Combinatorics: The Fine Art of Counting

Lecture Notes - Weeks 4 and 5 – Binomial Coefficients

Note – to improve the readability of these lecture notes, we will assume that multiplication takes precedence over division, i.e. A / B*C always means A / (B*C). Binomial coefficients are written horizontally, i.e. (4 2) means "4 choose 2" or 4!/2!2!

Introduction

In the previous lecture we looked at several different counting examples that involved the binomial coefficient (n 2). These examples fell into 4 major categories:

1. **Counting Subsets –** people shaking hands, edges in a graph, and unordered pairs are all examples of counting subsets of size 2. The answer we found in each case was **(n 2)**.
2. **Summing a Sequence –** adding up the numbers from 1 to n or counting the points in a triangle with n points on each side both involve summing an increasing sequence. The answer we got was **(n+1 2)**.
3. **Distinct Partitions** – bouquets of flowers of 3 colors and ordered triples (x,y,z) that sum to n were both examples of processes that involved partitioning or categorizing a set of identical objects (e.g. blank flowers, 1s) into distinct groups (e.g. red/white/pink, x/y/z). We used 2 separators to divide the objects into 3 groups and then chose the position of the two separators from among n+2 possible positions for objects and separators, obtaining **(n+2 2)**.
4. **Block Walking** – we counted the number of direct routes along an grid of city blocks 2 blocks north/south by n blocks east/west (note that each block in the grid is bounded by streets so there are 3 parallel streets running east/west and n+1 parallel streets running north/south) . Each direct route was exactly n+2 blocks long and we had to choose 2 points along our route to walk north, resulting in **(n+2 2)**.

We will see that all these counting problems generalize in a straight-forward manner, allowing us to replace 2 with k in each of the cases above. We will eventually see one more example, the Binomial Theorem, which is the origin of the mysterious name "binomial coefficients".

Counting Subsets

While there are many ways to define the binomial coefficient (n k), counting subsets can be regarded as the most fundamental. This is why we say (n k) or "n choose k" *means* the number of ways of choosing a subset of k elements from a set with n elements (as opposed to defining (n k) in terms of some algebraic formula). Many of the most basic facts about binomial coefficients follow immediately when we use this definition as our starting point. Recall that every subset of $\{x_1, x_2, \ldots, x_n\}$ elements corresponds to a binary string of length n where the i^{th} bit is 1 if and only if x_i is in the subset.

- **(n k) = (n n-k)** – this symmetry is immediately evident if we think about the complements of the subsets we are choosing, i.e. the subset of elements we *don't* choose. In terms of bit strings, an n-bit string with k 1's has exactly n-k 0's.
- **(n 0) = (n n) = 1** – there is one subset with 0 elements (the empty set) and one with n elements (the entire set). Alternatively, there is one n-bit string with all 0's and one with all 1's.

- **(n k) = 0   if k < 0 or k > n** – there are no subsets of size k in these cases. Note that this makes total sense and is a perfectly valid definition which can't be derived from the factorial formula.
- **(n 0) + (n 1) + (n 2) + … (n n) = 2^n** - if we add up all the subsets of different sizes that are possible, we count all possible subsets and there are 2^n of them. If we add up all the n-bit strings grouped according to the number of 1's in each string, we will count all n-bit strings.
- **(n k) = n!/k!(n-k)!   for 0 ≤ k ≤ n** – this follows immediately if we apply the Mississippi rule to count the number of bit strings with k 1's and n-k 0's (or k 0's and n-k 1's).

Supposing we didn't know the last formula above for computing (n k), how might we go about figuring out what (n k) is? One approach is to compute (n k) recursively starting with the case where n is 0 and working our way up to successively larger values of n.

(0 0) = 1 and (0 k) = 0 if k ≠ 0

This is our starting point. The empty set has just one subset, itself. Now we know what (n k) is whenever n is 0. Now of course we also know (1 0) = 1 and (1 1) = 1 and we can easily figure out what (2 1) is, but rather than working in an ad hoc fashion, let's suppose we have worked at all the values of (n k) for some particular n, and let's suppose we have in our hands a set S_n with n elements. How would we compute (n+1 k)? We know (n+1 0) = 1, but what about k > 0?

