggests a precise definition of a list of k elements from a set T: A *list of k elements* from a set T is a function from $\{1, 2, \ldots, k\}$ to T.

Exercise 1.2-2 Write down all the functions from the two-element set $\{1, 2\}$ to the two-element set $\{a, b\}$.

Exercise 1.2-3 How many functions are there from a two-element set to a three element set?

Exercise 1.2-4 How many functions are there from a three-element set to a two-element set?

In Exercise 1.2-2 one thing that is difficult is to choose a notation for writing the functions down. We will use f_1, f_2, etc., to stand for the various functions we find. To describe a function f_i from $\{1,2\}$ to $\{a,b\}$ we have to specify $f_i(1)$ and $f_i(2)$. We can write

$$\begin{array}{ll} f_1(1) = a & f_1(2) = b \\ f_2(1) = b & f_2(2) = a \\ f_3(1) = a & f_3(2) = a \\ f_4(1) = b & f_4(2) = b \end{array}$$

We have simply written down the functions as they occurred to us. How do we know we have all of them? The set of all functions from $\{1,2\}$ to $\{a,b\}$ is the union of the functions f_i that have $f_i(1) = a$ and those that have $f_i(1) = b$. The set of functions with $f_i(1) = a$ has two elements, one for each choice of $f_i(2)$. Therefore by the product principle the set of all functions from $\{1,2\}$ to $\{a,b\}$ has size $2 \cdot 2 = 4$.

To compute the number of functions from a two element set (say $\{1,2\}$) to a three element set, we can again think of using f_i to stand for a typical function. Then the set of all functions is the union of three sets, one for each choice of $f_i(1)$. Each of these sets has three elements, one for each choice of $f_i(2)$. Thus by the product principle we have $3 \cdot 3 = 9$ functions from a two element set to a three element set.

To compute the number of functions from a three element set (say $\{1,2,3\}$) to a two element set, we observe that the set of functions is a union of four sets, one for each choice of $f_i(1)$ and $f_i(2)$ (as we saw in our solution to Exercise 1.2-2). But each of these sets has two functions in it, one for each choice of $f_i(3)$. Then by the product principle, we have $4 \cdot 2 = 8$ functions from a three element set to a two element set.

A function f is called *one-to-one* or an *injection* if whenever $x \neq y$, $f(x) \neq f(y)$. Notice that the two functions f_1 and f_2 we gave in our solution of Exercise 1.2-2 are one-to-one, but f_3 and f_4 are not.

A function f is called *onto* or a *surjection* if every element y in the range is $f(x)$ for some x in the domain. Notice that the functions f_1 and f_2 in our solution of Exercise 1.2-2 are onto functions but f_3 and f_4 are not.

Exercise 1.2-5 Using two-element sets or three-element sets as domains and ranges, find an example of a one-to-one function that is not onto.

Exercise 1.2-6 Using two-element sets or three-element sets as domains and ranges, find an example of an onto function that is not one-to-one.

Notice that the function given by $f(1) = c$, $f(2) = a$ is an example of a function from $\{1,2\}$ to $\{a,b,c\}$ that is one-to one but not onto.

Also, notice that the function given by $f(1) = a$, $f(2) = b$, $f(3) = a$ is an example of a function from $\{1,2,3\}$ to $\{a,b\}$ that is onto but not one to one.

The Bijection Principle

Exercise 1.2-7 The loop below is part of a program to determine the number of triangles formed by n points in the plane.

```
(1)  trianglecount = 0
(2)  for i = 1 to n
(3)       for j = i + 1 to n
(4)            for k = j + 1 to n
(5)                 if points i, j, and k are not collinear
(6)                      trianglecount = trianglecount +1
```

How many times does the above code check three points to see if they are collinear in line 5?

In Exercise 1.2-7, we have a loop embedded in a loop that is embedded in another loop. Because the second loop, starting in line 3, begins with $j = i + 1$ and j increase up to n, and because the third loop, starting in line 4, begins with $k = j + 1$ and increases up to n, our code examines each triple of values i, j, k with $i < j < k$ exactly once. For example, if n is 4, then the triples (i, j, k) used by the algorithm, in order, are $(1, 2, 3)$, $(1, 2, 4)$, $(1, 3, 4)$, and $(2, 3, 4)$. Thus one way in which we might have solved Exercise 1.2-7 would be to compute the number of such triples, which we will call *increasing triples*. As with the case of two-element subsets earlier, the number of such triples is the number of three-element subsets of an n-element set. This is the second time that we have proposed counting the elements of one set (in this case the set of increasing triples chosen from an n-element set) by saying that it is equal to the number of elements of some other set (in this case the set of three element subsets of an n-element set). When are we justified in making such an assertion that two sets have the same size? There is another fundamental principle that abstracts our concept of what it means for two sets to have the same size. Intuitively two sets have the same size if we can match up their elements in such a way that each element of one set corresponds to exactly one element of the other set. This description carries with it some of the same words that appeared in the definitions of functions, one-to-one, and onto. Thus it should be no surprise that one-to-one and onto functions are part of our abstract principle.

Principle 1.5 (Bijection Principle) *Two sets have the same size if and only if there is a one-to-one function from one set onto the other.*

Our principle is called the *bijection principle* because a one-to-one and onto function is called a *bijection*. Another name for a bijection is a *one-to-one correspondence*. A bijection from a set to itself is called a *permutation* of that set.

