Math 306: Combinatorics & Discrete Mathematics

Northwestern University, Lecture Notes

Written by Santiago Cañez

These are notes which provide a basic summary of each lecture for Math 306, "Combinatorics & Discrete Mathematics", taught by the author at Northwestern University. The book used as a reference is the 4th edition of *A Walk Through Combinatorics* by Bona. Watch out for typos! Comments and suggestions are welcome.

Contents

Lecture 1: Pigeonhole Principle

Combinatorics is the subject which deals with counting. As simple as this may sound at first, it is broad topic of interest with very deep results. The point is that we actually do not often care about the concrete numerical result of whatever count we want to perform, but rather we care more about the techniques which can be developed to perform that count *and* the relationships such techniques illuminate between the counting problem at hand and a seemingly completely different one. It is really these connections between different problems which combinatorics—and truly all of mathematics in general—seeks to emphasize and understand.

Before we get going, we will outline three different counts we will eventually see how to perform in order to highlight some of the ideas above.

Problem 1. Take n pairs of open (and closed) parentheses. We seek to form expressions using these with the stipulation that any closed) must have a matching open (somewhere before it. For instance, when $n = 1$ the only expression which works is

$$()$$

The only other possibility)(is not valid since the first) does not have a matching open (before. When $n = 2$ we have

$$(()) \text{ and } ()()$$

Something like ())(is not valid since the second closed) has no matching open (before. We let C_n denote the number of such expressions we can be formed, so for instance:

$$C_1 = 1 \text{ and } C_2 = 2.$$

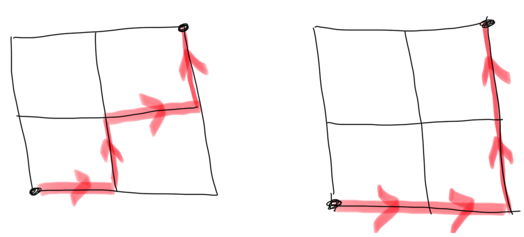
You can work out the $n = 3$ and $n = 4$ cases simply by writing out all the possibilities, and it turns out that

$$C_3 = 5 \text{ and } C_4 = 14.$$

To be sure, it will take some care to write out all the possibilities when $n = 4$, but 14 is still relatively small enough number so that it becomes feasible.

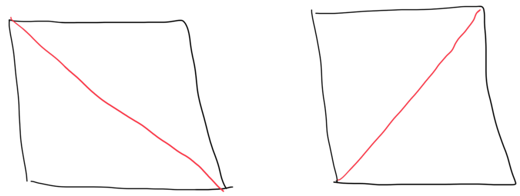
However, after this point it gets much harder to write out all possibilities as these numbers start getting large quickly. Already for $n = 5$ it turns out that $C_5 = 42$, and it would be a great task to write out all 42 of these possibilities. What we need then is a better, more efficient way of finding these numbers. It will turn out that we can find a nice *recursive* expression for these values which makes them feasible to compute. Now of course, we do not care so much what the actual value of, say, C_9 is, we care more about the techniques which allow us to find this value.

Problem 2. Take an $n \times n$ grid. We want to count paths which start at the lower left corner, end at the upper right corner, never go above the main diagonal, and which consist of only a move to the right or a move up at each step. For instance, in the $n = 2$ case we have the following two possibilities:

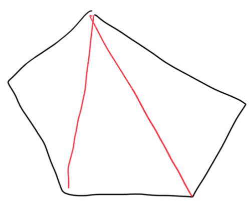


How much such paths are there? Again drawing all possibilities gets to be too cumbersome at some point, we need a better method.

Problem 3. Take a regular $(n + 2)$-gon, meaning a regular polygon with $n + 2$ sides. It turns out that any such polygon can be cut up into n triangles by connecting vertices. For instance, a square can be cut up into triangles in the following two ways:



Some possibilities for a pentagon in the $n = 3$ case are:



We wish to find an efficient way to count the number of ways in which a polygon can be cut up into triangles in this manner.

Upshot. Each of the three problems above are examples of counts where the first few simple cases can be worked out by hand, but at some point it becomes too difficult to determine all possibilities. Determining strategies for performing such counts efficiently is thus a key aspect of combinatorics.

But there is another point here to make: these three counts are *exactly* the same! That is, the number C_n the number of parenthetical expressions in Problem 1 is also the correct number in Problem 2 which counts paths, and is the number in Problem 3 which counts ways of splitting up a polygon into triangles. Thus, these three seemingly different problems actually end up being one and the same! Understanding why that is, and the deeper relation between these three problems, is the other aspect of combinatorics mentioned in the introduction above. The numbers C_n are called the *Catalan number*, and as these three problems suggest, the Catalan numbers show up in *tons* of different contexts.

The fact that so many different problems end up having the Catalan numbers as their answers suggest some deeper relation between the objects being counted in these problems. Practically, being able to view the Catalan numbers as arising in these different ways gives powerful ways of studying them. For instance, the "parentheses" point of view will give (as we mentioned) a nice recursive expression for the Catalan numbers (which will later make it possible to find their so-called *generating function*), while the "path" point of view will make it easier to find an explicit closed-form expression for C_n itself. So begins our walk into combinatorics!

Pigeonhole Principle. We start with a really basic fact, which illustrates the idea that seemingly complex problems can become quite simple if viewed in the right way. The *Pigeonhole Principle* states that if we are attempting to put "pigeons" into "hole", and we have more pigeons than holes, at least one hole will contain multiple pigeons. Hopefully you agree that this is an "obvious" statement, and you might not think it seems all that interesting. But again, the point is that many different problems can be phrased as "pigeonhole problems", and this principle helps to make them simpler to solve.

Truthfully it is not as if the Pigeonhole Principle will be a crucial fact going forward which we will use all the time, but rather we look at it now as an illustration of making complex problems easy by viewing them in the right way. (Of course, we will not be dealing with literal pigeons and holes—these are simpler terms commonly used to refer to more general types of "objects" and "boxes/containers" they are to be placed into.)

Example 1. Take any 5 random points on a sphere. The claim is that there is a closed hemisphere we can come up with which contains at least 4 of these points. That is, there is a way in which we can cut the sphere in half so that at least 4 points end up on the same half. The point is that we have no control over what the 5 points look like or where they are, but regardless we claim we find such a hemisphere. (To be clear, if a point happens to lie on the "equator" between two hemispheres, we consider that point as belonging to both hemispheres in question.) This is the kind of problem where if we start drawing examples, it seems clear that a correct "cut" can be always be made to divide the sphere in two, but giving a precise proof that this is always possible seems difficult to approach.

Here is the solution. Take any two of the points and look at the plane containing these two points and the origin, assuming that our sphere is indeed centered at the origin. This plane cuts the sphere into two halves. With 3 remaining points, the Pigeonhole Principle guarantees that at least two of these points lies in the same half, and so this hemisphere contains the 4 points we wanted: the two we get from the Pigeonhole application, and the two on the "equator" used to make the cut in the first place. Thus, once we find the right cut to make, the Pigeonhole Principle takes over to finish it off.

Example 2. Here is a basic fact which you no doubt first saw way back when in your schooling, but which perhaps you have not thought about fully. Recall that a *rational number* is a fraction $\frac{a}{b}$ of integers a, b where b is positive. The claim is that any rational number has a decimal expansion which ends in a repeating pattern. For instance,

$$\frac{13}{123} = 0.105691056910569\ldots.$$

To be clear, we treat a finite decimal expansion as ending in a repeating pattern of zeroes:

$$\frac{1}{4} = 0.2500000000\ldots.$$

To see that this is always true, the key observation is to notice the behavior of the *remainder* obtained when performing long division in order to obtain the require decimal expansion. The point is that once a remainder has been repeated, the entire decimal expansion begins to itself repeat since the same remainders as before are being used to compute remaining digits. For $\frac{a}{b}$ with $b > 0$, there are only b possible remainders obtained when dividing by b. So, if nothing else at the $(b+1)$-st digit in the decimal expansion we must come across a repeated remainder according to the Pigeonhole Principle since at this point we will have had $b+1$ remainders taking up b-many possibilities. Thus the decimal expansion of $\frac{a}{b}$ must end in a repeating pattern.

Example 3. Consider the numbers consisting of a string of more and more 1's:

$$1, 11, 111, 1111, \ldots$$

We claim that at least one of these is divisible by 13. Actually, working this out by hand is not too bad since it turns out that 111111 is divisible by 13, but the point is to find a way to justify this without having to do a bunch of dirty work. Set a_i to be the number:

$$a_i = \underbrace{11\ldots1}_{i \text{ times}}.$$

The remainders obtained when dividing $a_1, \ldots, a_{14}$ by 13 can only be one of the 13 possible values

$$0, 1, 2, \ldots, 12,$$

so by the Pigeonhole Principle at least two of the numbers $a_1, \ldots, a_{14}$ must have the same remainder upon division b 13—call them a_k and a_ℓ with $k > \ell$. Then $a_k - a_\ell$ has remainder zero when dividing by 13, so it is divisible by 13. But

$$a_k - a_\ell = \underbrace{11\ldots1}_{k-\ell \text{ times}}\underbrace{00\ldots0}_{\ell \text{ times}} = a_{k-\ell} \cdot 10^\ell$$

Since $a_{k-\ell} \cdot 10^\ell$ is divisible by 13 but 13 and 10^ℓ share no common factors, it must be the case that $a_{k-\ell}$ is divisible by 13, so one of the numbers in our original list is indeed divisible by 13.

Example 4. Say that two people are "friends" if they know each other, and "strangers" if they do not. The claim is that any group of 6 people contains 3 mutual friends or 3 mutual strangers, i.e. a subgroup of 3 people who all know each other, or a subgroup of 3 people where none knows the other two. (This is actually reflective of a certain topic in *graph theory* called *Ramsey Theory*, which you would learn about in Math 308. Here we only give this as an application of the Pigeonhole Principle, and will not mention the graph-theoretic interpretation.)

Take one person, called person A. By the Pigeonhole Principle, among the other five there must be either at least 3 who all know A or at least 3 none of which know A. Indeed, these 5 people fit into two "boxes" (depending on whether or not they know A), and in this case the Pigeonhole Principle does not only say that there is a box containing more than one person, but we can actually argue that at least one box contains 3 people.

In the first case, where A knows at least 3 others, consider those three others. If none of these knows the other two, then they form the group of 3 mutual strangers we want. If any two of these know each other, then those two together with A form the group of 3 mutual friends. In the second case, where A does not know at least 3 others, a similar reasoning applies. If those 3 all know each other, they are a group of 3 mutual friends; otherwise at least two of these do not know each other, which then together with A form a group of 3 mutual strangers.

Finishing off this problem required some clever reasoning, but nonetheless the Pigeonhole Principle is what got us started. Never doubt the power of simple ideas! We should note that this problem actually has a graph-theoretic interpretation, which you would see if you went on to take Math 308 in the spring. A *graph* is simply a collection of vertices (i.e. dots) and edges (i.e. lines) connecting some or all of them. If you interpret a person as a "vertex" and draw an edge between any vertices which correspond to "friends", you can construct a graph which models this scenario. This problem shows that such a graph as a certain property, but we will not get into this here.

Lecture 2: Mathematical Induction

Warm-Up 1. Say M is a set of 9 positive integers whose only prime factors are $2, 3$, and 5. We claim that there are two elements of M whose product is a perfect square. Here are two observations: any element of M looks like $2^a 3^b 5^c$ for some $a, b, c \geq 0$, and a product of such things—which itself will have a similar factorization—is a perfect square precisely when all the exponents are even since:

$$2^{2k} 3^{2\ell} 5^{2m} = (2^k 3^\ell 5^m)^2$$

Thus, the claim is really that there are two elements of M whose product in factored form has all even exponents. But in order to obtain such a product, we need the exponents of 2 in the individual elements to have the same *parity* (meaning they should both be odd or both be even), the exponents of 3 to have the same parity, and the exponents of 5 to have the same parity. In other words, if one elements has a factor 2^{even} and the other has 2^{odd}, then their product has

$$2^{even+odd} = 2^{odd}$$

and this cannot be part of a perfect square.

Thus, distribute our 9 elements into the 8 "boxes" arising from the 8 possibilities for parities of the three exponents in $2^a 3^b 5^c$: a even, b even, c even; a even, b even, c odd; and so on. There are 8 such "boxes" since each of the three exponents a, b, c can have one of two parities (even/odd) and $2 \cdot 2 \cdot 2 = 8$. By the Pigeonhole Principle, there are two elements of M whose exponents have the same parity, and the product of these then will involve all even exponents, and so this will be the perfect square we want.

Warm-Up 2. Say n people are at a party, and each shakes hands with some (no restriction on how many) of the others as they arrive. We claim that there are at least two people who will have shaken the same number of hands by the end. Indeed, each person will shake anywhere from 0 to $n-1$ hands, so we construct boxes which indicate how many hands a person shook. There are n such boxes, but actually only $n-1$ of them can be nonempty, since if there is a person who should 0 hands, then there cannot be one who shook $n-1$ hands, or if there is someone who shook $n-1$ hands, there cannot be one shook 0 hands. Thus we are grouping n people into $n-1$ possible (nonempty) boxes, so the Pigeonhole Principle guarantees that two people end up in the same box, meaning they shook the same number of hands.

Again, we will note that this problem also has a graph-theoretic interpretation. Take as vertices the people at the part and draw an edge between two vertices whenever the corresponding people have shaken hands. The number of edges coming out of a given vertex is called the *degree* of that vertex, and the claim is that there are at least two vertices which have the same degree!

Induction. As with the Pigeonhole Principle, *induction* is a topic which we will use from time to time, but will not be a major focus of the course. However, the reason why we are covering it now is to illustrate strategies we will use (such as recursion), and specifically the idea of using something we know to build up to something more complicated. Indeed, this to me is the entire point of induction: use what you know about one scenario to build up to a larger one. Instead of giving a big spiel as to what induction is and how to perform it at the start, we will simply illustrate how it works directly through various examples.

Example 1. We claim that for any positive integer $n \geq 1$, the following identity holds:

$$1 + 2 + 2^2 + \cdots + 2^{n-1} = 2^n - 1$$

where on the left we add up powers of 2 starting with the 0-th power $2^0 = 1$. The key idea in induction is to say that *if* we knew this formula was true for some n, can we use that alone to show it would be true for the next positive integer? In other words, we want to use the $n = 1$ formula to build up to the $n = 2$ formula, the $n = 2$ formula to build up to the $n = 3$ formula, and so on. As a sanity check, we can check that the formula indeed holds for some small cases:

$$\text{for } n = 1, 1 = 1 = 2^1 - 1 \text{ is true}$$

$$\text{for } n = 2, 1 + 2 = 3 = 2^2 - 1 \text{ is true}$$

$$\text{for } n = 3, 1 + 2 + 2^2 = 7 = 2^3 - 1 \text{ is true.}$$

Of course, we cannot check every possible case by hand, and induction gives a way to definitively justify this claim without having to do so.

So, say we know that

$$1 + 2 + 2^2 + \cdots + 2^{n-1} = 2^n - 1$$

is true for some $n \geq 1$. (This is what's called the *induction hypothesis*.) We want to then build up to the fact that

$$1 + 2 + 2^2 + \cdots + 2^n = 2^{n+1} - 1$$

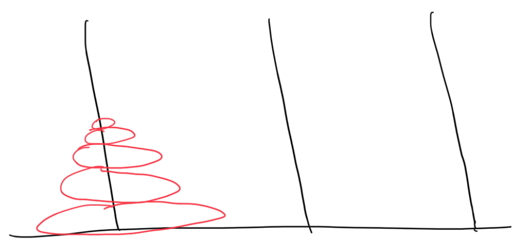
should also be true, which is what the claimed identity would say for the next integer $n+1$. The key is to find the previous "induction hypothesis" case somewhere within the case at hand, since we only know something definitive about this previous case. Here, we can realize that the $1 + 2 + \cdot + 2^{n-1}$ sum from the previous case occurs within the left-hand side of our current case, which is how we can build up to this case. We have:

$$1 + 2 + \cdots + 2^n = (1 + 2 + \cdots + 2^{n-1}) + 2^n = (2^n - 1) + 2^n = 2 \cdot 2^n - 1 = 2^{n+1} - 1.$$

Thus, if we know that one case $1 + 2 + 2^2 + \cdots + 2^{n-1} = 2^n - 1$ holds, we can use this to build up to the next case $1 + 2 + 2^2 + \cdots + 2^n = 2^{n-1} - 1$. Since we do have a starting case where the claimed identity is definitely true (say the "base case" $n = 1$), induction then says that the claimed identity does hold for all $n \geq 1$.