We need a bigger set to work with. Let's add one new element x not in S_n and define $S_{n+1} = S_n \cup \{x\}$. We can divide the subsets of S_{n+1} with k elements into two categories: those that contain x and those that don't. The ones that don't are also k element subsets of S_n and we already know there are (n k) of these – call these the *old* sets. The ones that do contain x are simply the union of a k-1 element subset of S_n with {x}, and there are (n k-1) of these – call these the *new* sets. The number of k element subsets of S_{n+1} is just the sum of the old sets and the new sets, i.e.

(n+1 k) = (n k) + (n k-1)

Note that this holds for any value of k and any positive n (we won't define binomial coefficients for negative n). Now that we have an easy way to compute (n+1 k) in terms (n k), we can make the following table:

(0 0)

(1 0)   (1 1)

(2 0)   (2 1)   (2 2

(3 0)   (3 1)   (3 2)   (3 3)

(4 0)   (4 1)   (4 2)   (4 3)   (4 4)

(5 0)   (5 1)   (5 2)   (5 3)   (5 4)   (5 5)

(6 0)   (6 1)   (6 2)   (6 3)   (6 4)   (6 5)   (6 6)

(7 0)   (7 1)   (7 2)   (7 3)   (7 4)   (7 5)   (7 6)   (7 7)

(8 0)   (8 1)   (8 2)   (8 3)   (8 4)   (8 5)   (8 6)   (8 7)   (8 8)

(9 0)   (9 1)   (9 2)   (9 3)   (9 4)   (9 5)   (9 6)   (9 7)   (9 8)   (9 9)

Note that entries which are 0 have been omitted, but we could extend each row to the left and the right with 0's. Each entry of this table after the row with (0 0) is simply the sum of the two entries immediately above it (including the entries on the sides, where one of the entries above is 0).

If we now fill in numbers we obtain the following well known triangle of numbers often referred to as Pascal’s triangle, although this configuration of numbers was known and studied long before Pascal.

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

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

There are many interesting patterns in this triangle of numbers, but the most useful thing to notice for our purposes is that if we draw an edge between each number and the one or two numbers immediately above it, we obtain a graph with the numbers as vertices. If we now count the number of possible descending paths from the vertex at (0 0) to the vertex at (n k) we find a remarkable fact:

> **There are exactly (n k) possible descending paths from (0 0) to (n k).**

First note that there is just one way to get from (0 0) to (0 0) - just stay where you are. Any descending path from (0 0) to (n+1 k) must pass through either (n k-1) or (n k) prior to its last step. Thus the number of paths to (n+1 k) is simply the number of paths to (n k-1) plus the number of path to (n k). But this is precisely the recurrence we used to build the triangle in the first place! It follows that the number of paths to (n+1 k) is in fact (n+1 k).

An even simpler way to prove this remarkable fact is to notice the following:

> **Every descending path from (0,0) to (n,k) has length n and contains exactly k right turns (and n-k left turns).**

Thus we can view picking a path from (0,0) to (n,k) as simply a matter of choosing when to turn right. There are n distinct steps to choose from and we can pick any k of them to be right turns. Thus picking a path is the same as choosing k elements from a set of size n, or (n k).

Block Walking

We now turn to the problem of picking a path between two intersections on a grid of city streets with one intersection x blocks north and y blocks east of the other. How many direct routes are there? Every route will involve walking x blocks north and y blocks east, and if the route is direct, we will never walk south or west so the route must be exactly x+y blocks long. If we write down a description of the route using an N for each block walked north and an E for each block walked east, we obtain a string of length x+y with x N’s and y E’s. Note that we can choose to put the N’s and E’s in any order, so long as there are exactly x and y of them. We can count the number of distinct permutations of a string with x N’s and y E’s using the Mississippi rule, or we can simply note that any such string corresponds to a string of x+y bits, x of which are 1’s. Thus we find that:

> **The number of direct paths in a grid from the origin to the point (x,y) is (x+y x). This is equivalent to (x+y y).**

If we replace x and y with k and n-k, we find that x+y is just n and the number of routes is exactly the same as the number of descending paths from (0 0) to (n k) in Pascal's triangle. We can see why this is true if we widen the triangle a bit to make it a right triangle and then rotate it so that (0 0) is at the origin, the (n 0) entries are along the x-axis and the (n n) entries are along the y-axis. This will result in the entry for (n,k) having the coordinates (k,n-k).