What is the bijection that is behind our assertion that the number of increasing triples equals the number of three-element subsets? We define the function f to be the one that takes the increasing triple (i, j, k) to the subset $\{i, j, k\}$. Since the three elements of an increasing triple are different, the subset is a three element set, so we have a function from increasing triples to three element sets. Two different triples can't be the same set in two different orders, so different triples have to be associated with different sets. Thus f is one-to-one. Each set of three integers can be listed in increasing order, so it is the image under f of an increasing triple. Therefore f is onto. Thus we have a one-to-one correspondence, or bijection, between the set of increasing triples and the set of three element sets.

k-element permutations of a set

Since counting increasing triples is equivalent to counting three-element subsets, we can count increasing triples by counting three-element subsets instead. We use a method similar to the one we used to compute the number of two-element subsets of a set. Recall that the first step was to compute the number of ordered pairs of distinct elements we could chose from the set $\{1, 2, \ldots, n\}$. So we now ask in how many ways may we choose an ordered triple of distinct elements from $\{1, 2, \ldots, n\}$, or more generally, in how many ways may we choose a list of k distinct elements from $\{1, 2, \ldots, n\}$. A list of k-distinct elements chosen from a set N is called a *k-element permutation* of N.[4]

How many 3-element permutations of $\{1, 2, \ldots, n\}$ can we make? Recall that a k-element permutation is a list of k distinct elements. There are n choices for the first number in the list. For each way of choosing the first element, there are $n-1$ choices for the second. For each choice of the first two elements, there are $n-2$ ways to choose a third (distinct) number, so by version 2 of the product principle, there are $n(n-1)(n-2)$ ways to choose the list of numbers. For example, if n is 4, the three-element permutations of $\{1, 2, 3, 4\}$ are

$$\begin{aligned} L = \{&123, 124, 132, 134, 142, 143, 213, 214, 231, 234, 241, 243, \\ &312, 314, 321, 324, 341, 342, 412, 413, 421, 423, 431, 432\}. \end{aligned} \tag{1.4}$$

There are indeed $4 \cdot 3 \cdot 2 = 24$ lists in this set. Notice that we have listed the lists in the order that they would appear in a dictionary (assuming we treated numbers as we treat letters). This ordering of lists is called the *lexicographic ordering*.

A general pattern is emerging. To compute the number of k-element permutations of the set $\{1, 2, \ldots, n\}$, we recall that they are lists and note that we have n choices for the first element of the list, and regardless of which choice we make, we have $n-1$ choices for the second element of the list, and more generally, given the first $i-1$ elements of a list we have $n-(i-1) = n-i+1$ choices for the ith element of the list. Thus by version 2 of the product principle, we have $n(n-1)\cdots(n-k+1)$ (which is the first k terms of $n!$) ways to choose a k-element permutation of $\{1, 2, \ldots, n\}$. There is a very handy notation for this product first suggested by Don Knuth. We use $n^{\underline{k}}$ to stand for $n(n-1)\cdots(n-k+1) = \prod_{i=0}^{k-1} n-i$, and call it the *$k$th falling factorial power of n*. We can summarize our observations in a theorem.

Theorem 1.1 *The number k-element permutations of an n-element set is*

$$n^{\underline{k}} = \prod_{i=0}^{k-1} n-i = n(n-1)\cdots(n-k+1) = n!/(n-k)! \ .$$

Counting subsets of a set

We now return to the question of counting the number of three element subsets of a $\{1, 2, \ldots, n\}$. We use $\binom{n}{3}$, which we read as "n choose 3" to stand for the number of three element subsets of

[4] In particular a k-element permutation of $\{1, 2, \ldots k\}$ is a list of k distinct elements of $\{1, 2, \ldots, k\}$, which, by our definition of a list is a function from $\{1, 2, \ldots, k\}$ to $\{1, 2, \ldots, k\}$. This function must be one-to-one since the elements of the list are distinct. Since there are k distinct elements of the list, every element of $\{1, 2, \ldots, k\}$ appears in the list, so the function is onto. Therefore it is a bijection. Thus our definition of a permutation of a set is consistent with our definition of a k-element permutation in the case where the set is $\{1, 2, \ldots, k\}$.

$\{1, 2, \ldots, n\}$, or more generally of any n-element set. We have just carried out the first step of computing $\binom{n}{3}$ by counting the number of three-element permutations of $\{1, 2, \ldots, n\}$.

Exercise 1.2-8 Let L be the set of all three-element permutations of $\{1, 2, 3, 4\}$, as in Equation 1.4. How many of the lists (permutations) in L are lists of the 3 element set $\{1, 3, 4\}$? What are these lists?

We see that this set appears in L as 6 different lists: 134, 143, 314, 341, 413, and 431. In general given three different numbers with which to create a list, there are three ways to choose the first number in the list, given the first there are two ways to choose the second, and given the first two there is only one way to choose the third element of the list. Thus by version 2 of the product principle once again, there are $3 \cdot 2 \cdot 1 = 6$ ways to make the list.