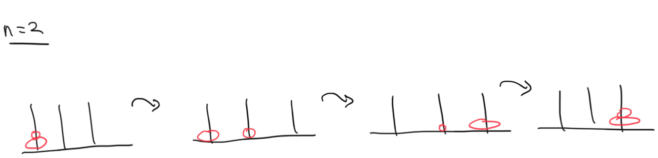
Example 2. In the example above, induction was used to justify some formula. We will use induction to do this from time to time, but just as often we will use induction not to prove some formula, but rather to show that a certain problem *can* be solved by, of course, building up from one scenario where the solution is known to a larger scenario.

The *Towers of Hanoi* puzzle begins with a stack of n disks in increasing size on one peg and two additional empty pegs:



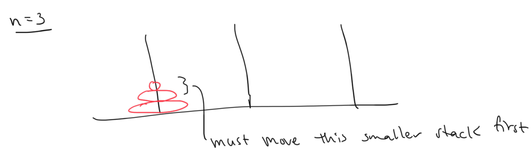
The goal is to move the entire stack onto one of the empty pegs after some number of moves, where at each step you can only move a disk onto an empty peg or onto a larger disk, but you can never move a larger disk onto a smaller disk. The question we are interested in is determining the minimal number of moves needed to solve the puzzle. Denote this minimal number of moves when we have n disks by a_n.

To get a feel for the puzzle, we can start by working out some simple cases. With $n = 1$ disk, certainly only $a_1 = 1$ move is needed: we just move that one disk onto a different peg. With $n = 2$ disks, $a_2 = 3$ moves are needed:



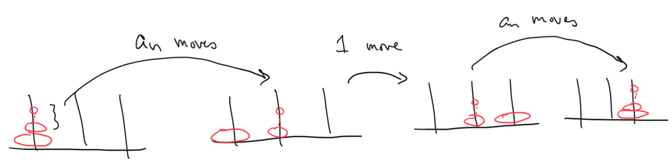
If you work it out by hand with $n = 3$ disks, you will see that $a_3 = 7$ moves are needed, but going beyond this to working out the case of more disks by hand will get tedious. However, based on these three values we can make a guess as to what the answer might be in general: $a_n = 2^n - 1$. This is actually true, and the goal now becomes to find a way to justify it.

If we are thinking along the lines of induction, we must then figure out how we can use the solution of the puzzle for a given number disks to build up to the solution for the next larger number. For instance, how can we use the $n = 2$ solution drawn above to build up to the $n = 3$ solution, and so on? To figure this out we must find the "previous" scenario hiding within our current one. Note that we are trying to solve the $n = 3$ case, it turns out that before we ever move the largest bottom disk, we must use the $n = 2$ solution to move the smaller 2-disk stack above off of that bottom disk:



This is how we can use the solution for one scenario to build up to a larger one: in general, with $n+1$ disks, we have to use the n-disk solution to move the smaller n-disk stack off the largest disk.

So, it takes a_n many moves to move these smaller stack, then 1 move to move the largest disk, and then another a_n many moves to move the smaller stack *back onto* the largest disk:



Thus, if $a_n = 2^n - 1$ is indeed true, we get that a_{n+1} should be

$$a_{n+1} = (2^n - 1) + 1 + (2^n - 1) = 2 \cdot 2^n - 2 + 1 = 2^{n+1} - 1.$$

Hence we conclude by induction that $a_n = 2^n - 1$ is indeed the correct minimal number of moves in general. Note that we can actually phrase this recursively: what we have really argued is that

$$a_{n+1} = a_n + 1 + a_n = 2a_n + 1,$$

and then we have used induction together with the fact that $a_1 = 1$ to show that $a_n = 2^n - 1$. This *recursive* way of interpreting this problem is why we were able to solve it in the end.

Back to Example 1. The value $2^n - 1$ for the number of moves needed in the Towers of Hanoi puzzle is precisely the one we saw in Example 1 as the value of the sum $1 + 2 + 2^2 + \cdots + 2^{n-1}$. So this begs the question: is there a way to interpret the sum $1 + 2 + 2^2 + \cdots + 2^{n-1}$ also in terms of the Towers of Hanoi puzzle? This is a common thought process in combinatorics: if we find a certain number as the result of a counting problem, and know that that number is related to some others in a different way, it would be good to know if this other numbers also related to the original counting problem somehow.

Indeed, $1 + 2 + 2^2 + \cdots + 2^{n-1}$ *should* also describe the number of moves needed in the Towers of Hanoi puzzle, since it breaks down the number of total moves needed according to how many times each *disk* moves in a valid solution. The largest bottommost disk will move exactly once overall, which is what gives the initial 1 in the sum $1 + 2 + 2^2 + \cdots + 2^{n-1}$. Next, the next largest disk moves twice: once to get off the largest disk and once to get back on; this is what gives the 2^1 in the sum. In general, we can see that each disk must move twice as many times as the one directly below, since for each move of the disk below the one on top must move off and then back on. So, the third largest disk moves 2^2 many times, the next one 2^3 many times, and so on. The total number of moves needed broken down according to the number of times each disk moves is then

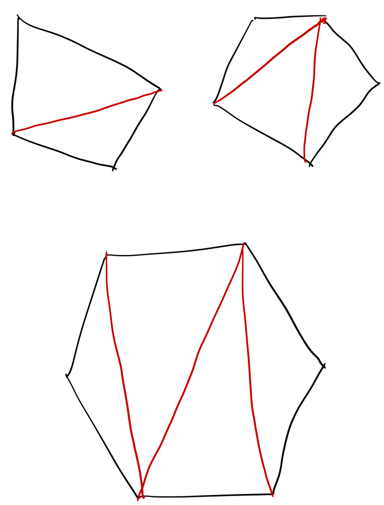
$$1 + 2 + 2^2 + \cdots + 2^{n-1},$$

and so it makes sense that $1 + 2 + \cdots + 2^{n-1} = 2^n - 1$ since both sides count the number of moves needed in the Towers of Hanoi puzzle, only in different ways. Huzzah!

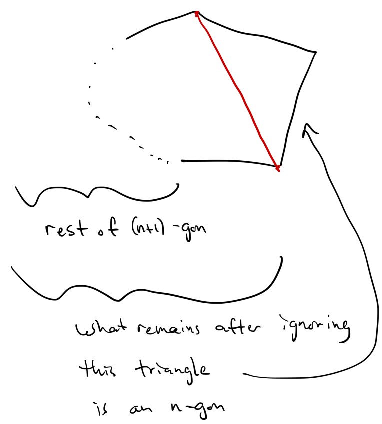
This one example illustrates many of the themes we will see in this course. We can indeed find a concrete numerical answer to our problem, $2^n - 1$ in this case, but more interesting is that fact that we can do this in multiple ways: via induction, recursively, and by performing the count in a different way, say keeping track of how many times each disk moves in this case. Moreover, the identity $1 + 2 + \cdots + 2^{n-1} = 2^n - 1$ which we get for free is not some random equality, but actually encodes something special within it, corresponding to different approaches to solving this puzzle. The idea that we can interpret both sides of this identity as counting the *same* thing is a crucial one which we will make use of time and time again.

Example 3. Finally, here is one more example where we use induction to show that some process can always be carried out. We claim that for $n \geq 3$, any convex n-sided polygon can be split up into $n - 2$ triangles. (A polygon is a shape all of whose sides are straight line segments, and saying a polygon is *convex* means that the line segment connecting two points in the polygon is itself fully contained in the polygon.) We will induct on n, the number of sides.

The base case $n = 3$ is a single triangle, so there is nothing to do since this already consists of $3 - 2 = 1$ triangle. If we consider a few examples, we see that we can clearly do what is being claimed:



We cannot very well check every possible polygon, so we now need a way to argue this can be done in general. Assuming we can do this for any n-gon (for some $n \geq 3$), we need to build up the case of an $(n + 1)$-gon. Given some $(n + 1)$-gon, we must thus find the "induction hypothesis" case of an n-gon hiding within our $(n + 1)$-gon. But observe that if we connect two vertices like so:



so we connect two vertices which happen to be adjacent to the *same* vertex, we end up dividing our original $(n+1)$-gon into an n-gon and a triangle; the resulting n-gon can be broken up into $n-2$ triangles by the induction hypothesis, so in the end we get that our original $(n+1)$-gon is broken up into $(n-2)+1 = (n+1)-2$ triangles as required. Note that convexity was used to guarantee that the segment we introduced to connect the two vertices above does indeed result in an n-gon and a triangle.

Lecture 3: More on Induction

Warm-Up 1. Define the numbers a_n by setting $a_0 = 1$ and $a_{n+1} = 10a_n - 1$ for $n \geq 1$. We show that these numbers are given by

$$a_n = \frac{8 \cdot 10^n + 1}{9}.$$

We can verify a few values by hand just to see that formula seems plausible:

$$a_0 = 1 = \frac{8 \cdot 10^0 + 1}{9}, \; a_1 = 10a_0 - 1 = 9 = \frac{8 \cdot 10 + 1}{9}, \; a_2 = 10a_1 - 1 = 89 = \frac{8 \cdot 10^2 + 1}{9}.$$

If we know that the stated formula is true for a_n, then we have:

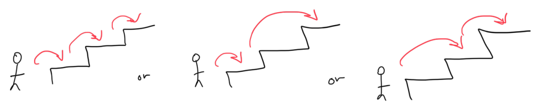
$$a_{n+1} = 10a_n - 1 = 10\left(\frac{8 \cdot 10^n + 1}{9}\right) - 1 = \frac{8 \cdot 10^{n+1} + 10}{9} - 1 = \frac{8 \cdot 10^{n+1} + 1}{9},$$

which is the required formula for a_{n+1}. Hence by induction we conclude that the stated formula holds for all $n \geq 0$.

Warm-Up 2. Say we are walking up a set of stairs, where at each point we can either go up one step or two steps. Let $f(n)$ denote the number of ways of reaching the n-th stair. We claim that these values satisfy the following bound:

$$f(n) \leq \left(\frac{1+\sqrt{5}}{2}\right)^n \quad \text{for } n \geq 0.$$

We can determine a few values by hand: $f(0) = 1$ since there is one way (i.e. do nothing!) to get to the 0-th step, which is the starting point; $f(1) = 1$ since we can only get to the first step by climbing one step; $f(2) = 2$ since we can get to the 2-nd step either by climbing one step up twice *or* by taking a single two step climb; and $f(3) = 3$ since we can either up one step three times, go up one then up two, or go up two then up one:



What we need in general is a way to relate $f(n)$ to previous values.

The key observation is that right *before* we reach the n-th step, we must either have been at the $(n-1)$-st step, in which case we will then move one step up, or we must have been at the $(n-2)$-nd step, in which case we will then move two steps up. Thus, we can recursively compute $f(n)$ by splitting the possible ways of reaching the n-th step into two groups: those which go through the $(n-1)$-st step and those which go through the $(n-2)$-nd step. There are $f(n-1)$ climbs of the first type and $f(n-2)$ of the second, so we get

$$f(n) = f(n-1) + f(n-2) \text{ for } n \geq 2.$$

The resulting numbers $f(n)$ are (starting with $n = 0$) then:

$$1, 1, 2, 3, 5, 8, 13, 21, \ldots$$

which you might recognize as the *Fibonacci sequence*. Coming up with his recursion was really the main point of this problem—the induction which follows is just icing on the cake.

For convenience, set $\phi = \frac{1+\sqrt{5}}{2}$. This number is known as the *golden ratio*, and the key property it satisfies for our purposes is that $\phi^2 = \phi + 1$. So, if we know inductively already that

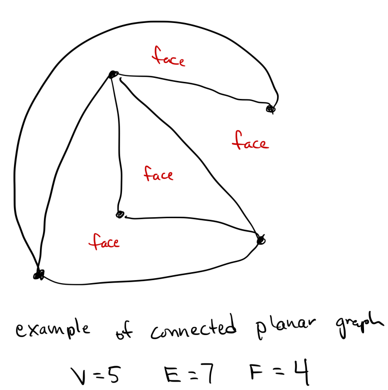
$$f(n-1) \leq \phi^{n-1} \text{ and } f(n-2) \leq \phi^{n-2},$$

then we get:

$$f(n) = f(n-1) + f(n-2) \leq \phi^{n-1} + \phi^{n-2} = \phi^{n-2}(\phi + 1) = \phi^{n-2}\phi^2 = \phi^n,$$

so the stated inequality holds for all n as claimed.

Euler's formula. At various points this quarter we will look at examples dealing with graphs, as we have already alluded to a few times. These are objects you will spend more time learning about in Math 308, so we will only look at a few properties here. Recall that a *graph* is a collection of dots (i.e. vertices) and lines (i.e. edges) connecting them. A graph is *planar* if it can be drawn so that no two edges cross one another, and a graph is *connected* if it is possible to reach any vertex from any given one by moving along edges:

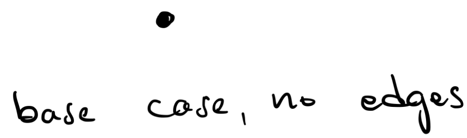


Any planar graph divides the plane up into regions we call *faces*; for instance, the planar graph drawn above has 4 faces. We will now prove *Euler's formula*, which states that any connected, planar graph satisfies

$$V - E + F = 2$$

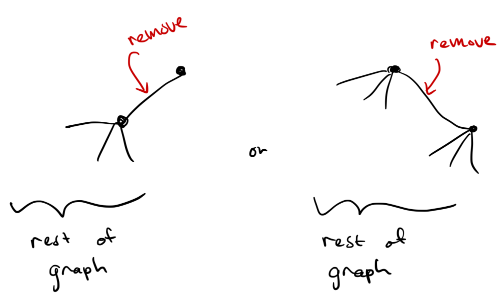
where V is the number of vertices, E the number of edges, and F the number of faces.

We will induct on the number E of edges. The simplest graph will have $E = 0$ edges, in which case we just have a single vertex:



Thus in this base case we have $V = 1, E = 0, F = 1$, so $V - E + F = 2$ as Euler's formula claims. In order to make the induction work, we need to figure out how to build up to larger cases from simpler ones. The key is to notice what happens if we add in an additional edge: either we add an edge which connects an existing vertex to a new vertex we introduce, as in the first picture below, or we add an edge which connects two existing vertices, as in the second picture. In the first scenario, F remains unchanged but V and E increase by 1, while in the second V is unchanged but E and F increase by 1. The fact that E is being subtracted guarantees that the quantity $V - E + F$ does not actually change.

Let us write this out a little more cleanly. Suppose Euler's formula holds for any connected planar graph with n edges and consider any connected planar graph with $n + 1$ edges. In order to be able to apply our induction hypothesis, we must find the n edge case hiding in our $n + 1$ edge case. But we can go from $n + 1$ edges down to n edges by removing an edge in one of two ways:



In the first case we must also remove the isolated vertex we are left with if our graph is to remain connected. Let V, E, F denote the number of vertices, edges, and faces in the smaller graph, which thus satisfies $V - E + F = 2$ by the induction hypothesis. Now we put the removed edge back in to obtain our original graph. In the first case above, we then have that our original graph has $V + 1$ vertices, $E + 1$ edges, and F faces, so

$$(V + 1) - (E + 1) + F = V - E + F = 2$$

holds for the original graph. In the second case, the original graph has V vertices, $E + 1$ edges, and $F + 1$ cases, so

$$V - (E + 1) + (F + 1) = V - E + F = 2$$

holds here as well. Hence Euler's formula holds for any connected, planar graph by induction.

Counting subsets. It will often be the case that when performing a certain count, we will need to know how many *subsets* a given set has, so we determine this now in multiple ways. For our sake, a *set* is simply a collection of objects, often finite, and a subset is just some possibly smaller collection of those objects. For instance, the possible subsets of $\{1, 2, 3\}$ (the set containing $1, 2$, and 3) are:

$$\{\}, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}.$$

Thus $\{1, 2, 3\}$ has 8 subsets. You can compute a few more small examples and guess that in general, it seems a set with n elements will have 2^n subsets, which is in fact true.