1									
1	**9**								
1	**8**	**36**							
1	**7**	**28**	**84**						
1	**6**	**21**	**56**	**126**					
1	**5**	**15**	**35**	**70**	**126**				
1	**4**	**10**	**20**	**35**	**56**	**70**			
1	**3**	**6**	**10**	**15**	**21**	**28**	**36**		
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**

Each entry in the grid above indicates the number of direct paths from the origin to the location. Note that each entry other than the origin is the sum of the numbers below it and to it's left, i.e. the number at (x,y) is the sum of the numbers at (x-1,y) and (x,y-1) which are the two possible locations a path to (x,y) must pass through on its last step. This corresponds directly to the fact that (x+y y) = ((x-1)+y y) + (x+(y-1) y-1) = (x+y-1 y) + (x+y-1 y-1). This is the same identity we found when counting subsets. Instead of adding a new element to a set, we are increasing the length of the path by 1.

The key thing to remember when counting paths (or block walking as it is often called) is that the number you are "choosing" from is the number of steps in the path, or **x+y,** not x or y, just as a string with x 0's and y 1's has x+y bits.

Distinct Partitions

We now consider the problem of counting distinct partitions of indistinct objects. This type of problem arises in a wide variety of situations, and is perhaps the most common use of binomial coefficients in combinatorics. In many cases, however, the exact nature of the problem is not immediately obvious. The key is to recognize the problem.

Example #1: How many ways can we write an expression involving non-negative integers that uses p plus signs which is equal to n? (e.g. 2 can be written as "1+1+0", "1+0+1", "0+1+1", "2+0+0", "0+2+0", and "0+0+2" or 6 different ways using 2 plus signs).

Note that any time a problem asks how many ways you can "write" something, the order matters, even if you might regard the objects the written expressions represent as identical – e.g. "{1,2}" and "{2,1}" are two different ways to write the same set.

An equivalent (but more formal) way to state the question is, "How many ordered m-tuples of non-negative integers (x_1, x_2, …, x_m) sum to n?". The number of integers is one more than the number of plus signs, so m = p+1.

Solution #1: The simplest way to approach this problem is to consider any such expression and replace all the numbers with a string of 1s of the appropriate length (0's will simply get omitted entirely). For example "3+0+2+2+1+0" becomes "111++11+11+1+". The resulting string will contain exactly n ones and p plus signs. Any distinct rearrangement of this string is equivalent to a distinct expression that is equal to n, e.g. "+1111++11+1+1" corresponds to 0+4+0+2+1+1. This one-to-one correspondence makes it clear that the number of distinct expressions is simply the number of distinct arrangements of n 1's and p +'s, which is just (n+p p), or equivalently, (n+p n). If we count the number of integers rather than the number of plus signs, note that we must **replace p with m-1**. This is how problems are usually stated, we aren't told the number of separators, just the number of objects being separated.

> **There are (n+m-1 m-1) = (n+m-1 n) ordered m-tuples of non-negative integers that sum to n.**

In the example above it may not have been clear that we were creating distinct partitions of indistinguishable objects, but if we consider the objects being partitioned as 1's, we were grouping them into distinct classes: the first group consisted of all the ones (if any) to the left of the first plus sign, the second group consisted of all the ones (if any) between the first and second plus sign, and so on. These groups were distinct because the order of the terms in the expression was distinct.

Example #2a: How many ways can we distribute 7 identical pieces of candy to 4 children?

Solution #2a: First note that while the candies are identical, we assume the children are not. We are being asked to partition the candy among the children. We might give 3 to the first child, 1 to the second, 3 to the third, and 0 to the fourth. However we allocate the candy, we can imagine writing down 4 non-negative integers that add up to 7, and the order of the integers matters – giving 1 to the first child and 3 to the second child would change things. Thus by our analysis of ordered partitions above, the answer is (7+4-1 4-1) = **(10 3) = 120**.

An somewhat simpler way to think about the situation is to use 3 separators, say rocks, and arrange the 7 pieces of candy and the 3 rocks in a line. Given the first child all the candy before the first rock, the second child the candy between the first and second rock and so on. Note that the separators do not need to be distinct, since we distinguish groups by their order in the line. We have 10 objects in all, 7 pieces of candy and 3 rocks, and there are exactly (7+3 3) = **(10 3)** possible line ups. The key to remember here is that the kids are not the separators, they represent distinct categories we are assigning to the pieces of candy.