Since there are $n(n-1)(n-2)$ permutations of an n-element set, and each three-element subset appears in exactly 6 of these lists, the number of three-element permutations is six times the number of three element subsets. That is, $n(n-1)(n-2) = \binom{n}{3} \cdot 6$. Whenever we see that one number that counts something is the product of two other numbers that count something, we should expect that there is an argument using the product principle that explains why. Thus we should be able to see how to break the set of all 3-element permutations of $\{1, 2, \ldots, n\}$ into either 6 disjoint sets of size $\binom{n}{3}$ or into $\binom{n}{3}$ subsets of size six. Since we argued that each three element subset corresponds to six lists, we have described how to get a set of six lists from one three-element set. Two different subsets could never give us the same lists, so our sets of three-element lists are disjoint. In other words, we have divided the set of all three-element permutations into $\binom{n}{3}$ mutually sets of size six. In this way the product principle does explain why $n(n-1)(n-2) = \binom{n}{3} \cdot 6$. By division we get that we have

$$\binom{n}{3} = n(n-1)(n-2)/6$$

three-element subsets of $\{1, 2, \ldots, n\}$. For $n = 4$, the number is $4(3)(2)/6 = 4$. These sets are $\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{1, 3, 4\}$, and $\{2, 3, 4\}$. It is straightforward to verify that each of these sets appears 6 times in L, as 6 different lists.

Essentially the same argument gives us the number of k-element subsets of $\{1, 2, \ldots, n\}$. We denote this number by $\binom{n}{k}$, and read it as "n choose k." Here is the argument: the set of all k-element permutations of $\{1, 2, \ldots, n\}$ can be partitioned into $\binom{n}{k}$ disjoint blocks[5], each block consisting of all k-element permutations of a k-element subset of $\{1, 2, \ldots, n\}$. But the number of k-element permutations of a k-element set is $k!$, either by version 2 of the product principle or by Theorem 1.1. Thus by version 1 of the product principle we get the equation

$$n^{\underline{k}} = \binom{n}{k} k!.$$

Division by $k!$ gives us our next theorem.

Theorem 1.2 *For integers n and k with $0 \leq k \leq n$, the number of k element subsets of an n element set is*

$$\frac{n^{\underline{k}}}{k!} = \frac{n!}{k!(n-k)!}$$

[5]Here we are using the language introduced for partitions of sets in Section 1.1

Proof: The proof is given above, except in the case that k is 0; however the only subset of our n-element set of size zero is the empty set, so we have exactly one such subset. This is exactly what the formula gives us as well. (Note that the cases $k = 0$ and $k = n$ both use the fact that $0! = 1$.[6]) The equality in the theorem comes from the definition of $n^{\underline{k}}$. ■

Another notation for the numbers $\binom{n}{k}$ is $C(n, k)$. Thus we have that

$$C(n, k) = \binom{n}{k} = \frac{n!}{k!(n-k)!}. \tag{1.5}$$

These numbers are called *binomial coefficients* for reasons that will become clear later.

Important Concepts, Formulas, and Theorems

1. *List.* A list of k items chosen from a set X is a function from $\{1, 2, \ldots k\}$ to X.

2. *Lists versus sets.* In a list, the order in which elements appear in the list matters, and an element may appear more than once. In a set, the order in which we write down the elements of the set does not matter, and an element can appear at most once.

3. *Product Principle, Version 2.* If a set S of lists of length m has the properties that

   (a) There are i_1 different first elements of lists in S, and

   (b) For each $j > 1$ and each choice of the first $j - 1$ elements of a list in S there are i_j choices of elements in position j of those lists,

   then there are $i_1 i_2 \cdots i_m$ lists in S.

4. *Product Notaton.* We use the Greek letter Π to stand for product just as we use the Greek letter Σ to stand for sum. This notation is called the *product notation*, and it is used just like summation notation. In particular, $\prod_{k=1}^{m} i_k$ is read as "The product from $k = 1$ to m of i_k." Thus $\prod_{k=1}^{m} i_k$ means the same thing as $i_1 \cdot i_2 \cdots i_m$.

5. *Function.* A *function* f from a set S to a set T is a relationship between S and T that relates exactly one element of T to each element of S. We write $f(x)$ for the one and only one element of T that the function f relates to the element x of S. The same element of T may be related to different members of S.

6. *Onto, Surjection* A function f from a set S to a set T is *onto* if for each element $y \in T$, there is at least one $x \in S$ such that $f(x) = y$. An onto function is also called a *surjection.*

7. *One-to-one, Injection.* A function f from a set S to a set T is *one-to-one* if, for each $x \in S$ and $y \in S$ with $x \neq y$, $f(x) \neq f(y)$. A one-to-one function is also called an *injection.*

8. *Bijection, One-to-one correspondence.* A function from a set S to a set T is a *bijection* if it is both one-to-one and onto. A bijection is sometimes called a *one-to-one correspondence.*

9. *Permutation.* A one-to-one function from a set S to S is called a *permutation* of S.

[6] There are many reasons why 0! is defined to be one; making the formula for $\binom{n}{k}$ work out is one of them.

10. *k-element permutation.* A *k-element permutation* of a set S is a list of k distinct elements of S.