Here is one approach. We can specify a subset by going through our list of elements an declaring whether it should be included or excluded from the subset. For instance, in the example above, to construct the subset $\{2\}$ of $\{1, 2, 3\}$, we go through the elements of $\{1, 2, 3\}$ and say that

1 should be excluced, 2 included, and 3 excluded.

For an n element set $\{a_1, a_2, \ldots, a_n\}$, there are two choices (include/exclude) for what can be done with a_1 when constructing a subset, two choices for what can be done with a_2, and so on. This gives

$$\underbrace{2 \cdot 2 \cdots 2}_{n \text{ times}} = 2^n$$

ways of including/excluding elements when constructing subsets, so there are 2^n many subsets.

For a second approach, we use induction. We can recursively relate the the number of subsets of a set with n elements to the number for sets with fewer elements as follows. We split up the possible subsets of $\{a_1, \ldots, a_n\}$ into two groups: those which do not contain a_n, and those which do. These two groups are disjoint (meaning they have no overlap), so:

$$(\# \text{ of subsets of } \{a_1, \ldots, a_n\}) = (\# \text{ of which do not include } a_n) + (\# \text{ of which do}).$$

A subset which does not include a_n can be viewed as a subset of the smaller set $\{a_1, \ldots, a_{n-1}\}$, and by induction we assume this has 2^{n-1} subsets. A subset which does include a_n is obtained by taking one which does not and throwing in a_n; for instance, in the case of $\{1, 2, 3\}$, all subsets which do contain 3 are obtained by taking the ones which do not:

$$\{\}, \{1\}, \{2\}, \{1, 2\}$$

and including 3. This says that there should be as many subsets which do contain a_n as there are ones which do not, since we can obtain one of the former in this case from precisely one of the latter. Thus (# of which do) in the expression above is also 2^{n-1} by the induction hypothesis. Thus

$$(\# \text{ of subsets of } \{a_1, \ldots, a_n\}) = 2^{n-1} + 2^{n-1} = 2 \cdot 2^{n-1} = 2^n$$

as claimed. Again the key here was in recognizing how to relate the count of subsets to smaller such counts, and also in recognizing how to count the number of subsets which do contain a_n by relating to the number number of which do not. If a_n denotes the number of subsets of an n-element set, we have actually in the course of this derived the recursive expression $a_n = 2a_{n-1}$ with a starting point of $a_0 = 1$ since $\{\}$ has exactly one subset, namely itself.

We will come back to this problem of counting subsets later on from another point of view. The idea is that we can also count the number of subsets by grouping them according to how large they are: we have that the total number of subsets of an n-element set is given by

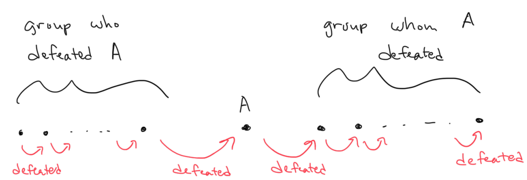
$$(\# \text{ with 0 elements}) + (\# \text{ with 1 element1}) + (\# \text{ with 2 elements}) + \cdots + (\# \text{ with } n \text{ elements}).$$

Once we have some nice way of expressing each of these terms, we will have another way of determining the total number of subsets. Even better: since we already know the answer should be 2^n, the fact the expression above counts the same thing should say that it itself will equal 2^n, and so we will get a nice identity for free.

Lecture 4: Counting Strategies

Warm-Up. Consider a round-robin tournament with n players, where "round-robin" means that each person plays every other exactly once. We claim that these n players can be arranged in a line having the property that each person defeated the person after: Suppose we know we can do this whenever we have fewer than n players. The question is then: how can we build up from these smaller situations to the current n person one?

Take one player, and call them A. Then everyone else can be divided into two groups: those who defeated A and those whom A defeated. Each of these groups has fewer than n people, so by the inductive hypothesis we may assume that each of these groups can be arranged into the required type of line separately:



Then construct one big line with first consists of the line for the first group (people who defeated A), then has person A, and then has the line of line in the second group (people whom A defeated). This big line then has the required property.

How to count. We now begin the task of figuring out how to count things more formally. There are four basic ideas we will exploit again and again:

- a count for a situation which can be broken up into disjoint pieces can be performed by *adding* the individual counts for those pieces;
- a count for a situation which can be structured as making a sequence of choices can be performed by *multiplying* the individual counts for those choices, as long as we deal with possible overcounting;
- make use of previous counts to *recursively* build up to a current count; and
- count something else and use this to derive the count we want.

These four techniques essentially give us everything we will need, except for possibly in addition the idea of using *generating functions*, which we will come to in the final few weeks.

Example 1. We have already made use of the first two strategies previously. For instance, when determining the number of subsets of an n-element set $\{a_1, \dots, a_n\}$, one approach was to break these subsets up into two disjoint groups: those which contain a_n and those which do not. We obtained the total number of subsets of adding together the sizes of these two groups, which is the first strategy above. Actually, we also used the fourth strategy here: to determine the number of subsets which do contain a_n, we instead count the number of which do not (i.e. we count something else) and then argue that this number is the same as the one we want.

Alternatively, another approach was to structure the process of constructing a subset as the sequence of choices we make to include/exclude each element: for $\{1, 2, \dots, n\}$, we choose to include/exclude 1, then 2, then 3 and so on. Each choice has 2 options, so we multiply (i.e. the second strategy above) to get $2 \cdot 2 \cdots 2$ (n times) as our answer. We also alluded to the idea of counting the number of subsets with 0 elements, the number with 1 elements, the number with 2 elements, and so on, which is also a reflection of the first strategy.

Example 2. We count the number of ways of rearranging the letters in MISSISSIPPI in order to produce *different* "words". To be clear, by "word" we mean simply some string of letters, regardless of whether or not the string corresponds to an actual English word, and we want different words in the sense that, for instance, rearranging the four S's among themselves does not lead to a new word, or similarly for rearranging the I's among themselves, or the P's.

First, there are 11 symbols in total in MISSISSIPPI, and there are 11! ways of arranging these symbols in some order: structure this as making a choice for the first symbol, then the next, and so on, and make use of the second strategy: $11 \cdot 10 \cdot 9 \cdots 2 \cdot 1$, where at each location we have one less symbol we can use. However, rearranging these 11 symbols in this way leads to overcounting, since for instance, swapping the first two S's with one another does not lead to a new word. To deal with this overcounting we divide by the number of ways of rearranging the S's among themselves, times the numbers of ways of rearranging the I's among themselves, times the analogous number for the P's:

$$\frac{11!}{4!4!2!}$$

The idea is that if we have one valid "word", we get 4!4!2! many copies of the *same* word, so 11! overcounts by a factor of 4!4!2! and hence we must divide. Dividing in this way is how we will usually, but not always, deal with overcounting.

Catalan numbers. The *Catalan numbers* were briefly introduced on the first day, and we now start to work towards understanding them better. In particular, we give an example of using the third strategy: using recursion to perform a count built out of smaller counts.

The definition of the Catalan numbers we use here is the following: C_n is the number of ways to form expressions with n open (and n closed) having the property that each closed) has a matching open (somewhere beforehand. For $n = 1$ we have () as the only valid example, so $C_1 = 1$, while for $n = 2$ we have

$$(()) \text{ and } ()()$$

as the only examples, so $C_2 = 2$. We can also write out the possibilities for $n = 3$ by hand:

$$((())), ()()(), (()()), (())(), ()(())$$

so $C_3 = 5$. The value of C_4 ends up being not so large that it becomes infeasible to write out all the possibilities, but it is a little cumbersome, and certainly once we are at the $n = 5$ case we really need a better strategy.

Consider some of the possibilities for $n = 4$:

$$(((()))), (()()()), (()())(), (())()().$$

What we can pick up on is that we have previous cases "hidden" within each of these! For instance, in the first one we have one of the possibilities for $n = 3$ within one big set of parentheses, as is also the case in the second example written above:

$$((((())))) = (\underbrace{((()))}_{n=3\text{ configuration}}) \qquad (()()()) = (\underbrace{()()()}_{n=3\text{ configuration}})$$

But also we can find $n = 2$ and $n = 1$ configurations hiddin throughout:

$$(()())() = (\underbrace{()()}_{\substack{n=2\\\text{configuration}}})\underbrace{()}_{\substack{n=1\\\text{configuration}}} \qquad (())()() = (\underbrace{()}_{\substack{n=1\\\text{configuration}}})\underbrace{()()}_{\substack{n=2\\\text{configuration}}}$$

This suggests that we can hope to recursively find C_4 using all of C_3, C_2, and C_1. The key in each of these is to pick out the closed) which matches with the initial open (, and see what occurs before and after this closed). For convenience, it will make sense to set $C_0 = 1$ and treat an "empty" space as a valid configuration for $n = 0$; for instance, the (((()))) example should be thought of as

$$(((()))) = (\underbrace{((()))}_{\substack{n=3\\\text{configuration}}})\underbrace{\quad}_{\substack{n=0\\\text{configuration}}}$$

Here then is our approach for the general case. Say we have $n + 1$ pairs of open and closed parentheses. Pick out the closed) which matches with the initial (. Within this pair of initial (and matching) we must have an arrangement of parentheses which contributes to one of the smaller Catalan numbers, and *also after* this pair we must have a valid smaller configuration:

$$(\ldots)\ldots$$

where ... denote smaller valid configurations. In particular, if we have k pairs of parentheses within the initial (and matching), we have $n - k$ pairs left to use after this pair:

$$(C_k)C_{n-k}$$

where we are abusing notation by using C_k and C_{n-k} to now denote valid configurations and not simply the number of such configurations. Thus, we get (using the first strategy to break this down into a sum depending on what the value of k is):

$$C_{n+1} = C_0C_n + C_1C_{n-1} + C_2C_{n-2} + \cdots + C_{n-1}C_1 + C_nC_0 = \sum_{k=0}^{n} C_kC_{n-k}.$$

This recursive identity, together with the initial value $C_0 = 1$, thus gives one characterization of the Catalan numbers. Note that we get

$$C_4 = C_0C_3 + C_1C_2 + C_2C_1 + C_3C_0 = 5 + 2 + 2 + 5 = 14,$$

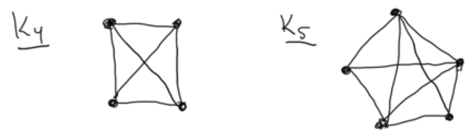
so not a huge number, but

$$C_5 = C_0C_4 + C_1C_3 + C_2C_2 + C_3C_1 + C_4C_0 = 14 + 5 + 2 \cdot 2 + 5 + 14 = 42$$

is already quite a bit larger. Nevertheless, we have found a way to determine these numbers concretely. (Later we will use the fourth strategy to find an explicit non-recursive formula for C_n.)

Lecture 5: More on Counting

Warm-Up 1. The *complete graph* K_n is the graph on n vertices obtained by drawing a single edge between each pair of vertices:



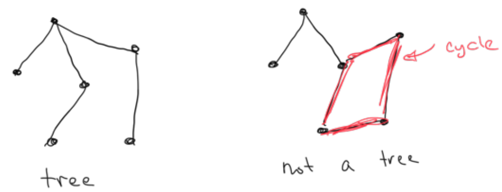
(The term "complete" comes from the fact that all possible edges are included, at least if we do not allow multiple edges between the same vertices nor edges which connect a vertex to itself.) We determine the number of edges in K_n.

We can structure this count as choosing one endpoint of an edge, and then the other. There are n possible choices for the first endpoint, and then $n-1$ remaining for the other, which ostensibly gives $n(n-1)$ as the number of edges. However, this is an overcount since if an edge is drawn by picking vertex A as the first endpoint and vertex B as the second, the same edge would be drawn again by picking B as the first endpoint and A as the second. Thus, we have overcounted by a factor of 2, so the correct number of edges is $\frac{n(n-1)}{2}$.

Warm-Up 2. We determine the number of subsets of $[n] = \{1, 2, \ldots, n\}$ with an odd number of elements. Intuitively, it seems to make sense that there should be as many subsets with an odd number of elements as there are subsets with an even number, since why should the difference between and odd number vs and even number make any difference? Thus we can guess that exactly half of all subsets will have an odd number of elements, so the number should be $\frac{2^n}{2} = 2^{n-1}$, where we use the fact that there are 2^n many subsets of $[n]$. But to be precise we would have to give a better reason as to why there should be as many subsets with an odd number of elements as there are with an even number, which I will leave it to you to think about.

Here is another approach which illustrates the idea of "counting something else". Say we are constructing a subset with an odd number of elements. We go through each of $1, 2, 3, \ldots, n$ and decide whether or not to include or exclude that element in the subset we want. The key point is that whether or not to include/exclude n is *fully determined* by what we did with the first $n-1$ elements $1, 2, \ldots, n-1$: if after this point we have only chosen to include an even number of elements, then we are forced to include n, while if after this point we have included an odd number of elements already, then we should not include n. Thus, specifying a subset of $[n]$ with an odd number of elements amounts to the same thing as specifying an arbitrary subset of $[n-1] = \{1, 2, \ldots, n-1\}$, since as we said above, once this subset is specified, what should happen to n is fully determined. Since $[n-1]$ has 2^{n-1} subsets, we conclude that there are 2^{n-1} many subsets of $[n]$ with an odd number of elements. (So, instead of counting such subsets, we counted "something else", namely arbitrary subsets of $[n-1]$, and argued that this number should be the same as the one we want.)

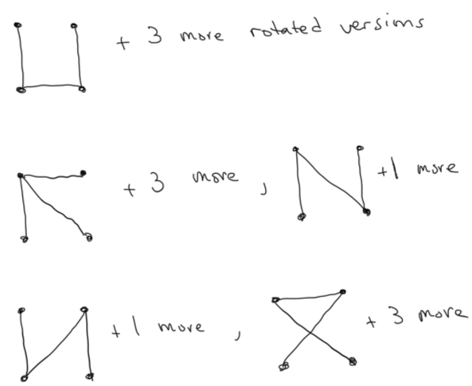
Counting trees. As another, more elaborate example of counting "something else", let us determine the number of trees on n vertices. A *tree* is a connected graph with no *cycles*, meaning no paths which end at the same place they started:



It turns out that an equivalent characterization is: a tree on n vertices is a connected graph with exactly $n-1$ edges. Trees are a key object of study in Math 308, and have numerous applications in computer science and elsewhere, but here we are only interested in counting such things.

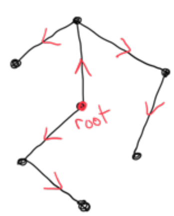
For $n = 3$ we have the following possible trees:

For $n = 4$ we have the following possibilities:

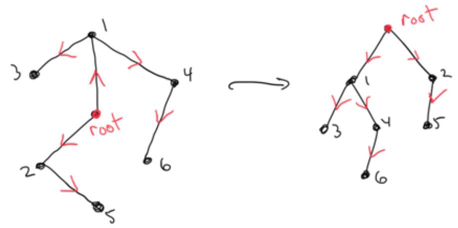


Let T_n denote the number of trees on n vertices; so far we have $T_3 = 3$ and $T_4 = 16$. After this point it comes difficult to draw out all the possibilities, in particular since (spoiler alert!) there are 125 trees on 5 vertices. So we need a better way to count them.

Instead, we will count something else. A *rooted* directed tree is a tree where some vertex has been chosen as the "root' and where we direct edges as moving away from the root:

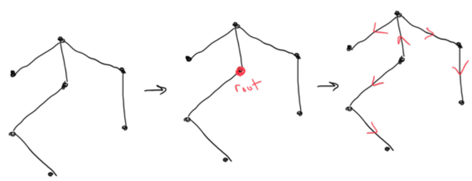


The idea is that such a tree can drawn instead as a "hierarchy" of nodes:

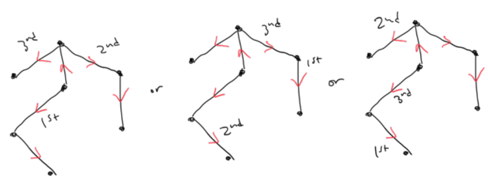


which is really where the term “tree” comes from. To find the number T_n of (undirected, unrooted) trees, we will instead count (in two ways!) the number of ways of *drawing* rooted directed trees.