Example #2b: How many ways can we distribute 7 identical pieces of candy to 4 children if each child must get at least one piece?

Solution #2b: The trick here is to satisfy the constraint first, i.e. give every child one piece of candy. There is only one way to do this. Now we have 3 pieces of candy left to distribute among the four children which we will do using 4-1 = 3 separators as above. Thus there are (3+3 3) = **(6 3) = 15** ways to ensure each child gets 1 piece.

Note that the number of objects being partitioned can be less than the number of partitions.

Example #2c: How many ways can we distribute 4 identical pieces of candy to 7 children?

Solution #2c: We solve this exactly the same way we solved problem #2a. This time we need 7-1 = 6 separators to partition 4 pieces of candy and there are (6+4 6) = **(10 6) = 210** ways to do this. Equivalently, we could choose the positions of the candy instead of the separators and get (6+4 4) = (10 4) = (10 6).

The key idea in all these problems is to figure out what are the objects being partitioned and how many separators you need to use.

Example #3: How many different combinations of 5 sweaters can you buy if you have 8 colors to choose from?

Solution #3: Don't be fooled by the wording of the problem. We may have 8 colors to "choose from" but we can choose any particular color as many times as we like (up to 5) or not at all, so we are not being asked to pick a subset of the colors.

The right way to think about the problem is to think of partitioning 5 identical objects (sweaters) into 8 distinct categories (colors). Since we have 8 categories, we need 7 separators, and since the categories are distinct, we can order them color #1, color #2, etc… Now arrange the 7 separators and 5 sweaters in a line and assign all the sweaters (if any) to the left of the first separator color #1, all the sweaters (if any) in between the first and second separator color #2 and so on. Thus there are (7+5 5) = **(12 5) = 192** possible color combinations.

The Hockey Stick

The last and most interesting type of problem we will look at is a generalization of the process of summing the integers from 1 to n. It is not as obvious as it was in the cases above how to generalize from the case of (n+1 2) to (n+1 k) – what are the analogs of the triangle numbers? Increasing the number of sides doesn't really change things much, as we saw when we counted the hexagonal lattice or points, we still had triangle numbers in the answer. We might instead consider increasing the dimension. Instead of stacking rows of decreasing numbers of points to make a triangle, we might stack triangles of decreasing size to make a tetrahedron. The number of points in the tetrahedron would be the sum of the number of points in triangles with side lengths 1, 2, 3, …, n, or (2 2) + (3 2) + (4 2) + … + (n+1 2)

Another example of summing comes up in the "Twelve days of Christmas" song. In this song the singer receives lots of presents. On the first day of Christmas 1 partridge in a pear tree is received, then on the second day they get 2 turtles and (another) 1 partridge in a pear tree or 2+1 = 3 presents on the second day. On the third day they get 3+2+1 = 6 presents and so on.

Example #4: How many gifts does the singer of the "Twelve days of Christmas" receive in total over the course of the 12 days?

Solution #4: We already know how to count the number of presents on any particular day. On the ith day of Christmas there are 1+2+…+i = (i+1 2) presents. We just need to add up the number of presents over the 12 days to get (2 2) + (3 2) + (4 2) + … + (13 2). *But what is this sum?*

We could of course just compute the numbers and add them up. We could also try to find a pattern and guess some general formula and prove it correct. Alternatively, we can try to analyze the problem combinatorially. Consider the grid of city streets with the origin at point (0,0). The number of paths to the point (0,2) is (0+2 2) = (2 2). The number of paths to the point (1,2) is (1+2 2) = (3 2). If we consider the twelve points (0,2) through (11,2), the numbers of paths to these points is (2 2) through (13 2). Now consider the point (11,3). Each and every path to (11,3) moves from the street running along y=2 at some particular point to reach the street running along y=3, i.e. for each path to (11,3) there is some point where the last step north was made, and this occurred at one of the twelve points (0,2) to (11,2). Additionally, once the last step north is made, the path can do nothing but turn east at every step until it reaches (11,3). Thus for each path that crosses the street y=2 at a given point, there is exactly one direct path to (11,3).