11. *k-element subsets. n choose k. Binomial Coefficients.* For integers n and k with $0 \leq k \leq n$, the number of k element subsets of an n element set is $n!/k!(n-k)!$. The number of k-element subsets of an n-element set is usually denoted by $\binom{n}{k}$ or $C(n, k)$, both of which are read as "n choose k." These numbers are called *binomial coefficients.*

12. The number of k-element permutations of an n-element set is

$$n^{\underline{k}} = n(n-1)\cdots(n-k+1) = n!/(n-k)!.$$

13. When we have a formula to count something and the formula expresses the result as a product, it is useful to try to understand whether and how we could use the product principle to prove the formula.

Problems

1. The "Pile High Deli" offers a "simple sandwich" consisting of your choice of one of five different kinds of bread with your choice of butter or mayonnaise or no spread, one of three different kinds of meat, and one of three different kinds of cheese, with the meat and cheese "piled high" on the bread. In how many ways may you choose a simple sandwich?

2. In how many ways can we pass out k distinct pieces of fruit to n children (with no restriction on how many pieces of fruit a child may get)?

3. Write down all the functions from the three-element set $\{1, 2, 3\}$ to the set $\{a, b\}$. Indicate which functions, if any, are one-to-one. Indicate which functions, if any, are onto.

4. Write down all the functions form the two element set $\{1, 2\}$ to the three element set $\{a, b, c\}$ Indicate which functions, if any, are one-to-one. Indicate which functions, if any, are onto.

5. There are more functions from the real numbers to the real numbers than most of us can imagine. However in discrete mathematics we often work with functions from a finite set S with s elements to a finite set T with t elements. Then there are only a finite number of functions from S to T. How many functions are there from S to T in this case?

6. Assuming $k \leq n$, in how many ways can we pass out k distinct pieces of fruit to n children if each child may get at most one? What is the number if $k > n$? Assume for both questions that we pass out all the fruit.

7. Assume $k \leq n$, in how many ways can we pass out k identical pieces of fruit to n children if each child may get at most one? What is the number if $k > n$? Assume for both questions that we pass out all the fruit.

8. What is the number of five digit (base ten) numbers? What is the number of five digit numbers that have no two consecutive digits equal? What is the number that have at least one pair of consecutive digits equal?

9. We are making a list of participants in a panel discussion on allowing alcohol on campus. They will be sitting behind a table in the order in which we list them. There will be four administrators and four students. In how many ways may we list them if the administrators must sit together in a group and the students must sit together in a group? In how many ways may we list them if we must alternate students and administrators?

10. (This problem is for students who are working on the relationship between k-element permutations and k-element subsets.) Write down all three element permutations of the five element set $\{1, 2, 3, 4, 5\}$ in lexicographic order. Underline those that correspond to the set $\{1, 3, 5\}$. Draw a rectangle around those that correspond to the set $\{2, 4, 5\}$. How many three-element permutations of $\{1, 2, 3, 4, 5\}$ correspond to a given 3-element set? How many three-element subsets does the set $\{1, 2, 3, 4, 5\}$ have?

11. In how many ways may a class of twenty students choose a group of three students from among themselves to go to the professor and explain that the three-hour labs are actually taking ten hours?

12. We are choosing participants for a panel discussion allowing on allowing alcohol on campus. We have to choose four administrators from a group of ten administrators and four students from a group of twenty students. In how many ways may we do this?

13. We are making a list of participants in a panel discussion on allowing alcohol on campus. They will be sitting behind a table in the order in which we list them. There will be four administrators chosen from a group of ten administrators and four students chosen from a group of twenty students. In how many ways may we choose and list them if the administrators must sit together in a group and the students must sit together in a group? In how many ways may we choose and list them if we must alternate students and administrators?

14. In the local ice cream shop, you may get a sundae with two scoops of ice cream from 10 flavors (in accordance with your mother's rules from Problem 12 in Section 1.1, the way the scoops sit in the dish does not matter), any one of three flavors of topping, and any (or all or none) of whipped cream, nuts and a cherry. How many different sundaes are possible?

15. In the local ice cream shop, you may get a three-way sundae with three of the ten flavors of ice cream, any one of three flavors of topping, and any (or all or none) of whipped cream, nuts and a cherry. How many different sundaes are possible(in accordance with your mother's rules from Problem 12 in Section 1.1, the way the scoops sit in the dish does not matter) ?

16. A tennis club has $2n$ members. We want to pair up the members by twos for singles matches. In how many ways may we pair up all the members of the club? Suppose that in addition to specifying who plays whom, for each pairing we say who serves first. Now in how many ways may we specify our pairs?

17. A basketball team has 12 players. However, only five players play at any given time during a game. In how may ways may the coach choose the five players? To be more realistic, the five players playing a game normally consist of two guards, two forwards, and one center. If there are five guards, four forwards, and three centers on the team, in how many ways can the coach choose two guards, two forwards, and one center? What if one of the centers is equally skilled at playing forward?

18. Explain why a function from an n-element set to an n-element set is one-to-one if and only if it is onto.

19. The function g is called an *inverse* to the function f if the domain of g is the range of f, if $g(f(x)) = x$ for every x in the domain of f and if $f(g(y)) = y$ for each y in the range of f.

    (a) Explain why a function is a bijection if and only if it has an inverse function.