Here is one way. At the end of our drawing, a rooted directed tree takes the shape of one of the regular trees contributing to T_n, so we can first count such things by starting with this final undirected, unrooted tree, then specifying the root, and then drawing in the directions on the edges:

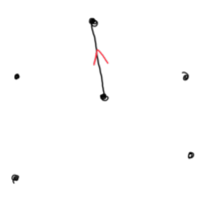


There are T_n choices for the shape of the underlying tree, then n choices for the root, and then $(n-1)!$ ways of drawing in the directed edges, since there are $n-1$ edges and we can draw in the required directions in any order:

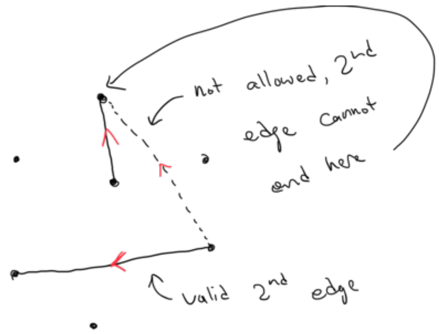


Thus there are $T_n n(n-1)!$ ways of drawing rooted, directed trees on n vertices.

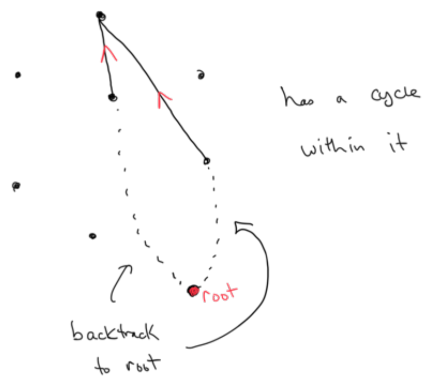
To count these another way, we instead start drawing in the directed edges from a picture of the vertices alone *without* having the underlying tree already in mind. Each edge has a starting point and an ending point, so for the first edge we draw we have n possible starting points and $n-1$ possible ending points:



For the next edge we can also pick any of our n vertices as the starting point, but the key fact is that we now only have $n-2$ possible ending points since the second edge *cannot* end at the same point at which the first edge ends:



If we *did* have something like this, we could "backtrack" from each of these edges back to the overall root of the tree, which would create a cycle in the end:



Trees have no cycles, so this cannot happen, and hence we cannot have two edges ending at the same point. For the third edge we still have n possible starting points, but not only $n-3$ possible endpoint points (we have to exclude the ending points of the first and second edges), and so on. Thus we get that the number of ways of drawing such trees is:

$$\underbrace{n(n-1)}_{\substack{\text{pick start/end}\\\text{for first edge}}}\ \underbrace{n(n-2)}_{\substack{\text{pick start/end}\\\text{for second edge}}}\ n(n-3)\cdots n(2)\ \underbrace{n(1)}_{\substack{\text{pick start/end}\\\text{for }(n-1)\text{-st edge}}} = n^{n-1}(n-1)!.$$

Since the values $T_n n(n-1)!$ and $n^{n-1}(n-1)!$ both count the same thing, they must be equal:

$$T_n n(n-1)! = n^{n-1}(n-1)!.$$

Thus we get $T_n = n^{n-2}$ as the number of trees on n vertices. Never doubt the power of counting something else!

n **choose** k**.** One more useful concept we will use again and again arises when wanting to determine the number of ways of picking k from among n objects *without* regard to the order in which they are chosen. For example, if we are interested in picking two numbers from among $1, 2, 3, 4$ without caring about order, then if we pick 2 first and then 3, we should treat this the same as if we had picked 3 first and then 2. All we are about in the end is *which* things are chosen, not *how* they are chosen.

The number of ways of picking k from among n objects without regard to order is usually denoted by

$$\binom{n}{k}$$

and is pronounced "n choose k". The precise value is found by structuring our count as "choose the first thing, then choose the second, and so on", and to then deal with overcounting. There are n choices for the first thing we pick, then $n-1$ for the second thing, and so on until we have $n-(k-1)$ choices for the k-th thing we pick. This gives so far

$$n(n-1)(n-2)\cdots(n-(k-1))$$

ways of picking k from n objects where for now the order *does* matter. Using properties of factorials, this expression can more succinctly be written as

$$n(n-1)(n-2)\cdots(n-(k-1)) = \frac{n!}{(n-k)!}$$

Now, to discount the order in which these things were chosen, notice that the specific k objects which are chosen could have been picked in any of $k!$ many possible orders, so the expression above overcounts by this factor. Hence the value we want is:

$$\binom{n}{k} = \frac{\frac{n!}{(n-k)!}}{k!} = \frac{n!}{k!(n-k)!}.$$

As usual in this course, the precise value of this construction will not be as important as understanding how to use it effectively. The number $\binom{n}{k}$ is also known as a *binomial coefficient*, due to its appearance in the *Binomial Theorem*, which we will look at soon enough.

Back to subsets. The binomial coefficient $\binom{n}{k}$ has a standard interpretation in terms of counting sets: it equals the number of subsets of $[n] = \{1, 2, \ldots, n\}$ consisting of k elements. After all, to construct a k-element subset of $[n]$ all we do is simply pick the k from among n elements which it will contain, and the order in which they are chosen is irrelevant: $\{2, 3, 4\}$ is the same set as $\{3, 4, 2\}$. This observation allows us to finish off something we mentioned when first counting subsets a few lectures ago. We noted that the number of subsets of $[n]$, which is 2^n can also be determined by breaking this count down into separate pieces: count the number of subsets with 0 elements, the number with 1 element, the number with 2 elements, and so on. The total number of subsets is thus expressible as

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n}$$

according to this method. Since we already know that there are 2^n subsets, we get the identity

$$\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n$$

for free: both sides count the same thing, so they must be equal. Later we will see another way to derive this identity via the Binomial Theorem. The point is that this is not merely some random identity, but one which encodes within it two ways of counting subsets of $[n]$.

Back to K_n. Finally, one more observation. Note that

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

But the number $\frac{n(n-1)}{2}$ is one we saw already back in the first Warm-Up, as the number of edges in K_n. So, the natural question to ask is whether $\binom{n}{2}$ also has a natural interpretation in terms of this graph, or, why does it make sense that the the number of edges in K_n should be $\binom{n}{2}$? The point is that to specify an edge all we need to do is to specify its two endpoints, which can be chosen from any of the n possible vertices. The number of ways of picking these two endpoints $\binom{n}{2}$, and so this should indeed be the number of edges in K_n as well.

Lecture 6: Yet More on Counting

Warm-Up 1. Let us count the number of paths in an $n \times n$ grid from the lower left corner to the upper right corner, where we only allow paths which at each step either move right or up. The key observation is that in total there will be $2n$ steps overall (since we have to move horizontally n times and vertically n times overall), and that n of these will consist of rightward moves and n will be upward moves. The entire path is completely determined as soon as we specify which of the $2n$ many moves will be "up", since then all the remaining moves must be "right". So we are left choosing the n from $2n$ many possible steps which will be up, giving a total of $\binom{2n}{n}$ many paths.

Another way of saying this is that a path is characterized by a $2n$-character string of n R's and n U's (standing for "right" and "up" respectively), so for instance $RURURU$ or $RRUUUR$ in the $n = 3$ case, and there are $\binom{2n}{n}$ many such strings again due to the fact that we need only specific where the n U's are.

Warm-Up 2. We show that the number $\binom{2n}{n}$ even. The point is not to look at the actual numerical value of $\binom{2n}{n}$:

$$\binom{2n}{n} = \frac{(2n)!}{n!n!}$$

since it is not clear from this at all that the value should be even, let alone an integer! The point is that, after the first Warm-Up, we have a way of interpreting $\binom{2n}{n}$ combinatorially as the number of paths in an $n \times n$ grid, which we can now exploit.

Any path in an $n \times n$ grid has a "mirror" path obtained by reflecting it about the main diagonal. Or, said another way, we turn each "up" into "right" and each "right" into "up". Thus, the possible paths come in pairs, so there should be an even number of paths. But this number of paths is precisely $\binom{2n}{n}$, so $\binom{2n}{n}$ is even.

Back to Catalan. We now make use of the $\binom{n}{k}$ values in order to find an explicit formula for the n-th Catalan number C_n. We do this using the "path" interpretation of C_n, which is the number of paths in an $n \times n$ grid which never go above the main diagonal. We can count these paths using the difference:

$$C_n = (\# \text{ of all paths}) - (\# \text{ which go above the diagonal}).$$

Indeed, excluding the paths we don't want from the total number of paths should leave us with the number we do want. We saw above that the number of all paths is $\binom{2n}{n}$.

To count the number of paths we don't want, which are those which go above the main diagonal, we proceed as follows. Consider such a "bad" path, and look at the point at which it first goes above the diagonal:

Create a new path (in a different grid) which consist of the same path up until this point, but afterwards exchanges all the "rights" and "ups"; so, if after this point the original path goes, say up up right, the new one will go right right up. This new path will lie on an $(n-1) \times (n+1)$ sized grid, or one with $n-1$ colums and $n+1$ rows. Indeed, in the original path, up until the point at which we first go above the diagonal there would have been in total one more up than right (along the diagonal there have been an equal number of rights and ups), meaning that the remaining part of the path must have one more right than up since in total there are an equal number of rights and ups in the original path. After we exchange "right" and "up", the new path will have one more up than right, meaning that the corresponding grid must have one more row than column *after* the point where we first go above the main diagonal; up until this point there was also one more row than column (since the original path had one more up and right at this point), so in total there are two more rows than columns, and hence an $(n-1) \times (n+1)$ grid:

The point is that the number of paths in the $n \times n$ grid which go above the main diagonal is the same as the total number of paths in the new $(n-1) \times (n+1)$ grid, since the process above is reversible: given a path in the $(n-1) \times (n+1)$ grid, by exchanging "right" and "up" after the point at which we first go above the main diagonal we get a path in an $n \times n$ grid which goes above the main diagonal. This gives a *bijection* between the paths in an $n \times n$ grid which go above the main diagonal and the paths in an $(n-1) \times (n+1)$ grid, so there should be the same number of each. The total number of paths in an $(n-1) \times (n+1)$ grid is $\binom{2n}{n-1} = \binom{2n}{n+1}$ using the reasoning as in the first Warm-Up: we just have to specify which $n-1$ of the $2n$ many moves are "right", or equivalently which $n+1$ are "up". Hence the number of paths in an $n \times n$ grid which go above the diagonal is also $\binom{2n}{n-1}$, so the n-th Catalan number is:

$$C_n = (\text{\# of all paths}) - (\text{\# which go above the diagonal}) = \binom{2n}{n} - \binom{2n}{n-1}.$$

Some algebra using the explicit factorial expression for $\binom{n}{k}$ gives:

$$C_n = \frac{(2n)!}{n!n!} - \frac{(2n)!}{(n-1)!(n+1)!} = \frac{(2n)!(n+1) - (2n)!n}{(n+1)n(n-1)!n!} = \frac{(2n)!}{(n+1)n!n!} = \frac{1}{n+1}\binom{2n}{n}$$

as the explicit value of the n-th Catalan number.

Combinatorial proofs. Finally, we consider a few simple identities, where the focus now is on interpreting these *combinatorially*. For instance, the following holds:

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

Of course, if you use the factorial expression for both sides you can work out that they are indeed the same, but the point is that a better way to argue they are the same is to argue that both sides actually count the *same* thing, only in different ways. After all, we defined $\binom{n}{k}$ precisely in terms of counting something, so it would make sense that we can find a reason along the lines of what this actually counts to show that the identity above holds. In this case, the point is that choosing k things from n objects, as the left side counts, is precisely the same as *not* choosing the remaining $n-k$ things, or said another way, the same as "choosing" not to choose those $n-k$ things. That is, the k things which are to be chosen are fully determined by specifying the $n-k$ which should not be chosen, and the right side above gives the number of ways of doing so. This is what we would call a *combinatorial proof* of this identity, which is more satisfying than simply a proof via some formula manipulation.

As another example, we have:

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

The left side counts the number of ways of picking, say, k elements from the set $\{1, 2, \ldots, n\}$. So the question is whether we can argue that the right counts the exact same thing, only a clever way. Among the elements which are to be chosen, either n will be included or it won't be. If n is to be included, then the remaining $k-1$ things should from the set $\{1, 2, \ldots, n-1\}$, so that

$$\binom{n-1}{k-1} = \#\text{ of } k\text{-element subsets of } [n] \text{ which include } n.$$

If n is not to be included in such a subset, then all k things to be chosen should come from $\{1, 2, \ldots, n-1\}$, so

$$\binom{n-1}{k} = \#\text{ of } k\text{-element subsets of } [n] \text{ which do not include } n.$$

These together give all possible k-element subsets of $[n]$, so we get

$$\underbrace{\binom{n}{k}}_{k\text{-element subsets}} = \underbrace{\binom{n-1}{k-1}}_{\text{those containing } n} + \underbrace{\binom{n-1}{k}}_{\text{those not containing } n}$$

as claimed. Again, the point is that we can interpret both sides combinatorially as counting the same thing, so both sides should be equal.

Lecture 7: Binomial Theorem

Warm-Up. We justify the following identity:

$$\binom{k}{k} + \binom{k+1}{k} + \binom{k+2}{k} + \cdots + \binom{n}{k} = \binom{n+1}{k+1}$$

via a combinatorial interpretation. The right side counts the number of $(k+1)$-element subsets of $[n+1]$, and we claim the left side does as well, broken down according to the *largest* element of that subset. If the largest element in a $(k+1)$ element subset is to be ℓ, then the remaining k terms in this subset must come from $\{1, 2, \ldots, \ell - 1\}$, and there are $\binom{\ell-1}{k}$ ways of picking these elements. Thus, there are:

$$\begin{aligned}
&\binom{k}{k} \text{ k-element subsets whose largest term is } k+1,\\
&\binom{k+1}{k} \text{ k-element subsets whose largest term is } k+2,\\
&\binom{k+2}{k} \text{ k-element subsets whose largest term is } k+3,\\
&\quad\vdots\\
&\binom{n}{k} \text{ k-element subsets whose largest term is } n+1.
\end{aligned}$$

Adding these gives the number of all possible $(k+1)$-element subsets of $[n+1]$, which gives the required identity. (Note that there cannot be a $(k+1)$-element subset whose largest term is smaller than $k+1$, which is why $k+1$ is the first possible "largest term" we consider.)

Teams and captains. Here is a problem from discussion. We split up n people into two teams A and B, and then choose a "captain" from team A. In how many ways can this be done? As the solutions to the discussion problems shows, one way to express the answer is as the sum

$$\sum_{k=1}^{n} k\binom{n}{k},$$

where we choose some number of k people to go into team A in $\binom{n}{k}$ ways, and then pick one of these people to serve as captain in k ways; the sum occurs over the possible sizes k of team A. Alternatively, the answer is

$$n2^{n-1},$$

obtained by first picking the person who will serve as captain of team A in n ways, and then a subset of the remaining $n-1$ people to make up the rest of team A in 2^{n-1} ways. Since both expressions count the same thing, we get the identity

$$n2^{n-1} = \sum_{k=1}^{n} k\binom{n}{k}$$

as a result. We have thus given a *combinatorial proof* of this identity, which means a justification in terms of interpreting both sides as certain counts.

Binomial Theorem. But there is another way in which we can derive the identity above, using the *Binomial Theorem*, which tells us how to express $(x+y)^n$ in general as a sum analogous to

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

Consider the problem of multiplying out

$$\underbrace{(x+y)(x+y)\cdots(x+y)}_{n \text{ times}}.$$

Each term we get in this expansion will involve some power of x and some power of y, and in fact we get terms like $x^k y^{n-k}$ for $k = 0, 1, \ldots, n$ since the exponents have to add up to n because we are multiplying n terms originally.