The number of direct paths to (11,3) is thus equal to the sum of the numbers of paths to (0,2) through (11,2). But we know that there are (11+3 3) = (14 3) such paths. Therefore:

(2 2) + (3 2) + (4 2) + … + (13 2) = (14 3)

Note that in the argument above there was nothing special about the street y=2, we could have just as easily used y=k to obtain a more general result. But first we want to examine the situation a little more closely.

To understand what's really going on here, let's return to the triangle numbers. We know that 1 + 2 + 3 + … + n = (n+1 2). The number (n+1 2) is just the number of two element subsets of a set with n+1 elements. Suppose we wanted to make a list of these subsets. To be concrete, let's look at the set {0,1,2,3,4,5,6,7} which has 8 elements. If we had to make a list of the two element subsets of this set, we might organize our list as follows:

{0,1} {0,2} {0,3} {0,4} {0,5} {0,6} {0,7}	**(7 1) subsets starting with 0**
{1,2} {1,3} {1,4} {1,5} {1,6} {1,7}	**(6 1) subsets starting with 1**
{2,3} {2,4} {2,5} {2,6} {2,7}	**(5 1) subsets starting with 2**
{3,4} {3,5} {3,6} {3,7}	**(4 1) subsets starting with 3**
{4,5} {4,6} {4,7}	**(3 1) subsets starting with 4**
{5,6} {5,7}	**(2 1) subsets starting with 5**
{6,7}	**(1 1) subsets starting with 6**

If we think about what we were doing when we wrote down this list, in the first line for each subset in the list we had chose 1 of the 7 elements greater than 0. For the second line for each subset in the list we chose 1 of the 6 elements greater than 1, and so on. Adding up the number of subsets on each line we see the triangle numbers in a slightly different form, i.e.

(8 2) = (7 1) + (6 1) + (5 1) + (4 1) + (3 1) + (2 1) + (1 1) = 7 + 6 + 5 + 4 + 3 + 2 + 1

In fact when we computed the triangle numbers by summing the integers from 1 to n we weren't summing just any old integers, we were summing binomial coefficients!

Let's generalize the idea above. Suppose we now want to list all the three element subsets of the set {0,1,2,3,4,5,6,7}. The list will be a bit longer, but we'll use the same approach – list all the subsets starting with 0 on the first line, all the subsets starting with 1 on the second line and so on:

{0,1,2} {0,1,3} … {0,1,7} {0,2,3} … {0,6,7}	**(7 2) subsets starting with 0**
{1,2,3} {1,2,4} … {1,2,7} {1,3,4} … {1,6,7}	**(6 2) subsets starting with 1**
{2,3,4} {2,3,5} … {2,3,7} {2,4,5} … {2,6,7}	**(5 2) subsets starting with 2**
{3,4,5} {3,4,6} … {3,4,7} {3,5,6} … {3,6,7}	**(4 2) subsets starting with 3**
{4,5,6} {4,5,7} {4,6,7}	**(3 2) subsets starting with 4**
{5,6,7}	**(2 2) subsets starting with 5**

Notice that we had to stop one line earlier because there aren't any three element subsets that have 6 as the least element. Adding up the number of subsets in each line gives:

(8 3) = (7 2) + (6 2) + (5 2) + (4 2) + (3 2) + (2 2)

If we did the same thing and listed four element subsets we would find that:

(8 4) = (7 3) + (6 3) + (5 3) + (4 3) + (3 3)

Add in general we have the following result:

(n+1 k) = (n k) + (n-1 k) + (n-2 k) + … + (k k)

Equivalently, we could write this:

(k k) + (k+1 k) + (k+2 k) + … + (n k) = (n+1 k)

When we built Pascal's triangle, we used the identity (n+1 k) = (n k) + (n k-1) to build successive rows of the triangle. We could have just as easily built Pascal's triangle using the identity above. We start a new row with (n+1 0) = 1, and to compute the rest of the entries in the row we simply sum along with diagonal above and to the left.

The diagram below shows the triangle which when the entry for (9 3) has just been added. The diagonal which sums to (9 3) is in bold.