    (b) Explain why a function that has an inverse function has only one inverse function.

1.3 Binomial Coefficients

In this section, we will explore various properties of binomial coefficients. Remember that we defined the quantitu $\binom{m}{k}$ to be the number of k-element subsets of an n-element set.

Pascal's Triangle

Table 1 contains the values of the binomial coefficients $\binom{n}{k}$ for $n = 0$ to 6 and all relevant k values. The table begins with a 1 for $n = 0$ and $k = 0$, because the empty set, the set with no elements, has exactly one 0-element subset, namely itself. We have not put any value into the table for a value of k larger than n, because we haven't directly said what we mean by the binomial coefficient $\binom{n}{k}$ in that case. However, since there are no subsets of an n-element set that have size larger than n, it is natural to say that $\binom{n}{k}$ is zero when $k > n$. Therefore we define $\binom{n}{k}$ to be zero[7] when $k > n$. Thus we could could fill in the empty places in the table with zeros. The table is easier to read if we don't fill in the empty spaces, so we just remember that they are zero.

Table 1.1: A table of binomial coefficients

$n\backslash k$	0	1	2	3	4	5	6
0	1						
1	1	1					
2	1	2	1				
3	1	3	3	1			
4	1	4	6	4	1		
5	1	5	10	10	5	1	
6	1	6	15	20	15	6	1

Exercise 1.3-1 What general properties of binomial coefficients do you see in Table 1.1

Exercise 1.3-2 What is the next row of the table of binomial coefficients?

Several properties of binomial coefficients are apparent in Table 1.1. Each row begins with a 1, because $\binom{n}{0}$ is always 1. This is the case because there is just one subset of an n-element set with 0 elements, namely the empty set. Similarly, each row ends with a 1, because an n-element set S has just one n-element subset, namely S itself. Each row increases at first, and then decreases. Further the second half of each row is the reverse of the first half. The array of numbers called *Pascal's Triangle* emphasizes that symmetry by rearranging the rows of the table so that they line up at their centers. We show this array in Table 2. When we write down Pascal's triangle, we leave out the values of n and k.

You may know a method for creating Pascal's triangle that does not involve computing binomial coefficients, but rather creates each row from the row above. Each entry in Table 1.2, except for the ones, is the sum of the entry directly above it to the left and the entry directly

[7] If you are thinking "But we did define $\binom{n}{k}$ to be zero when $k > n$ by saying that it is the number of k element subsets of an n-element set, so of course it is zero," then good for you.

Table 1.2: Pascal's Triangle

						1						
					1		1					
				1		2		1				
			1		3		3		1			
		1		4		6		4		1		
	1		5		10		10		5		1	
1		6		15		20		15		6		1

above it to the right. We call this the *Pascal Relationship*, and it gives another way to compute binomial coefficients without doing the multiplying and dividing in Equation 1.5. If we wish to compute many binomial coefficients, the Pascal relationship often yields a more efficient way to do so. Once the coefficients in a row have been computed, the coefficients in the next row can be computed using only one addition per entry.

We now verify that the two methods for computing Pascal's triangle always yield the same result. In order to do so, we need an algebraic statement of the Pascal Relationship. In Table 1.1, each entry is the sum of the one above it and the one above it and to the left. In algebraic terms, then, the Pascal Relationship says

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}, \tag{1.6}$$

whenever $n > 0$ and $0 < k < n$. It is possible to give a purely algebraic (and rather dreary) proof of this formula by plugging in our earlier formula for binomial coefficients into all three terms and verifying that we get an equality. A guiding principle of discrete mathematics is that when we have a formula that relates the numbers of elements of several sets, we should find an explanation that involves a relationship among the sets.

A proof using the Sum Principle

From Theorem 1.2 and Equation 1.5, we know that the expression $\binom{n}{k}$ is the number of k-element subsets of an n element set. Each of the three terms in Equation 1.6 therefore represents the number of subsets of a particular size chosen from an appropriately sized set. In particular, the three terms are the number of k-element subsets of an n-element set, the number of $(k-1)$-element subsets of an $(n-1)$-element set, and the number of k-element subsets of an $(n-1)$-element set. We should, therefore, be able to explain the relationship among these three quantities using the sum principle. This explanation will provide a proof, just as valid a proof as an algebraic derivation. Often, a proof using the sum principle will be less tedious, and will yield more insight into the problem at hand.

Before giving such a proof in Theorem 1.3 below, we work out a special case. Suppose $n = 5$, $k = 2$. Equation 1.6 says that

$$\binom{5}{2} = \binom{4}{1} + \binom{4}{2}. \tag{1.7}$$

Because the numbers are small, it is simple to verify this by using the formula for binomial coefficients, but let us instead consider subsets of a 5-element set. Equation 1.7 says that the number of 2 element subsets of a 5 element set is equal to the number of 1 element subsets of a 4 element set plus the number of 2 element subsets of a 4 element set. But to apply the sum principle, we would need to say something stronger. To apply the sum principle, we should be able to partition the set of 2 element subsets of a 5 element set into 2 disjoint sets, one of which has the same size as the number of 1 element subsets of a 4 element set and one of which has the same size as the number of 2 element subsets of a 4 element set. Such a partition provides a proof of Equation 1.7. Consider now the set $S = \{A, B, C, D, E\}$. The set of two element subsets is