The only thing left is to figure out the coefficient of $x^k y^{n-k}$. But we get one such term precisely from choosing the k factors in $(x+y)^n$ from where the x will come, and then we are forced to pick the y term from the remaining factors. Thus, there should be as many ways of getting $x^k y^{n-k}$ as there are ways of choosing the k factors which give x, and there are $\binom{n}{k}$ ways to do this. So we get

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

as the required expansion, which is the statement of the Binomial Theorem.

Now, taking a derivative with respect to x gives:

$$n(x+y)^{n-1} = \sum_{k=1}^{n} \binom{n}{k} k x^{k-1} y^{n-k},$$

and setting $x = 1 = y$ gives

$$n2^{n-1} = \sum_{k=1}^{n} k \binom{n}{k}$$

as desired. Hopefully you will agree, however, that justifying this identity "combinatorially" as we did before is much more enlightening that deriving it via some formula manipulation. Also note that if we set $x = 1 = y$ in the statement of the Binomial Theorem we get

$$2^n = \sum_{k=0}^{n} \binom{n}{k},$$

which is also something for which we've also already seen a combinatorial interpretation in terms of counting subsets of $[n]$.

Other combinatorial identities. Setting $x = 2, y = 1$ in the Binomial Theorem gives

$$3^n = \sum_{k=0}^{n} \binom{n}{k} 2^k,$$

so it is natural to seek a combinatorial interpretation of this identity. Here is one. Suppose we have n people, to each of which we will give either a red hat, white hat, or blue hat. Then 3^n is the number of ways in which these hats can be distributed, and we claim that the right side of the identity above counts the same thing.

Indeed, choose first the k people who will receive either a red hat or a white hat in $\binom{n}{k}$ ways; everyone else is then forced to receive a blue hat. Then for these k people, we can choose the subset among them who will receive a red hat in 2^k ways, and all others among these k will thus receive a white hat. Finally, we sum up over the possible values of k to get the expression on the right.

Another example. We justify the fact that

$$\binom{2n}{n} = \sum_{k=0}^{n} \binom{n}{k}^2.$$

The left side counts the number of ways of picking a committee of n people from among $2n$ people. The right sides does as well; to see this, let us first order our $2n$ people:

$$1, 2, \dots, n, n+1, \dots, 2n.$$

Any committee of n people will consist of sum people from the first half $1, 2, \dots, n$ and some from the second half $n+1, \dots, 2n$. In particular, if we choose k people from the first half of n people, then we must pick the remaining $n-k$ from the second half of n people. Thus we have

$$\binom{n}{k}\binom{n}{n-k}$$

many ways of forming the required committee when k will come from the first half, and summing up over the possible values of k gives

$$\binom{2n}{n} = \sum_{k=0}^{n} \binom{n}{k}\binom{n}{n-k}.$$

The fact that $\binom{n}{n-k} = \binom{n}{k}$ then gives the required identity.

In a similar way, we also have

$$n\binom{2n-1}{n-1} = \sum_{k=1}^{n} k\binom{n}{k}^2$$

where the difference is that now we pick a "leader" of the committee which is chosen in the scenario above, but where we require that his leader come from the first half $1, 2, \dots, n$ of all possible $2n$ people. On the right side, we again pick k people from $1, 2, \dots, n$ to be on the committee, then $n-k$ from the second half, and then the "leader" from the k chosen in the first step, giving

$$k\binom{n}{k}\binom{n}{n-k} = k\binom{n}{k}\binom{n}{k}$$

possible choices. Then summing over the possible values of k (k must be at least 1 because the leader must come from the first half) gives the right side of the expression above. On the left side, we first pick the person will serve as leader from among $1, \dots, n$ in n ways, and $n-1$ people from the remaining $2n-1$ people to make up the rest of the committee.

Hopefully you'll agree that these "combinatorial interpretations" of such identities is much more enlightening than deriving them via formula manipulations alone!

Lecture 8: Compositions

Warm-Up. We generalize the Binomial Theorem by determining the expression for $(x_1+\cdots+x_k)^n$ as a sum of monomials involving powers of the x_i, meaning we want terms of the form $x_1^{a_1}x_2^{a_2}\cdots x_k^{a_k}$ The answer we give is known as the *Multinomial Theorem.* For instance:

$$(x+y+z)^3 = x^3 + 3x^2y + 3x^2z + 3xy^2 + 6xyz + 3xz^2 + y^3 + 3y^2z + 3yz^2 + z^3.$$

The first observation is that the exponents $a_1, \dots, a_k$ we use are nonnegative and must add up to n, which comes from the fact that we have n factors total in

$$(x_1 + \cdots + x_k)^n = \underbrace{(x_1 + \cdots + x_k)\cdots(x_1 + \cdots + x_k)}_{n \text{ times}}$$

and we are picking a single x_i term from each.

So, the expression we want is of the form

$$(x_1 + \cdots + x_k)^n = \sum_{\substack{a_1+\cdots+a_k=n \\ a_1,\ldots,a_k \geq 0}} (\text{coefficient}) x_1^{a_1} \cdots x_k^{a_k}$$

and we must determine the coefficient needed, which is the number of times the specific monomial $x_1^{a_1} \cdots x_k^{a_k}$ appears when multiplying out the left-hand side. To take the

$$(x+y+z)^3 = (x+y+z)(x+y+z)(x+y+z)$$

example above as a guide, let us focus on the xz^2 term. This comes from picking one x and two z's in the factors above, which can be done in the following ways:

$$xzz, zxz, zzx$$

where in the first expression means that we pick x from the first factor, z from the second, and z from the third, and similarly for the second and third expressions. There are three such listings, which is why the coefficient of xz^2 in this expansion is 3. Now, the key fact is that this just amounts to forming "words" using the symbols x, z, z, where we care about the placement of x in relation to the z's, but we don't care about the order of the z's among themselves.

In general, to get $x_1^{a_1} \cdots x_k^{a_k}$, we must pick a_1 many x_1's, a_2 many x_2's, and so on, so we get a "word" like

$$\underbrace{x_1 \ldots x_1}_{a_1 \text{ times}} \underbrace{x_2 \ldots x_2}_{a_2 \text{ times}} \cdots \underbrace{x_k \ldots x_k}_{a_k \text{ times}}.$$

Thus, the coefficient we're looking for is the same as number of "words" which can be formed using these symbols, but where the order of each group of x_i's among themselves does not matter, and only the placement of different x_i's in relation to each other matters. We have n symbols altogether, and these can be listed in $n!$ ways. However, to deal with the over-counting effect we from the order the x_1's among all such listings, we must divide by $a_1!$ since the x_1's can be rearranged among themselves in this many ways, and then to deal with the over-counting which comes from the arrangement of the x_2's we must divide by $a_2!$, and so on for the other terms as well. Thus we get

$$\frac{n!}{a_1! a_2! \cdots a_k!}$$

such "words" in total, and this is the coefficient we need. Hence:

$$(x_1 + \cdots + x_k)^n = \sum_{\substack{a_1+\cdots+a_k=n \\ a_1,\ldots,a_k \geq 0}} \frac{n!}{a_1! a_2! \cdots a_k!} x_1^{a_1} \cdots x_k^{a_k}.$$

The number $\frac{n!}{a_1! a_2! \cdots a_k!}$ is, as a result, often referred to as a *multinomial coefficient* and denoted $\binom{n}{a_1, a_2, \ldots, a_k}$, mimicking the notation for binomial coefficients, which are a special case.

Where we're headed. And so ends the first "chunk" of the course, which focused on basic strategies and techniques for counting. Now we move into the second chunk, where we specialize to look at the types of numbers arising in some specific types of problems, and hope to understand various relations between them. These are numbers related to, first, various types of *partitions*, and, later, *cycles* in permutations.

Looking ahead, the third chunk of the course will be devoted to what are called *generating functions*, which is the most in-depth thing we'll look at, and on which will spend the most time. Generating functions essentially give a way to do everything we've done so far in one full swoop from a new perspective, which will also help to answer questions we have not been easily able to thus far. Take note, however: generating functions is often where the intensity of the course seems to rapidly increase, but the payoff will be well worth it.

Distribution problems. Consider the problems of distributing n toys among k children. We can put various spins on this problem by specifying what we treat as "identical" and what we treat as "distinct":

- We can treat the toys as identical and the children as distinct. By this we mean that we make no distinction between the toys, so it does no matter which toy each child gets, only how many toys each child gets. But, we do make a distinction between the children, so that child A getting say more toys than child B is different than B getting more toys than A.
- We can treat the toys as distinct but the children as identical. In this case it does matter which toys go to which children, but *not* which child gets what group of toys. For instance, child A getting toys $1, 2$ and B getting toys $3, 4$ is different than A getting $1, 3$ and B getting $2, 4$, but is not different than A getting $3, 4$ and B getting $1, 2$. So, what matters is which toys end up being grouped together, but not the order in which these groupings are distributed.
- We can treat the toys and children as distinct. So, it matters which toys are grouped together and how exactly they are distributed: A getting toys $1, 2$ and B getting $3, 4$ is different than A getting $3, 4$ and B getting $1, 2$ in this case.
- We can treat the toys and children as identical. So, it does not matter which toys are grouped together, nor the order in which the groupings are distributed.

Each of these scenarios will lead to a certain type of number of interest. Each will give a different perspective on the limits and usefulness of "counting", but of course there will be some relation between them, as expected since they all came from essentially the same type of problem, only slightly modified in each case.

Compositions. If we distribute identical toys to distinct children, then all that matters is the number of toys each child gets. (For now let us assume that each child gets at least one toy.) A key simplification is in realizing that we can express such a distribution as a sum of positive integers; for instance, with $n = 8$ toys distributed among $k = 3$ children, we have

$$\begin{aligned} 8 &= 5 + 2 + 1 \\ &= 2 + 1 + 5 \\ &= 2 + 3 + 3 \\ &= 3 + 3 + 2 \\ &= 4 + 2 + 2 \end{aligned}$$

as some possibilities, where the first part of each sum indicates the numbers of toys the first child gets, the second part is the number the second child gets, and the third part the number for the third child. Note that $5+2+1$ and $2+1+5$ are treated as different here, since there is a difference between the first child getting 5 toys versus the first child getting 2 toys; in other words, the children are distinct. But, it doesn't matter which 5 toys make up the 5 the first child gets in the first case, or which toys make up the 2 the second child gets, since the toys are identical.

Such an expression is a called a composition of 8. In general, a *composition* of n is a way of expressing n as a sum of positive integers, where the order of the integers matters. Each integer used is called a *part* of the composition, so the compositions of 8 above each consist of 3 parts. The problem of distributing n identical toys among distinct children is thus the same as the problem of determining compositions of n.

Stars and bars. The compositions of 5 into 3 parts are:

$$3+1+1, 1+3+1, 1+1+3, 1+2+2, 2+1+2, 2+2+1,$$

so there are 6 of them. In general, however, it will be quite tedious to list out all allowed compositions if we want to determine how many there are. But, we can actually determine the explicitly the number of compositions of n into k parts by phrasing the problem in a different way (i.e. counting something else) as follows.

Imagine we list all n toys, which we will indicate by dots, or "stars":

$$\underbrace{\bullet \quad \bullet \quad \bullet \quad \cdots \quad \bullet}_{n \text{ total}}$$

We can then, starting from the left, indicate the first part of a given composition by putting a "bar" after the required number of dots, then indicate the second part by putting another bar, and so on:

$$\underbrace{\bullet \quad \bullet \quad \bullet}_{3} \quad | \quad \underbrace{\bullet \quad \bullet}_{2} \quad | \quad \cdots$$

For instance, the compositions of 5 into 3 parts would be drawn as:

$$\begin{array}{cl}
\bullet \; \bullet \; \bullet \; | \; \bullet \; | \; \bullet & 3+1+1 \\
\bullet \; | \; \bullet \; \bullet \; \bullet \; | \; \bullet & 1+3+1 \\
\bullet \; | \; \bullet \; | \; \bullet \; \bullet \; \bullet & 1+1+3 \\
\bullet \; | \; \bullet \; \bullet \; | \; \bullet \; \bullet & 1+2+2 \\
\bullet \; \bullet \; | \; \bullet \; | \; \bullet \; \bullet & 2+1+2 \\
\bullet \; \bullet \; | \; \bullet \; \bullet \; | \; \bullet & 2+2+1
\end{array}$$

We call this a "stars and bars" representation of a composition.

So, if we now consider compositions of n into k parts, the key realization is that this requires inserting $k-1$ bars in among the $n-1$ possible gaps occurring between "stars". So, the number of compositions of n into k parts is the number of ways in which this can be done, which is the number of ways of choosing from the $n-1$ gaps the $k-1$ locations where we will put a bar, and this is thus

$$\binom{n-1}{k-1} = \text{number of compositions of } n \text{ into } k \text{ parts.}$$

When $n = 5$ and $k = 3$, this gives $\binom{5-1}{3-1} = \binom{4}{2} = 6$, agreeing with the number we found via a brute force listing. In summary, the number of ways of distributing n identical toys to k distinct children, assuming each child gets at least one toy, is $\binom{n-1}{k-1}$.

Weak compositions. But now suppose we allow for the possibility that a child can get zero toys. In other words, we allow for the possibility of using 0 as a part when we express n as a sum of, now, nonnegative integers:

$$8 = 4 + 2 + 0 + 1 + 0 + 3.$$

Such an expression, where we write n as a sum of nonnegative integers, is called a *weak composition* of n. Among the weak compositions are all the ordinary compositions, so there should be more weak ones in addition to the ones we counted above.

We can again determine the number of weak compositions of n into k parts explicitly using "stars and bars" representations. For a weak composition we thus allow for the possibility that two bars might occur next to each other with no star in between; for instance,

$$\bullet \quad | \quad | \quad \bullet \quad \bullet \quad | \quad | \quad | \quad \bullet \qquad 1 + 0 + 2 + 0 + 0 + 1$$

is a weak composition of 4 into 6 parts. Again this amounts to choosing locations for the $k-1$ bars, but now there are more allowable locations. In total we will have $n + k - 1$ many "symbols" to arrange, coming from n stars and $k - 1$ bars. Imagine we visualize the locations for these $n + k - 1$ symbols as underscores:

$$\underbrace{\_ \ \ \_ \ \ \_ \ \ \cdots \ \ \_}_{n+k-1 \text{total}}.$$

Some of these will be filled in with stars, and other with bars, so to specify a weak composition we need only specify which of these $n + k - 1$ locations will consist of a bar, and hence there are

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

possibilities. (Equivalently, we can specify which locations are the n starts, which gives the same answer since $\binom{m}{\ell} = \binom{m}{m-\ell}$.) Thus, the number of weak compositions of n into k parts is $\binom{n+k-1}{k-1}$, which is hence the number of ways to distribute n identical toys to k distinct children where we allow for the possibility that some children might get no toys.

Number of compositions. The possible compositions of 4, regardless of the number of parts, is

$$4, 3 + 1, 1 + 3, 2 + 2, 2 + 1 + 1, 1 + 2 + 1, 1 + 1 + 2, 1 + 1 + 1 + 1,$$

so 4 has 8 compositions in total. Using the answer we found above for the number of compositions of n into k parts, in general the number of compositions of n is

$$\binom{n-1}{1-1} + \binom{n-1}{2-1} + \cdots + \binom{n-1}{n-1}$$

where the first term counts the number of compositions into 1 part, the second is the number into 2 parts, and so on with the final term being the number of compositions into n parts. But we know the exact value of this sum, say by using either the Binomial Theorem or the method for counting subsets of $[n-1]$ where we break the count down according to the sizes of the subsets; we have:

$$\binom{n-1}{1-1} + \binom{n-1}{2-1} + \cdots + \binom{n-1}{n-1} = 2^{n-1}$$

total compositions of n. For $n = 4$ this gives the $2^{4-1} = 8$ we listed above.