(0 0)

(1 0) (1 1)

(2 0) (2 1) **(2 2)**

(3 0) (3 1) **(3 2)** (3 3)

(4 0) (4 1) **(4 2)** (4 3) (4 4)

(5 0) (5 1) **(5 2)** (5 3) (5 4) (5 5)

(6 0) (6 1) **(6 2)** (6 3) (6 4) (6 5) (6 6)

(7 0) (7 1) **(7 2)** (7 3) (7 4) (7 5) (7 6) (7 7)

(8 0) (8 1) **(8 2)** (8 3) (8 4) (8 5) (8 6) (8 7) (8 8)

(9 0) (9 1) (9 2) **(9 3)**

Perhaps you can see why this combinatorial identity is affectionately known as the "hockey stick". The diagonal (2 2) … (8 2) is the handle of the hockey stick and (9 3) is the tip of the hockey stick..

The hockey stick is often useful in combination with many of the counting techniques we have already. One example is when we are partitioning a number of objects which is not necessarily fixed.

Example #5: Assuming a small packet of m&m's can contain anywhere from 20 to 40 m&m's in 6 different colors. How many different m&m packets are possible?

Solution #5: If we knew there were exactly 20 m&m's in each packet we could partition the m&m's among the 6 colors (using 5 separators) to get (20+5 5) = (25 5) packets. On the other hand there could be 40 m&m's in a packet, which would give (40+5 5) = (45 5) different packets. Or there could be any number of m&m's between 20 and 40. What we really want is the sum (25 5) + (26 5) + … + (45 5). We can use the hockey stick to simplify this sum since (5 5) + … + (45 5) = (46 6), and also (5 5) + … + (24 5) = (25 6). Subtracting we obtain (25 5) + … + (45 5) = **(46 6) – (25 6)**.

There are many more applications of the hockey stick that we will see in future problems. Of all the combinatorial identities you may encounter, this is the one most worth remembering. It makes a lot of apparently hard problems very easy.

Summary

We have seen four different ways of using binomial coefficients: **Counting Subsets**, **Summing Sequences**, **Distinct Partitions**, and **Block Walking**. There are three steps that you must take to master the art of counting with binomial coefficients:

1) Learn to recognize which of these approaches may be most applicable to a given counting problem.
2) Become adept at applying each of these techniques individually.
3) Realize that these four different ideas are all just different aspects of the same thing. Any problem involving binomial coefficients can be approached using all of them.

Many of the deepest theorems in mathematics have come from solving the same problem in more than one way.

The Binomial Theorem

No discussion of binomial coefficients would be complete without at least mentioning the theorem which is the origin of their name. Consider the process of raising a simple binomial (x+y) to the nth power starting at n = 0:

$$(x+y)^0 = 1$$

$$(x+y)^1 = x + y$$

$$(x+y)^2 = x^2 + 2xy + y^2$$

$$(x+y)^3 = x^3 + 3x^2y + 3xy^2 + y^3$$

$$(x+y)^4 = x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4$$

$$(x+y)^5 = x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + y^5$$

What's going on here? If we manually multiply each term in the expansion of (x+y)(x+y) we find that there are actually 4 terms, but 2 of them (the xy terms) are similar and can be combined. If we didn't bother combining like terms and we kept on multiplying by (x+y) we would find that the number of terms doubled each time, so that $(x+y)^n$ had 2^n terms when fully expanded. When we combine like terms we end up with only n+1 terms, one for each possible power of x (or y - note the powers of y are complementary and each term has degree n). To compute the coefficients of these terms, we need to figure out where a term x^ky^{n-k} could have come from. Each of these terms came from choosing either an x or a y from each binomial factor, and there are (n k) ways to choose x from k of the n factors. Thus the process of combining like terms is really a process of counting subsets.

If we write in the binomial coefficients explicitly and remember that $x^0 = 1$, we obtain the general result:

$$(x+y)^n = \Sigma(n\ k)x^ky^{n-k} = (n\ n)x^ny^0 + (n\ n\text{-}1)x^{n-1}y^1 + \ldots + (n\ 0)x^0y^n$$

Equivalently, if we focus on the y's instead of the x's:

$$(x+y)^n = \Sigma(n\ k)x^{n-k}y^k = (n\ 0)x^ny^0 + (n\ 1)x^{n-1}y^1 + \ldots + (n\ n)x^0y^n$$

The binomial theorem can be used to prove all sorts of wonderful identities involving the binomial coefficients. For example if we plug in the value 1 for both x and y we obtain a formula for $(1+1)^n = 2^n$:

$$2^n = (1+1)^n = (n\ 0)1^n1^0 + (n\ 1)1^{n-1}1^1 + \ldots + (n\ n)\ 1^01^n = (n\ 0) + (n\ 1) + \ldots + (n\ n)$$

We already proved this fact combinatorially, but here is a simple algebraic proof.