$$S_1 = \{\{A, B\}, \{AC\}, \{A, D\}, \{A, E\}, \{B, C\}, \{B, D\}, \{B, E\}, \{C, D\}, \{C, E\}, \{D, E\}\}.$$

We now partition S_1 into 2 blocks, S_2 and S_3. S_2 contains all sets in S_1 that do contain the element E, while S_3 contains all sets in S_1 that do not contain the element E. Thus,

$$S_2 = \{\{AE\}, \{BE\}, \{CE\}, \{DE\}\}$$

and

$$S_3 = \{\{AB\}, \{AC\}, \{AD\}, \{BC\}, \{BD\}, \{CD\}\}.$$

Each set in S_2 must contain E and thus contains one other element from S. Since there are 4 other elements in S that we can choose along with E, we have $|S_2| = \binom{4}{1}$. Each set in S_3 contains 2 elements from the set $\{A, B, C, D\}$. There are $\binom{4}{2}$ ways to choose such a two-element subset of $\{A < B < C < D\}$. But $S_1 = S_2 \cup S_3$ and S_2 and S_3 are disjoint, and so, by the sum principle, Equation 1.7 must hold.

We now give a proof for general n and k.

Theorem 1.3 *If n and k are integers with $n > 0$ and $0 < k < n$, then*

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}.$$

Proof: The formula says that the number of k-element subsets of an n-element set is the sum of two numbers. As in our example, we will apply the sum principle. To apply it, we need to represent the set of k-element subsets of an n-element set as a union of two other disjoint sets. Suppose our n-element set is $S = \{x_1, x_2, \ldots x_n\}$. Then we wish to take S_1, say, to be the $\binom{n}{k}$-element set of all k-element subsets of S and partition it into two disjoint sets of k-element subsets, S_2 and S_3, where the sizes of S_2 and S_3 are $\binom{n-1}{k-1}$ and $\binom{n-1}{k}$ respectively. We can do this as follows. Note that $\binom{n-1}{k}$ stands for the number of k element subsets of the first $n-1$ elements $x_1, x_2, \ldots, x_{n-1}$ of S. Thus we can let S_3 be the set of k-element subsets of S that don't contain x_n. Then the only possibility for S_2 is the set of k-element subsets of S that *do* contain x_n. How can we see that the number of elements of this set S_2 is $\binom{n-1}{k-1}$? By observing that removing x_n from each of the elements of S_2 gives a $(k-1)$-element subset of $S' = \{x_1, x_2, \ldots x_{n-1}\}$. Further each $(k-1)$-element subset of S' arises in this way from one and only one k-element subset of S containing x_n. Thus the number of elements of S_2 is the number of $(k-1)$-element subsets of S', which is $\binom{n-1}{k-1}$. Since S_2 and S_3 are two disjoint sets whose union is S, the sum principle shows that the number of elements of S is $\binom{n-1}{k-1} + \binom{n-1}{k}$. ■

Notice that in our proof, we used a bijection that we did not explicitly describe. Namely, there is a bijection f between S_3 (the k-element sets of S that contain x_n) and the $(k-1)$-element subsets of S'. For any subset K in S_3, We let $f(K)$ be the set we obtain by removing x_n from K. It is immediate that this is a bijection, and so the bijection principle tells us that the size of S_3 is the size of the set of all subsets of S'.

The Binomial Theorem

Exercise 1.3-3 What is $(x+y)^3$? What is $(x+1)^4$? What is $(2+y)^4$? What is $(x+y)^4$?

The number of k-element subsets of an n-element set is called a *binomial coefficient* because of the role that these numbers play in the algebraic expansion of a binomial $x+y$. The **Binomial Theorem** states that

Theorem 1.4 (Binomial Theorem) *For any integer $n \geq 0$*

$$(x+y)^n = \binom{n}{0}x^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{n-1}xy^{n-1} + \binom{n}{n}y^n, \tag{1.8}$$

or in summation notation,

$$(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^{n-i}y^i \ .$$

Unfortunately when most people first see this theorem, they do not have the tools to see easily why it is true. Armed with our new methodology of using subsets to prove algebraic identities, we can give a proof of this theorem.

Let us begin by considering the example $(x+y)^3$ which by the binomial theorem is

$$\begin{aligned} (x+y)^3 &= \binom{3}{0}x^3 + \binom{3}{1}x^2y + \binom{3}{2}xy^2 + \binom{3}{3}y^3 && (1.9)\\ &= x^3 + 3x^2y + 3xy^2 + x^3 \ . && (1.10) \end{aligned}$$

Suppose that we did not know the binomial theorem but still wanted to compute $(x+y)^3$. Then we would write out $(x+y)(x+y)(x+y)$ and perform the multiplication. Probably we would multiply the first two terms, obtaining $x^2 + 2xy + y^2$, and then multiply this expression by $x+y$. Notice that by applying distributive laws you get

$$(x+y)(x+y) = (x+y)x + (x+y)y = xx + xy + yx + y. \tag{1.11}$$

We could use the commutative law to put this into the usual form, but let us hold off for a moment so we can see a pattern evolve. To compute $(x+y)^3$, we can multiply the expression on the right hand side of Equation 1.11 by $x+y$ using the distributive laws to get

$$\begin{aligned} (xx+xy+yx+yy)(x+y) &= (xx+xy+yx+yy)x + (xx+xy+yx+yy)y && (1.12)\\ &= xxx + xyx + yxx + yxx + xxy + xyy + yxy + yyy && (1.13) \end{aligned}$$