Because the answer is a power of 2, we see that in fact there should be twice as many compositions of $n+1$ as there are of n in general. But this is something we can argue separately without making use of the 2^{n-1} answer, and we can view this as a recursive way to see why there are 2^{n-1} compositions in general. Indeed, we claim that given any composition of n, there are two compositions of $n+1$ we can construct from it, in a way which will end up giving all compositions of $n+1$. Suppose we have a composition of n:

$$n = k_1 + \cdots + k_t.$$

To obtain a composition of $n+1$, we can either add a new part which is just 1, or we can increase the existing final part by 1:

$$n+1 = k_1 + \cdots + k_t + 1 \quad \text{or} \quad n+1 = k_1 + \cdots + (k_t + 1).$$

For instance, performing the first operation gives the following compositions of 5:

$$4+1, 3+1+1, 1+3+1, 2+2+1, 2+1+1+1, 1+2+1+1, 1+1+2+1, 1+1+1+1+1$$

and performing the second gives:

$$5, 3+2, 1+4, 2+3, 2+1+2, 1+2+2, 1+1+3, 1+1+1+2.$$

We end up with all the compositions of 5 here, each listed exactly once, and indeed this is what happens in general: adding a new final $+1$ to the compositions of n give all the compositions of $n+1$ which have final part equal to 1, and increasing the final part of the compositions of n by one produce all compositions of $n+1$ which final part larger than 1. Any composition of $n+1$ falls into one of these two categories, so we get all of them. Furthermore, this procedure does not produce any over-counting, essentially because we begin with distinct compositions of n: if $n = k_1 + \cdots + k_t$ and $n = \ell_1 + \cdots + \ell_s$ are different compositions of n, then adding a final $+1$ or increasing the final part by 1 produces different compositions of $n+1$ because they still differ in the initial k's and ℓ's.

So, this shows that if a_n denotes the number of compositions of n, then it should be true that $a_{n+1} = 2a_n$, and together with the initial fact that $a_1 = 1$, via induction we can justify the fact that $a_n = 2^{n-1}$ in general.

Lecture 9: Set Partitions

Warm-Up 1. We determine the number of compositions of $2n$ into even parts, meaning compositions where each part is even, or in other words ways of writing $2n$ as a sum of positive even integers. But if we take such a composition, dividing everything by 2 produces a composition of n:

$$2n = 2k_1 + \cdots + 2k_t \rightsquigarrow n = k_1 + \cdots + k_t.$$

Every composition of n arises in this way, since given a composition of n we can go backwards and multiply each part by 2 to obtain a composition of $2n$ into even parts. Thus, the number of compositions of $2n$ into even parts is the same as the number of compositions of n, which is 2^{n-1}.

Warm-Up 2. Now we determine the number of compositions of n into an even number of parts. To be clear, here we are talking about having an even number of positive integers in total, without any restriction on what types of integers those can be. Intuitively, by "symmetry", the number of compositions into an even number of parts should be the same (maybe) as the number of compositions into an odd number of parts, since from the point of view of taking sums of positive integers, why should even vs. odd make a difference?

Indeed this is true. But we have to be careful about why, since there is no obvious way to turn a composition of n into an even number of parts into one which has an odd number parts. Certainly for some specific composition, say $4+3+2$ of 9, we can say "split the 2 into $1+1$ to get an even number of parts overall", but it is not clear how to phrase this type of procedure so that it works in general without making reference to the specific composition we were looking at. But, we know from some facts derived last time that the number of compositions of n into an even number of parts is given by

$$\binom{n-1}{2-1} + \binom{n-1}{4-1} + \binom{n-1}{6-1} + \cdots,$$

where the final term is $\binom{n-1}{n-1}$ or $\binom{n-1}{(n-1)-1}$ depending on whether n is even or odd, and where we break the number down into those which have 2 parts, 4 parts, 6 parts, etc. Similarly, the number of compositions of n into an odd number of parts is given by

$$\binom{n-1}{1-1} + \binom{n-1}{3-1} + \binom{n-1}{5-1} + \cdots$$

where again the explicit form of the final term depends on whether n is even or odd. Thus, what we claim is that

$$\sum_{\text{even } k \le n} \binom{n-1}{k-1} = \sum_{\text{odd } k \le n} \binom{n-1}{k-1}.$$

This we can see from a clever use of the Binomial Theorem, substituting $x=-1$ and $y=1$:

$$0 = ((-1)+1)^{n-1} = \sum_{k=0}^{n-1} \binom{n-1}{k} (-1)^k 1^{n-k} = \sum_{k=1}^{n} \binom{n-1}{k-1} (-1)^{k-1}.$$

Because of the $(-1)^{k-1}$,the even values of k contribute a term to be subtracted on the right, while the odd values of k contribute a term to be added, so

$$0 = -\sum_{\text{even } k \le n} \binom{n-1}{k-1} + \sum_{\text{odd } k \le n} \binom{n-1}{k-1},$$

and after rearranging we get our claim. Thus, there are as many compositions of n into an even number of parts as there are into an odd number of parts, so each of these should be half the number of total compositions, which is $\frac{1}{2}2^{n-1} = 2^{n-2}$.

Set partitions. Returning to our distribution problems, we now consider the problem of distributing n distinct toys among k identical children. To be clear about what we mean by "identical children", what we care about is which toys are grouped together, so toys $1, 2$ being given to the same child is different than toys $3, 4$ being given to that child, but we don't care about which child gets the group $1, 2$, or the group $3, 4$, and so on. In other words, the order in which are children are listed does not matter, but how the toys are grouped together does.

If we label our toys $1, 2, \ldots, n$, then this type of distribution produces a *set partition* of $[n]$, which is a decomposition of $[n]$ into disjoint subsets. For instance, the set partition of $[5]$ given by

$$\{1,2\}, \{3,5\}, \{4\}$$

corresponds to having 3 children, one of which gets toys $1, 2$, another gets toys $3, 5$, and the other gets toy 4. But, we can also write this partition as

$$\{4\}, \{1, 2\}, \{3, 5\}, \text{ or as } \{3, 5\}, \{4\}, \{1, 2\}$$

without changing anything since, again, the children are identical. Having $\{1, 4\}$ as a subset however would give rise to a different partition. In summary, the number of ways of distributing n distinct toys to k identical children (so that each child gets at least one toy) is equal to the number of set partitions of $[n]$ into k non-empty subsets.

Example. The partitions of [4] are as follows, arranged according to the number of subsets used:

$k = 1$	$k = 2$	$k = 3$	$k = 4$
$\{1, 2, 3, 4\}$	$\{1, 2\}, \{3, 4\}$	$\{1, 2\}, \{3\}, \{4\}$	$\{1\}, \{2\}, \{3\}, \{4\}$
	$\{1, 3\}, \{2, 4\}$	$\{1, 3\}, \{2\}, \{4\}$	
	$\{1, 4\}, \{2, 3\}$	$\{1, 4\}, \{2\}, \{3\}$	
	$\{1, 2, 3\}, \{4\}$	$\{2, 3\}, \{1\}, \{4\}$	
	$\{1, 2, 4\}, \{3\}$	$\{2, 4\}, \{1\}, \{3\}$	
	$\{1, 3, 4\}, \{2\}$	$\{3, 4\}, \{1\}, \{2\}$	
	$\{2, 3, 4\}, \{1\}$		

Thus [4] has 1 partition into one subset, 7 into two subsets, 6 into three subsets, and 1 into four subsets, giving $1 + 7 + 6 + 1 = 15$ partitions in total.

Stirling numbers. We denote by $S(n, k)$ the number of partitions of $[n]$ into k non-empty subsets, and call this a *Stirling number of the second kind*. (This begs the question: what are the Stirling numbers of the *first kind*? We will come back to these later.) So, $S(n, k)$ is the number of ways of distributing n distinct toys among k identical children, so that each child gets at least one toy. From the example above, we can see for instance that $S(4, 1) = 1, S(4, 2) = 7, S(4, 3) = 6$, and $S(4, 4) = 1$. The total number of partitions of n, which is thus the sum of the $S(n, k)$ for varying k, is denoted $B(n)$ and called a *Bell number*:

$$B(n) = \sum_{k=1}^{n} S(n, k).$$

So, for instance we have $B(4) = 15$ as the value of the fourth Bell number.

However, as opposed to the case of compositions, we will not in general have a nice, explicit formula for $S(n, k)$, nor for $B(n)$. Certainly we can find a nice formula in some special cases:

$$\begin{aligned} S(n, 1) &= 1, \text{ since there is only one } n\text{-element subset of } [n] \\ S(n, n) &= 1, \text{ since the only decomposition into } n \text{ subsets is all singetons } \{1\}, \{2\}, \ldots, \{n\} \\ S(n, n-1) &= \tbinom{n}{2}, \text{ since in this case we need one 2-element subset and the rest singletons,} \\ &\qquad \text{so we count the ways of forming the 2-element subset.} \end{aligned}$$

As a next best thing, however, we will see next time nice *recursive* identities for $S(n, k)$. And even later, we will see ways of relating $S(n, k)$ to other numbers of interest, in particular the still-to-be-defined mysterious Stirling numbers of the first kind.

Surjective functions. Let us return to our distribution problems, and consider now the problem of distributing n distinct toys among k distinct children. Now we care about everything: which toys are grouped together with which, and which child gets which grouping. But, if we ignore the "which child gets which grouping" for now, we have $S(n,k)$ many ways of grouping (i.e. partitioning) the toys. Given such a partition, we can now decide how to distribute them among the k distinct children. If we take our partition, we have k choices for which child gets the first subset, then $k-1$ choices for who gets the second subset, $k-2$ for the third, and so on, giving $k!$ many ways of distributing the k subsets. Or, said another way, we can now list these k subsets in any order (first child gets first subset, second gets second, etc), and there are $k!$ many such listings. Thus, the number of ways to distribute n distinct toys among k distinct children (each child getting at least 1 toy) is $k!S(n,k)$.

Here is an important reformulation of this same problem. A function $f : [n] \to [k]$ (i.e. *from* the set $\{1, 2, \dots, n\}$ and *to* the set $\{1, 2, \dots, k\}$) is *surjective* if every element of $[k]$ is actually attained as the value of f at some input: for any j in $[k]$, there exists i in $[n]$ such that $f(i) = j$. For such a surjective function to exist we definitely need $n \geq k$, and so can ask in this case how many surjective functions $[n] \to [k]$ there actually are? The answer is (drumroll) $k!S(n,k)$.

Indeed, given a surjective function, we can first form a partition of $[n]$ by grouping together those elements which are sent to the same element of $[k]$, and $S(n,k)$ counts the number of such groupings. Then, we can specify which subset is sent to which element of $[k]$ in $k!$ many ways, by picking one of the k elements for the value of the things in the first grouping under the function in question, then one of the remaining $k-1$ elements for the value of the elements in the second subset, and so on. Or, once we have our partition, we list them in the order where things in the first subset are sent to 1, things in the second subset are sent to 2, etc. Thus, $k!S(n,k)$ also gives the number of surjective functions $[n] \to [k]$, so this is the same problem as counting ways of distributing distinct toys among distinct children. This perspective on $S(n,k)$ in terms of counting functions will lead to some nice observations next time.

Lecture 10: Stirling Numbers

Warm-Up 1. We compute the explicit value of $S(n,2)$. A partition of $[n]$ into two nonempty subsets will consist of one subset and its *complement*, which is the set of all things not in the first subset:

$$\{\text{some subset}\}, \{\text{things not in first subset}\}.$$

So, at first glance, such a partition is determined by specifying the first subset, which can any of the $2^n - 2$ subsets of $[n]$ which are not empty and not all of $[n]$. (We must exclude $[n]$ since the complement of this is empty, so we would not get two non-empty subsets.) However, this leads to over-counting by a factor of 2 since if we had chosen

$$\{\text{subset } A\}, \{\text{things not in } A\}$$

we would get the same partition by instead choosing

$$\{\text{things not in } A\}, \{\text{subset } A\}.$$

Thus, we divide by 2 to get $S(n,2) = \frac{1}{2}(2^n - 2) = 2^{n-1} - 1$.

Warm-Up 2. We find a combinatorial interpretation of the following true identity:

$$6S(n,3) + 6S(n,2) + 3S(n,1) = 3^n.$$

We claim that both sides count the number of all functions $[n] \to [3]$. For the right side, to specify a given function we must pick one of 3 possible choices for where 1 is sent, one of 3 possible choices for where 2 is sent, and so on. This gives $3 \cdot 3 \cdots 3 = 3^n$ functions in total.

Now, the first term $6S(n,3) = 3!S(n,3)$ on the left side counts the number of *surjective* functions $[n] \to [3]$, as we saw last time. The second term $6S(n,2)$ counts the number of functions $[n] \to [3]$ which take on 2 values: we first partition our n inputs into two subsets in $S(n,2)$ many ways according to which inputs will be sent to the same thing, then we choose the 2 values we want to attain in $\binom{3}{2} = 3$ many ways, and finally we decide which subset gets send to which value in 2! many ways. Thus there are indeed

$$2!\binom{3}{2}S(n,2) = 6S(n,2)$$

functions $[n] \to [3]$ which take on 2 values. The final term $3S(n,1)$ counts the number of constant functions $[n] \to [3]$, that is functions which take on only one value. There are 3 of these, which comes from picking that one value. However, it will make sense to think of this as

$$1!\binom{3}{1}S(n,1) = 3,$$

where we group all inputs in $[n]$ into one subset in $S(n,1)$ ways, then pick the 1 value the function will attain in $\binom{3}{1}$ ways, and finally assign this value to the chosen subsets in 1! ways.

To summarize, both sides of $6S(n,3) + 6S(n,2) + 3S(n,1) = 3^n$ count the number of functions from $[n]$ to $[3]$, values which a function will actually attain, which for $[3]$ are either 3 values, 2 values, or 1 value. Note as a byproduct of this that, since we know $S(n,2) = 2^{n-2} - 1$ from the first Warm-Up, and since $S(n,1) = 1$, we can use this identity to find an explicit expression for $S(n,3)$ by isolating the $S(n,3)$ term on one side.

Counting functions. We can generalize the second Warm-Up to count functions $[n] \to [k]$ for any k. On the one hand, there are k^n many such functions, since each of the n possible inputs can take on one of k values, so we get $k \cdot k \cdots k = k^n$ ways of constructing such functions. On the other hand, we can count these functions by breaking the count down according to the number of values the function actually takes on, which we can any number from 1 to k.

Suppose a function takes on i values from $[k]$. There are $\binom{k}{i}$ many ways of choosing these i values. Then, our set $[n]$ of inputs can be partitioned into i subsets (grouping together those which will take on the same value) in $S(n,i)$ many ways. And finally, we have $i!$ many ways of assigning the i chosen values to the i partitioned subsets, so we get

$$i!\binom{k}{i}S(n,i)$$

many functions $[n] \to [k]$ which take on i values. Summing these over the possible values of i gives

$$1!\binom{k}{1}S(n,1) + 2!\binom{k}{2}S(n,2) + \cdots + k!\binom{k}{k}S(n,k) = \sum_{i=1}^{k} i!\binom{k}{i}S(n,i)$$

functions in total, and thus we must have

$$\sum_{i=1}^{k} i!\binom{k}{i}S(n,i) = k^n$$

as a result. Again to be clear, both sides count functions from $[n]$ to $[k]$, only that on the left we break the count down according to the number of values a function can take on. It will be nicer to write the sum on the left as going all the way to $i = 0$, which we can do by interpreting $\binom{k}{i}$ to be 0 when $i > k$, which makes sense since there should be zero ways of choosing i things from k objects when there are more things to be chosen than there are objects from which to choose.

A polynomial identity. So, we have the identity

$$\sum_{i=1}^{n} i!\binom{k}{i} S(n,i) = k^n.$$

Now, writing out the $\binom{k}{i}$ in terms of factorials gives:

$$i!\binom{k}{i} = i!\frac{k!}{i!(k-i)!} = \frac{k!}{(k-i)!} = k(k-1)(k-2)\cdots(k-i+1).$$

Notice that this still makes sense when $i > k$ as well: in this case one of the factors on the right will look like $(k-k)$, which will make the entire right side equal zero. Thus we have

$$\sum_{i=1}^{n} S(n,i)k(k-1)\ldots(k-i+1) = k^n,$$

which we are saying is true for any positive integer k. But consider now the two polynomials

$$\sum_{i=1}^{n} S(n,i)x(x-1)(x-2)\cdots(x-i+1) \quad \text{and} \quad x^n$$

for a real variable x. To be clear, the polynomial on the left looks like:

$$S(n,1)x + S(n,2)x(x-1) + S(n,3)x(x-1)(x-2) + \cdots + S(n,n)x(x-1)\cdots(x-n+1),$$

which is overall a polynomial of degree n. The upshot is that we have shown this polynomial of degree n and the polynomial x^n also of degree n agree on any positive integer input, and this in fact means that they must be the same polynomial! Indeed, if $p(x)$ and $q(x)$ are two polynomials of degree n such that $p(n) = q(n)$ for all positive integers n, then the polynomial $p(x) - q(x)$ of degree at most n has infinitely many roots, which can only happen if it is the zero polynomial since a nonzero polynomial of degree at most n can only have at most n roots.