Combinatorial Identities

There are many other examples of identities that can be proven both algebraically and combinatorially.

Example #6: How many subsets of n elements have an odd number of elements?

Solution #6a: At first glance this seems easier to solve if n is odd, because in this situation the complement of a set with an odd number of elements has an even number of elements, so there must be the same number of odd subsets as even subsets. Thus half of all the subsets must have an odd number of elements, or $2^n/2 = 2^{n-1}$. But what about when n is even?

The approach we used above was to find a one-to-one correspondence between odd an even subsets to show that there were the same number of both. This type of correspondence is called a **bijection** and is an extremely useful counting technique that we have used several times before. The difficulty we have here is that the particular bijection we used (taking the complement) doesn't work when n is even. However there is another bijection that will work in both cases.

Solution #6b: Consider a subset as a binary string with 1's indicating membership in the subset. If we switch the first bit of such a string, the number of elements in the subset will change by 1 (up or down), which will transform any odd subset into an even one and any even subset into an odd one. This gives us a bijection between even and odd subsets that will work for any n, so there must be the same number of both, hence in all cases there are $2^n/2 = \mathbf{2^{n-1}}$ odd subsets.

A consequence of this fact is that if we take a row of Pascal's triangle and compute the alternating sum we get 0 since we are subtracting the number of odd subsets from the number of even subsets. For example (4 0) – (4 1) + (4 2) – (4 3) + (4 4) = 1–4+6–4+1 = 0. In general we have:

(n 0) – (n 1) + (n 2) – (n 3) + … ± (n n) = 0

This can also be proven algebraically by applying the binomial theorem to $(1+ (-1))^n = 0$:

$$\mathbf{(1+(-1))^n = (n\ 0)1^n(-1)^0 + (n\ 1)1^{n-1}(-1)^1 + \dots + (n\ n)1^0(-1)^n}$$

$$\mathbf{0 = (n\ 0) - (n\ 1) + (n\ 2) - (n\ 3) + \dots \pm (n\ n)}$$

Example #7: Prove that n*(n-1 k-1) = k*(n k)

Solution #7: This is straight-forward to prove algebraically using the factorial formula for the binomial coefficients, but, as is often the case with algebraic proofs, the proof is not especially illuminating (it shows that the statement is true, but doesn't always help us understand why it must be true). We will give a combinatorial proof instead. Suppose we have a group of n people and we want to choose a committee of k of them and we want to designate one person as the head of the committee, how many ways can we do this? There are two different ways we can count the possibilities. First we could pick one of the n people to be the head of the committee (n options) and then choose k-1 people from the remaining k-1 to fill up the rest of the committee in (n-1 k-1) ways – this is the LHS of the equation above. Alternatively, we could pick k people to be on the committee in (n k) possible ways, and then designate one of them (k options) to be the head of the committee – this is the RHS of the equation. We have come up with two different ways of counting the same thing, so the expressions must be equal.

Note that the identity above is often written as: **(n-1 k-1) = (k/n) * (n k)**

There are hundreds of combinatorial identities involving the binomial coefficients that can be proven using the techniques we have learned, however there are really only a few worth remembering and they all follow immediately from the concepts we have learned. These are listed below, along with the key concept behind them:

(n k) = (n n-k)	**(complementary subsets)**
(n 0) + (n 1) + … + (n n) = 2^n	**(number of n-bit binary strings)**
(n k+1) = (n k) + (n k+1)	**(constructing subsets, Pascal's triangle)**
(k k) + (k+1 k) + … + (n k) = (n+1 k)	**(hockey stick identity)**

The last one is perhaps the most useful one to memorize, as it is not as readily apparent as the others. Beyond these, virtually any other combinatorial identity can be proven as required using a combination of these identities together with an appropriate combinatorial argument (often using a bijection), or the binomial theorem (or both). We will explore several examples in the homework. The purpose of proving combinatorial identities is not to add yet another formula to a list of facts to remember – once you become adept you can derive them when you need to solve a particular problem.