Each of these 8 terms that we got from the distributive law may be thought of as a product of terms, one from the first binomial, one from the second binomial, and one from the third binomial. Multiplication is commutative, so many of these products are the same. In fact, we have one xxx or x^3 product, three products with two x's and one y, or x^2y, three products with one x and two y's, or xy^2 and one product which becomes y^3. Now look at Equation 1.9, which summarizes this process. There are $\binom{3}{0} = 1$ way to choose a product with 3 x's and 0 y's, $\binom{3}{1} = 3$ way to choose a product with 2 x's and 1 y, etc. Thus we can understand the binomial theorem as counting the subsets of our binomial factors from which we choose a y-term to get a product with k y's in multiplying a string of n binomials.

Essentially the same explanation gives us a proof of the binomial theorem. Note that when we multiplied out three factors of $(x + y)$ using the distributive law but not collecting like terms, we had a sum of eight products. Each factor of $(x+y)$ doubles the number of summands. Thus when we apply the distributive law as many times as possible (without applying the commutative law and collecting like terms) to a product of n binomials all equal to (x+y), we get 2^n summands. Each summand is a product of a length n list of x's and y's. In each list, the ith entry comes from the ith binomial factor. A list that becomes $x^{n-k}y^k$ when we use the commutative law will have a y in k of its places and an x in the remaining places. The number of lists that have a y in k places is thus the number of ways to select k binomial factors to contribute a y to our list. But the number of ways to select k binomial factors from n binomial factors is simply $\binom{n}{k}$, and so that is the coefficient of $x^{n-k}y^k$. This proves the binomial theorem.

Applying the Binomial Theorem to the remaining questions in Exercise 1.3-3 gives us

$$\begin{aligned}
(x+1)^4 &= x^4 + 4x^3 + 6x^2 + 4x + 1 \\
(2+y)^4 &= 16 + 32y + 24y^2 + 8y^3 + y^4 \text{ and} \\
(x+y)^4 &= x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4.
\end{aligned}$$

Labeling and trinomial coefficients

Exercise 1.3-4 Suppose that I have k labels of one kind and $n - k$ labels of another. In how many different ways may I apply these labels to n objects?

Exercise 1.3-5 Show that if we have k_1 labels of one kind, k_2 labels of a second kind, and $k_3 = n - k_1 - k_2$ labels of a third kind, then there are $\frac{n!}{k_1!k_2!k_3!}$ ways to apply these labels to n objects.

Exercise 1.3-6 What is the coefficient of $x^{k_1}y^{k_2}z^{k_3}$ in $(x + y + z)^n$?

Exercise 1.3-4 and Exercise 1.3-5 can be thought of as immediate applications of binomial coefficients. For Exercise 1.3-4, there are $\binom{n}{k}$ ways to choose the k objects that get the first label, and the other objects get the second label, so the answer is $\binom{n}{k}$. For Exercise 1.3-5, there are $\binom{n}{k_1}$ ways to choose the k_1 objects that get the first kind of label, and then there are $\binom{n-k_1}{k_2}$ ways to choose the objects that get the second kind of label. After that, the remaining $k_3 = n - k_1 - k_2$ objects get the third kind of label. The total number of labellings is thus, by the product principle, the product of the two binomial coefficients, which simplifies as follows.

$$\binom{n}{k_1}\binom{n-k_1}{k_2} = \frac{n!}{k_1!(n-k_1)!}\frac{(n-k_1)!}{k_2!(n-k_1-k_2)!}$$

$$
\begin{aligned}
&= \frac{n!}{k_1!k_2!(n-k_1-k_2)!} \\
&= \frac{n!}{k_1!k_2!k_3!}\,.
\end{aligned}
$$

A more elegant approach to Exercise 1.3-4, Exercise 1.3-5, and other related problems appears in the next section.

Exercise 1.3-6 shows how Exercise 1.3-5 applies to computing powers of trinomials. In expanding $(x+y+z)^n$, we think of writing down n copies of the trinomial $x+y+z$ side by side, and applying the distributive laws until we have a sum of terms each of which is a product of x's, y's and z's. How many such terms do we have with k_1 x's, k_2 y's and k_3 z's? Imagine choosing x from some number k_1 of the copies of the trinomial, choosing y from some number k_2, and z from the remaining k_3 copies, multiplying all the chosen terms together, and adding up over all ways of picking the k_is and making our choices. Choosing x from a copy of the trinomial "labels" that copy with x, and the same for y and z, so the number of choices that yield $x^{k_1}y^{k_2}z^{k_3}$ is the number of ways to label n objects with k_1 labels of one kind, k_2 labels of a second kind, and k_3 labels of a third. Notice that this requires that $k_3 = n - k_1 - k_2$. By analogy with our notation for a binomial coefficient, we define the *trinomial coefficient* $\binom{n}{k_1,k_2,k_3}$ to be $\frac{n!}{k_1!k_2!k_3!}$ if $k_1+k_2+k_3 = n$ and 0 otherwise. Then $\binom{n}{k_1,k_2,k_3}$ is the coefficient of $x^{k_1}y^{k_2}z^{k_3}$in $(x+y+z)^n$. This is sometimes called the *trinomial theorem.*

Important Concepts, Formulas, and Theorems

1. *Pascal Relationship.* The Pascal Relationship says that
$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}\,,$$
whenever $n > 0$ and $0 < k < n$.