Thus we get that

$$\sum_{i=1}^{n} S(n,i)x(x-1)(x-2)\cdots(x-i+1) = x^n$$

not just for positive integer x but in fact for all real x. The product $x(x-1)(x-2)\cdots(x-i+1)$ is often denoted by $(x)_k$, and is the part of $\binom{x}{i}$—at least when x is a positive integer—where we DO take order into consideration, before figuring out how to ignore it afterwards integer. So, we can write the identity above a bit more concisely as the following, where we now just call the indexing variable k instead of i:

$$\sum_{k=1}^{n} S(n,k)(x)_k = x^n.$$

This strange-looking identity is one will come back to when deriving the *other* type of Stirling number we'll consider, and will provide a link between two seemingly different but related concepts.

One more rewrite: it will at times be convenient to start the sum at $k = 0$, which we can do if we interpret $S(n, 0)$ as 0, which makes sense since there are zero ways to partition $[n]$ into 0 non-empty subsets. So, we can also take

$$\sum_{k=0}^{n} S(n,k)(x)_k = x^n$$

as the expression for the identity we've derived.

Finding $S(n, k)$ recursively. As with the brief comment made at the end of the second Warm-Up, we could use the identity

$$1!\binom{k}{1}S(n,1) + 2!\binom{k}{2}S(n,2) + \cdots + k!\binom{k}{k}S(n,k) = k^n$$

obtained above to give a recursive way to compute $S(n, k)$ in general from $S(n, \ell)$ for $\ell < k$ by isolating the $S(n, k)$ term on one side. But, this will get quite tedious quickly, so we seek another recursive approach to computing these numbers.

The point is that we can compute $S(n, k)$ alternatively by focusing on the subset in a given partition to which n itself will belong. If we partition $[n]$ into k nonempty subsets, there are two possibilities: n occurs as the only element in a singleton $\{n\}$, or n is part of some subset of size larger than 1. In the first case, we have $\{n\}$ as one subset in our partition, and the rest of the partition must consist of $k-1$ subsets using the numbers $1, 2, \ldots, n-1$; this amounts to partitioning $[n - 1]$ into $k - 1$ subsets, which can be done in $S(n - 1, k - 1)$ many ways.

In the second case, when n is part of a subset of size larger than 1, we can begin by partitioning all other elements $1, 2, \ldots, n - 1$ into k subsets, and then inserting n into one of these existing subsets: the first step can be done in $S(n - 1, k)$ many ways, and then we have k choices for the subset in which to include n, giving $kS(n - 1, k)$ possibilities in this case. Thus overall we get:

$$S(n,k) = \underbrace{S(n-1,k-1)}_{\{n\} \text{ as one subset}} + \underbrace{kS(n-1,k)}_{n \text{ part of a larger subset}}$$

So, for instance, if we wanted to compute say $S(6, 4)$, we could use

$$S(6,4) = S(5,3) + 4S(5,4),$$

and then $S(5, 3) = S(4, 2) + 3S(4, 3)$, and finally the fact $S(n, 2) = 2^{n-1} - 1$ in general and that $S(n, n - 1) = \binom{n}{2}$ in general (twice!) to finish it off. In theory, then, we could compute any value of $S(n, k)$ given enough time and dedication!

Lecture 11: Integer Partitions

Warm-Up 1. We find a combinatorial interpretation of the identity

$$\sum_{k=1}^{n} (-1)^k k! S(n,k) = (-1)^n,$$

which comes from setting $x = -1$ in the identity

$$\sum_{k=1}^{n} S(n,k)(x)_k = x^n$$

we saw last time, where we make use of the fact that

$$(-1)_k = \underbrace{(-1)(-1-2)(-1-3)\cdots(-1-k+1)}_{k \text{ terms}} = (-1)^k k!.$$

This identity was derived via counting functions, so our interpretation will be phrased along these lines as well.

The sum on the left looks like:

$$-1!S(n,1) + 2!S(n,2) - 3!S(n,3) + 4!S(n,4) + \cdots$$

where the point is that all the terms for odd k are negative and those even k are positive. So, rewrite this sum by grouping positive and negative terms together as:

$$\sum_{\text{even } k \leq n} k!S(n,k) - \sum_{\text{odd } k \leq n} k!S(n,k) = (-1)^n.$$

We have seen that each $k!S(n,k)$ term counts surjective functions with image of size k, so the point is that the first sum above counts all possible surjective functions on $[n]$ which image of even size, while the second sum counts all surjective functions on $[n]$ with image of odd size.

The upshot, then, is that these two numbers differ by exactly 1; it might be that the first sum is larger than the second, or that the second is larger, but their difference is ± 1 regardless. To be more precise, for even n, there is one more surjective function on $[n]$ with an even-sized image than there is with one odd-sized image, while for n odd there is one more with an odd-sized image than with an even-sized image.

Warm-Up 2. Recall that the Bell numbers $B(n)$ give the number of all partitions of $[n]$. We justify the fact that

$$B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(k).$$

By convention, we set $B(0) = 1$, so that the right side does make sense for all $k = 0, \ldots, n$. (This is purely for the sake of convenient notation: there are of course no partitions of the empty set into nonempty subsets, but regardless we declare the 0-th Bell number to be 1.)

As with many other similar recursions we've now seen, the key is to focus on where the final element $n+1$ goes in a given partition of $[n+1]$: it can go into a singleton, or a subset of size 2, or of size 3, or of size 3 etc. Say we pick the k elements among $1, 2, \ldots, n$ which will NOT go into the same subset as $[n+1]$. (So, k here can be zero.) There are $\binom{n}{k}$ ways of doing so, and then these k elements can be partitioned in anyway whatsoever in $B(k)$ many ways. This gives $\binom{n}{k}B(k)$ possible partitions of $[n+1]$ where k elements are not in the same subset as $[n+1]$, and summing over the possible values of k gives the stated identity.

Alternatively, we can phrase this as saying to pick the k elements from $[n]$ which *do* go into the same subset as $n+1$. Then the remaining $n-k$ elements can be partitioned in $B(n-k)$ many ways, so we get

$$B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(n-k)$$

as the recursive expression in this case. However, by reindexing and making use of the fact that $\binom{n}{k} = \binom{n}{n-k}$, we can see that this sum is the same as the one we had before.

Integer partitions. We finally come to our final distribution problem, that of distributing n identical toys among k identical children, so that each child gets at least one toy. This is quite similar to the case of compositions, only that in this case we do not care about the order in which we write the integers in a certain sum:

$$1+3+2 \text{ is the same as } 3+2+1 \text{ is the same as } 2+3+1$$

since it does not matter which child gets 1 toy, or 2 toys, or 3 toys, since the children are all identical. These distinctions *did* matter for compositions.

An *integer partition* of n is a way of writing n as a some of positive integers, where the order of the integers does not matter. That being the case, we make the convention to always write the integers in a partition in *decreasing* (or really "non-increasing") order. For instance, the possible partitions of 5 are:

$$5, 4+1, 3+2, 3+1+1, 2+2+1, 2+1+1+1, 1+1+1+1+1.$$

We set $p(n)$ to be the number of integer partitions of n, and $p_k(n)$ to be the number of partitions of n which consist of k parts. So, $p(5) = 7$ and $p_3(5) = 2$ for instance. We will often be interested in other restricted types of partitions, and will use notation similar to these in general: $p_{odd}(n)$ for the number of partitions consisting of all odd parts, $p_{distinct}(n)$ for the number of partitions consisting of distinct parts, $p_{\leq k}(n)$ for the partitions with at most k parts, and so on.

Taking a step back. So, we've now seen three types of numbers arising from distribution problems: compositions, set partitions, and integer partitions. These are all different, but of course there are some relations between them: integer partitions are like compositions except we forget about order, the sizes of subsets in a set partition is given by an integer partition, and so on.

But, note that we actually have, in a sense, less and less information about these numbers as we move from one to the next. For compositions we had explicit formulas:

$$\binom{n-1}{k-1} \text{ and } 2^{n-1}$$

for the number with k parts and for the number in total respectively. For set partitions we don't have explicit formulas in general (we do in special cases like $S(n, n-1)$), but we *do* have recursive identities like

$$S(n,k) = S(n-1,k-1) + kS(n-1,k) \text{ and } B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(k),$$

which are almost as nice as having an explicit formula.

However, with integer partitions we have none of these in general: no explicit formula, and no nice recursive identities. Yes in some special scenarios we will find some valid recursions, but nothing that will work in general as we had for set partitions.

Young diagrams. So, it might seem at first as if integer partitions are difficult things to learn anything about, but actually there are many interesting things we can say about them. What we need, then, are new ways of working with integer partitions, which go beyond the types of techniques we've developed thus far.

Here is a first interesting observation. Consider the integer partitions of 6, arranged as follows according to their largest part (recall that convention that we list the integers in a given partition in decreasing order):

$$6 = 6$$

$$
\begin{aligned}
&= 5+1 \\
&= 4+2 = 4+1+1 \\
&= 3+3 = 3+2+1 = 3+1+1+1 \\
&= 2+2+2 = 2+2+1+1 = 2+1+1+1+1 \\
&= 1+1+1+1+1+1.
\end{aligned}
$$

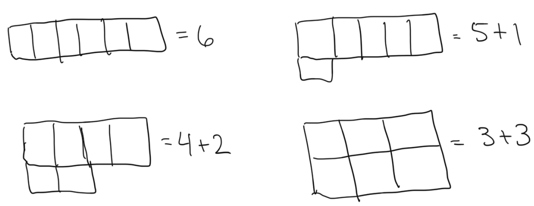
So, $p(6) = 11$. Now, there are 3 partitions here which consist of three parts $(4+1+1, 3+2+1, 2+2+2)$, so $p_3(6) = 3$. There are *also* 3 partitions whose largest part is 3, namely the ones in the fourth line. Next we have $p_4(6) = 2$, and also 2 partitions whose largest part is 4. Finally, there 3 partitions whose largest part is 2, and we can see that $p_2(6) = 3$ as well. So, at least in this example, it seems to be the case that the number of partitions consisting of k parts is the same as the number of partitions whose largest part is k. Coincidence?

No, of course not, or else we wouldn't be pointing this out! To show that this is true in general:

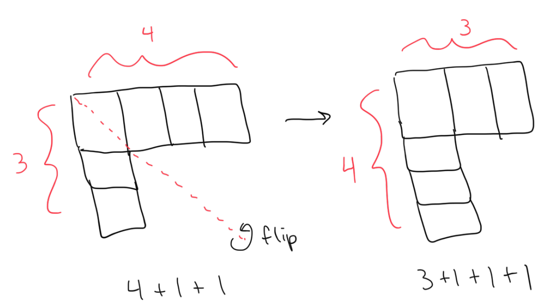
$$p_{k \text{ parts}}(n) = p_{\text{largest part } k}(n)$$

we need a way to turn a partition with k parts into one with largest part k, and vice-versa. It is not at all obvious how to do this if we focus only on the integer sum expressions, but the great thing is that we actually have a very nice way to interpret partitions *visually*.

The *Young diagram* (or *Ferrers shape*) of a partition of n is the picture consisting of n boxes overall arranged in rows, where the number of boxes in a given row is the size of the corresponding part, and the number of rows is the number of parts:



(The book uses the term "Ferrers shape" exclusively, which is fine, but I think "Young diagram" is nowadays a more common term for these, so that's what I'll mainly use.) The point is that such a picture allows us to better visualize certain aspects of partitions. For instance, the fact that the number of partitions of n with k parts is the same as the number where the largest part is k is just the observation that we can turn a Young diagram of the first type into a Young diagram for the second by looking at it "sideways", or more precisely by turning rows into columns and columns into rows:



If the first diagram has k rows (so k parts), the new one will have k columns, meaning that the first new row will have k boxes, which gives largest part k.

We can visualize this as "flipping" the diagram across the main diagonal, and the resulting partition is called the *conjugate* of the first. For instance, in the picture above, the first diagram represents the partition $4+1+1$ of 6, which is thus conjugate to the partition $3+1+1+1$ represented by the second diagram. The operation of taking conjugates is a bijection, so we get as many Young diagrams as conjugate Young diagrams, which is why $p_{k \text{ parts}}(n) = p_{\text{largest part } k}(n)$ for all n in general.

Final example. As we said earlier, in general we will not have a nice formula for $p(n)$. We do have something like

$$p(n) = \sum_{1 \leq k \leq n} p_k(n)$$

where we group partitions according to their number of parts, but this isn't so hopeful without a nice formula for $p_k(n)$, which again we do not have. But, here is an example of where we can find some nice type of recursion at least.

We determine the number of partitions of n where all parts are at least of size 2. This is the same as the number of Young diagrams where all rows have at least 2 boxes. The idea is that we can take the number $p(n)$ of all partitions of n, and *exclude* the ones we don't want, meaning partitions with at least one part equal to 1:

$$(\text{number with all parts at least } 2) = p(n) - (\text{number with 1 as a part}).$$

But if we consider the Young diagram for a partition of n which has at least row with one box, by removing one such row altogether we will be left with a Young diagram with one less box, so for a partition of $n-1$. Conversely, given any partition of $n-1$, by adding an extra part of 1 at the end, or visually by adding an extra row with a single box to the Young diagram, we get a partition of n with at least one part equal to 1, so

$$(\text{number of partitions of } n \text{ with 1 as a part}) = \underbrace{(\text{number of partitions of } n-1)}_{p(n-1)},$$

and hence

$$(\text{number of partitions of } n \text{ with all parts at least } 2) = p(n) - p(n-1).$$

So, by adding certain parts or including others, we can often relate certain partitions of one number to partitions of another. By taking conjugates we can see that $p(n) - p(n-1)$ is also the number of partitions of n where the two largest parts are equal, which comes from having all rows with at least 2 boxes in the original diagram.

Lecture 12: More on Partitions

Warm-Up. We argue that the following inequality holds:

$$p(1) + p(2) + \cdots + p(n) < p(2n).$$

The idea, as usual, is to figure out how we can take any partitions contributing to the left side and turn it into a partition contributing to the right, in a way so that different things on the left give different things on the right; this will give a non-strict inequality, and then we get the strict inequality by coming up with one thing on the right which doesn't arise via this process.

To get a feel for this, we can write out some possibilities for a small n, say $n = 4$. On the left we then have the partitions of $1, 2, 3$, and 4, which are:

$$\begin{aligned}&1\\&2 = 1 + 1\\&3 = 2 + 1 = 1 + 1 + 1\\&4 = 3 + 1 = 2 + 2 = 2 + 1 + 1 = 1 + 1 + 1 + 1\end{aligned}$$

So $p(1) + p(2) + p(3) + p(4) = 1 + 2 + 3 + 5 = 12$ in this case. Now, we want to turn each of these into a partition of 8 for the right side of the stated inequality. To get such a partition from 1, we can for instance add 7: $7 + 1$. To get a partition of 8 from the ones listed above for 2, we can add 6 to each: $6 + 2$ and $6 + 1 + 1$. And so on, to each partition of $k \leq 4$ we can add $8 - k$ to get the following partitions of 8:

$$\begin{aligned}&7 + 1\\&6 + 2, 6 + 1 + 1\\&5 + 3, 5 + 2 + 1, 5 + 1 + 1 + 1\\&4 + 4, 4 + 3 + 1, 4 + 2 + 2, 4 + 2 + 1 + 1, 4 + 1 + 1 + 1 + 1\end{aligned}$$

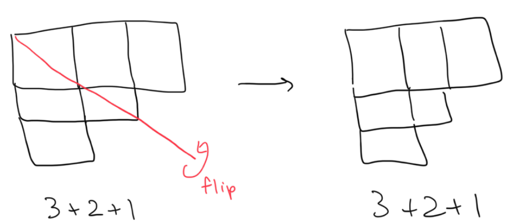
Note that the resulting partitions of 8 we get are all distinct: the one in the first row is the only one with a 7, those in the second rows are the only ones with 6, those in the third row have 5, and those in the last row have 4. So we get $p(1) + p(2) + p(3) + p(4) \leq p(8)$, and the inequality is strict since for instance 8 is a partition of 8 which does not arise in this way.