2. *Pascal's Triangle.* Pascal's Triangle is the triangular array of numbers we get by putting ones in row n and column 0 and in row n and column n of a table for every positive integer n and then filling the remainder of the table by letting the number in row n and column j be the sum of the numbers in row $n-1$ and columns $j-1$ and j whenever $0 < j < n$.

3. *Binomial Theorem.* The **Binomial Theorem** states that for any integer $n \geq 0$
$$(x+y)^n = x^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{n-1}xy^{n-1} + \binom{n}{n}y^n\,,$$
or in summation notation,
$$(x+y)^n = \sum_{i=0}^{n}\binom{n}{i}x^{n-i}y^i\,.$$

4. *Labeling.* The number of ways to apply k labels of one kind and $n-k$ labels of another kind to n objects is $\binom{n}{k}$.

5. *Trinomial coefficient.* We define the *trinomial coefficient* $\binom{n}{k_1,k_2,k_3}$ to be $\frac{n!}{k_1!k_2!k_3!}$ if $k_1+k_2+k_3 = n$ and 0 otherwise.

6. *Trinomial Theorem.* The coefficient of $x^iy^jz^k$ in $(x+y+z)^n$ is $\binom{n}{i,j,k}$.

Problems

1. Find $\binom{12}{3}$ and $\binom{12}{9}$. What can you say in general about $\binom{n}{k}$ and $\binom{n}{n-k}$?

2. Find the row of the Pascal triangle that corresponds to $n = 8$.

3. Find the following

   a. $(x+1)^5$
   b. $(x+y)^5$
   c. $(x+2)^5$
   d. $(x-1)^5$

4. Carefully explain the proof of the binomial theorem for $(x+y)^4$. That is, explain what each of the binomial coefficients in the theorem stands for and what powers of x and y are associated with them in this case.

5. If I have ten distinct chairs to paint, in how many ways may I paint three of them green, three of them blue, and four of them red? What does this have to do with labellings?

6. When n_1, n_2, $\ldots n_k$ are nonnegative integers that add to n, the number $\frac{n!}{n_1!, n_2!, \ldots, n_k!}$ is called a *multinomial coefficient* and is denoted by $\binom{n}{n_1, n_2, \ldots, n_k}$. A polynomial of the form $x_1 + x_2 + \cdots + x_k$ is called a multinomial. Explain the relationship between powers of a multinomial and multinomial coefficients. This relationship is called the Multinomial Theorem.

7. Give a bijection that proves your statement about $\binom{n}{k}$ and $\binom{n}{n-k}$ in Problem 1 of this section.

8. In a Cartesian coordinate system, how many paths are there from the origin to the point with integer coordinates (m, n) if the paths are built up of exactly $m+n$ horizontal and vertical line segments each of length one?

9. What is the formula we get for the binomial theorem if, instead of analyzing the number of ways to choose k distinct y's, we analyze the number of ways to choose k distinct x's?

10. Explain the difference between choosing four disjoint three element sets from a twelve element set and labelling a twelve element set with three labels of type 1, three labels of type two, three labels of type 3, and three labels of type 4. What is the number of ways of choosing three disjoint four element subsets from a twelve element set? What is the number of ways of choosing four disjoint three element subsets from a twelve element set?

11. A 20 member club must have a President, Vice President, Secretary and Treasurer as well as a three person nominations committee. If the officers must be different people, and if no officer may be on the nominating committee, in how many ways could the officers and nominating committee be chosen? Answer the same question if officers may be on the nominating committee.

12. Prove Equation 1.6 by plugging in the formula for $\binom{n}{k}$.

13. Give two proofs that
$$\binom{n}{k} = \binom{n}{n-k}.$$

14. Give at least two proofs that
$$\binom{n}{k}\binom{k}{j} = \binom{n}{j}\binom{n-j}{k-j}.$$

15. Give at least two proofs that
$$\binom{n}{k}\binom{n-k}{j} = \binom{n}{j}\binom{n-j}{k}.$$

16. You need not compute all of rows 7, 8, and 9 of Pascal's triangle to use it to compute $\binom{9}{6}$. Figure out which entries of Pascal's triangle not given in Table 2 you actually need, and compute them to get $\binom{9}{6}$.

17. Explain why
$$\sum_{i=0}^{n}(-1)^i\binom{n}{i} = 0$$

18. Apply calculus and the binomial theorem to $(1+x)^n$ to show that
$$\binom{n}{1} + 2\binom{n}{2} + 3\binom{n}{3} + \cdots = n2^{n-1}.$$

19. True or False: $\binom{n}{k} = \binom{n-2}{k-2} + \binom{n-2}{k-1} + \binom{n-2}{k}$. If true, give a proof. If false, give a value of n and k that show the statement is false, find an analogous true statement, and prove it.