The idea works in general: to the partitions of 1 add $2n - 1$, to those of 2 add $2n - 2$, to those of 3 add $2n - 3$ and so on until finally to those of n we add $2n - n = n$. The resulting partitions of $2n$ are all different: if they came from a different $1 \leq k \leq n$ to start with, they are different since the largest part is different; and if they cane from the same k, they are different since they came from different partitions of k. Thus

$$p(1) + p(2) + \cdots + p(n) \leq p(2n),$$

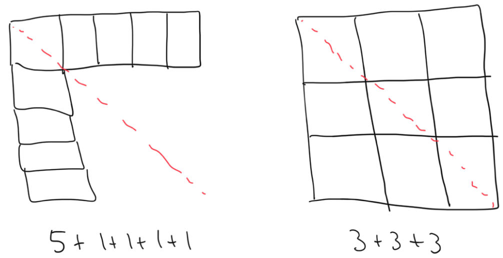
and since $2n$ for instance does not arise in this way (or the partition of all 1's), the inequality is actually strict.

Self-conjugate = distinct odd. Here is one more observation, which is difficult to see without the use of Young diagrams. Notice that for $n = 6$, we have the partition $3 + 2 + 1$. This actually equals its own conjugate, since flipping its Young diagram across the main diagonal gives the same diagram:



We say that $3+2+1$ is a *self-conjugate* partition, and this is in fact the only self-conjugate partition of 6. Now consider a seemingly different type of partition: those with distinct, odd parts. For 6 the only possibility is $5 + 1$, so 6 has one partition consisting of distinct odd parts.

For $n = 9$ we have two possible self-conjugate partitions: $5 + 1 + 1 + 1 + 1$ and $3 + 3 + 3$:

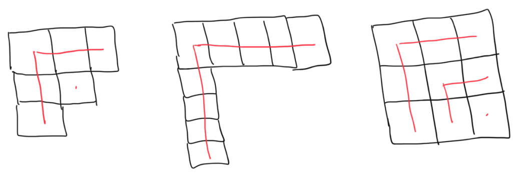


It turns out we also have two partitions into distinct odd parts: 9 and $5+3+1$. So, what are we getting at? Is it a coincidence that for these n the number of self-conjugate partitions agrees with the number of partitions into distinct odd parts? No, this is no coincidence, or else we wouldn't be mentioning it. To show that

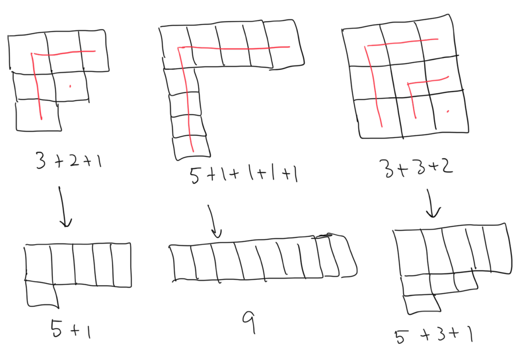
$$p_{\text{self-conjugate}}(n) = p_{\text{distinct,odd}}(n)$$

in general we thus we need a way to turn any one of the former type into one of the latter, as usual in a one-to- one manner.

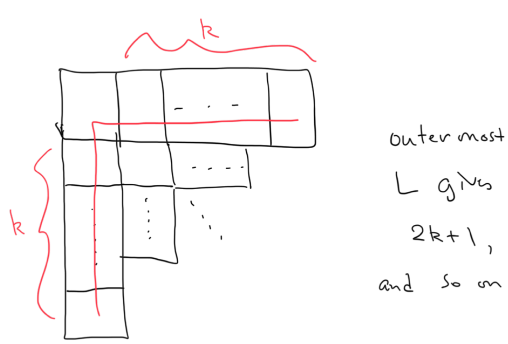
But Young diagrams help us out here again. Notice the following in the self-conjugate partitions of 6 and 9 drawn above:



These "outer L" figures all consist of an odd number of boxes, and as we move down to the next smaller such shape (in the $3+2+1$ and $3+3+3$ cases) we get smaller odd numbers. These outer L shapes can then be used to describe a partition in distinct odd parts: take the size of the outermost L as the first part, the next L as the next part, and so on:



This same procedure works in general. Take any self-conjugate partition of n:



The outermost L has as many things to the right of the initial box as it has below since the partition is self-conjugate, so this L has an odd number of boxes once we take the "initial" box into account as well. The next smaller L has an odd number of boxes for the same reason, and this odd number is smaller than the first because the L is smaller. And so on, we get a partition of n into distinct odd parts. Conversely, given a partition of n into distinct odd parts, we can use the boxes in each part to form an appropriate L shape, starting with largest part to get the outermost layer, and so on. This give a bijection between the self-conjugate partitions and those with distinct odd parts, so the number of each are the same.

A property still to come. Let us count one more thing, in cases where we can do so by brute-force. The partitions of 6 with consist of all distinct parts are:

$$6, 5+1, 4+2, 3+2+1$$

and the partitions of 6 which consist of all odd parts are:

$$5+1, 3+3, 3+1+1+1+1, 1+1+1+1+1=1,$$

so we have 4 of each. For $n = 7$ we have

$$7, 6+1, 5+2, 4+3, 4+2+1$$

for all distinct parts and

$$7, 5+1+1, 3+3+1, 3+1+1+1+1, 1+1+1+1+1+1+1$$

for all parts, so we again have the same number (5 in this case) of each. This is actually true in general:

$$p_{\text{distinct parts}}(n) = p_{\text{all odd parts}}(n),$$

but we need a new approach to see why this is. In this case, there is no obvious way of turning a Young diagram of one type into a Young diagram of the other, at least no way which will work for all n at once. What we need is more ways of working with partitions, and indeed more ways of working with combinatorial objects in general. We'll leave this as a teaser for now, but what follows below is the core idea of what will eventually give us a way to approach this problem.

An infinite product of infinite sums. Let us do something which seems totally random at first, but which, as we'll see, relates to what we've been doing in a crucial way.

Consider the following product:

$$(1+x+x^2+\cdots)(1+x^2+x^4+\cdots)(1+x^3+x^6+\cdots)(1+x^4+x^8+\cdots)\cdots \text{ and so on.}$$

This is a product of infinitely many factors, *each* of which is an infinite sum: the first factor is the infinite sum of powers of x, the second factor is the infinite sum of powers of x^2, the third factor uses powers of x^3, the fourth powers of x^4, and so on without end. If we imagine multiplying this all out, we get an expression which involves powers of x, but where each such power (except for x^0 and x^1) can be obtained in multiple ways: the only way to get $x^0 = 1$ is to take the "1" term from each factor and multiply those all together; the only way to get $x^1 = x$ is to take the x term from the first factor and then the 1 term from all the other factors; but, there are two ways to get x^2, namely take the terms $x^2, 1, 1, 1, \ldots$ or the terms $1, x^2, 1, 1, 1, \ldots$ from the factors respectively. So, when multiplying this out we get a $2x^2$ term overall:

$$(1+x+x^2+\cdots)(1+x^2+x^4+\cdots)(1+x^3+x^6+\cdots)\cdots = 1+x+2x^2+\cdots$$

Now, what will the coefficient of, say, x^5 end up being? We can get x^5 in the following ways, listed according to the terms we take from the first, second, third, etc factors in order:

$$\begin{gathered}1, 1, 1, 1, x^5, 1, \dots \\ x, 1, 1, 1, x^4, 1, \dots \\ 1, x^2, x^3, 1, 1, 1, \dots \\ x^2, 1, x^3, 1, 1, 1, \dots \\ x, x^4, 1, 1, 1, 1, \dots \\ x^3, x^2, 1, 1, 1, 1, \dots \\ x^5, 1, 1, 1, 1, 1, \dots\end{gathered}$$

To clarify the notation again, the first expression means we take 1 from each of the first four factors, x^5 from the fifth factor (the one made by taking powers of x^5), and then one from all the remaining factors. So, we get $7x^5$ in our multiplied-out expression:

$$(1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots) \cdots = 1 + x + 2x^2 + \cdots + 7x^5 + \cdots$$

Here is the punchline: 7 is precisely the number of partitions of 5! (That is an exclamation point, not a factorial.) Indeed, if you work out the first few remaining coefficients, you'll get:

$$(1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots) \cdots = 1 + x + 2x^2 + 3x^3 + 5x^4 + 7x^5 + \cdots$$

and the coefficients $1, 2, 3, 5, 7$ of x, x^2, x^3, x^4, x^5 respectively are precisely the number of partitions of $1, 2, 3, 4, 5$. We claim that this is true in general: the coefficient of x^n in this infinite product of infinite sums is $p(n)$, so that

$$(1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots) \cdots = \sum_{n=0}^{\infty} p(n)x^n$$

where we set the constant term $p(0)$ to be $p(0) = 1$. And so now we see why we considered this infinite product in the first place.

One hint as to why there should be a relation between partitions and this infinite product can be found in the observation that all the expressions above for getting x^5 involve exponents which add up to 5, so that these (nonzero) exponents form a partition of 5. But, we have to be careful about how exactly we relate partitions to exponents since so far there is nothing which seems to give $2 + 2 + 1$ for instance, and we also have say x^2x^3 appearing three times, albeit in different orders. Let us instead write the possible exponents occurring as:

$$\begin{gathered}x^{0(1)}, x^{1(1)}, x^{2(1)}, x^{3(1)}, \dots \text{ for powers of } x \\ x^{0(2)}, x^{1(2)}, x^{2(2)}, x^{3(2)}, \dots \text{ for powers of } x^2 \\ x^{0(3)}, x^{1(3)}, x^{2(3)}, x^{3(3)}, \dots \text{ for powers of } x^3\end{gathered}$$

and so on, so that for powers of x^m we have terms like x^{km}. The interpretation is then that k tells us how many times m should be used in the corresponding partition! For instance, for

$$1, 1, 1, 1, x^5, 1, \dots$$

in the $n = 5$ case, we should think of these as

$$x^{0(1)}, x^{0(2)}, x^{0(3)}, x^{0(4)}, x^{1(5)}, \dots$$

which says that we should take no 1's, no 2's, no 3's, no 4's, one 5, and then no other integer, which gives the partition 5 of 5. Similarly,

$$x^{1(1)}, 1, 1, 1, x^{1(4)}, 1, \ldots$$

says we take one 1 and one 4, and no other integer, giving the partition $1 + 4$. Thus, the correspondence between exponents and partitions in the $n = 5$ case is concretely:

$$\begin{aligned}
1, 1, 1, 1, x^5, 1, \ldots &\rightsquigarrow 5 \\
x, 1, 1, 1, x^4, 1, \ldots &\rightsquigarrow 4 + 1 \\
1, x^2, x^3, 1, 1, 1, \ldots &\rightsquigarrow 3 + 2 \\
x^2, 1, x^3, 1, 1, 1, \ldots &\rightsquigarrow 3 + 1 + 1 \\
x, x^4, 1, 1, 1, 1, \ldots &\rightsquigarrow 2 + 2 + 1 \\
x^3, x^2, 1, 1, 1, 1, \ldots &\rightsquigarrow 2 + 1 + 1 + 1 \\
x^5, 1, 1, 1, 1, 1, \ldots &\rightsquigarrow 1 + 1 + 1 + 1 + 1
\end{aligned}$$

In general, we interpret each factor in the product

$$(1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots)(1 + x^4 + x^8 + \cdots)\cdots$$

as keeping track of how many times we should each integer when forming partitions: the first factor keeps track of how many 1's to use, the second keeps track of how many 2's to use, the third is the number of 3's to use, and so on. This gives a bijection between the different terms we get in this product and integer partitions, where specifically each way of obtaining x^n corresponds in a one-to-one manner to a certain partition of n.

So, to reiterate, we get

$$\sum_{n=0}^{\infty} p(n)x^n = (1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots)(1 + x^4 + x^8 + \cdots)\cdots.$$

To introduce a concept we will see in a few weeks, the expression on the right is called the *ordinary generating function* of the sequence $p(n)$ of integer partitions, which means that it is a function whose expansion as a *power series* encodes the terms in the sequence $p(n)$. We will review basic facts about power series later on when needed, but they will be important in understanding generating functions in general. The key idea is that this single function encodes the infinite amount of data given by the sequence in question, and we can thus hope to understand properties of this sequence via properties of this function instead. We will elaborate on this soon enough, but will as a consequence give us a way to show that the number of partitions of n into distinct parts is the same as the number of partitions into odd parts, by interpreting this problem in terms of generating functions instead, where it will be easier to make senes of.

One final thing. The product on the right of

$$\sum_{n=0}^{\infty} p(n)x^n = (1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots)(1 + x^4 + x^8 + \cdots)\cdots$$

gets cumbersome to write every single time, but luckily we have a much simpler way in which to express this. First, if we do think back to material on power series, specifically the notion of a *geometric series*, each factor here can be written as a single fraction! In general, we have

$$1 + y + y^2 + y^3 + \cdots = \frac{1}{1 - y}$$

so by substituting $y = x^k$ we get

$$1 + x^k + x^{2k} + x^{3k} + \cdots = \frac{1}{1 - x^k}$$

Thus, our product expression simplifies to

$$\sum_{n=0}^{\infty} p(n)x^n = \frac{1}{1-x} \cdot \frac{1}{1-x^2} \cdot \frac{1}{1-x^3} \cdot \frac{1}{1-x^4} \cdots$$

More compactly we can write this as

$$\sum_{n=0}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \frac{1}{1-x^k}$$

where the product notation $\prod_{k=1}^{\infty}$ is meant to be analogous to summation notation Σ, only we multiply instead of add: for $k = 1$ we get $\frac{1}{1-x}$, for $k = 2$ we get $\frac{1}{1-x^2}$, for $k = 2$ we get $\frac{1}{1-x^2}$, and so on, and we multiply all terms we get. Thus

$$\prod_{k=1}^{\infty} \frac{1}{1-x^k}$$

is the compact form of the ordinary generating function of the sequence $p(n)$ of integer partitions.

Lecture 13: Cycles in Permutations

Warm-Up. If we have an integer partition of n, we can interpret the parts involved as describing the sizes of the subsets used in a *set* partition of $[n]$. For instance, $6 = 3 + 2 + 1$ leads to set partitions of $[6]$ into three subsets, one of size 3, one of size 2, and one of size 1. We determine the number of set partitions of $[13]$ which correspond to the integer partition

$$13 = 3 + 3 + 2 + 2 + 2 + 1$$

in this way; such partitions hence consist of two subsets of size 3, three of size 2, and one of size 1.

First, we have $\binom{13}{3}$ ways of choosing the first subset of size 3, and then $\binom{10}{3}$ ways of choosing the second subset of size 3. But, multiplying these together over-counts the possibilities by a factor of 2, since we could have chosen these two subsets in the other order instead. Thus we get

$$\frac{1}{2}\binom{13}{3}\binom{10}{3}$$

ways of choosing the two subsets of size 3. Continuing on, we next have $\binom{7}{2}\binom{5}{2}\binom{3}{2}$ ways of specifying the first, second, and third subset of size 2, but again this over-counts, now by a factor of $3! = 6$ coming from the possible orders in which these three subsets could have been chosen. So,

$$\frac{1}{6}\binom{7}{2}\binom{5}{2}\binom{3}{2}$$

is the number of ways of forming the 2-elements subsets once we've chosen the 3-element subsets. At the end of this we have one element remaining, which must go into its own subset of size 1 in one way, but let us think of this one way as $\binom{1}{1}$. In total we thus get

$$\frac{1}{2}\binom{13}{3}\binom{10}{3}\frac{1}{6}\binom{7}{2}\binom{5}{2}\binom{3}{2}\binom{1}{1}$$

set partitions of [13] which correspond to the integer partition $13 = 3+3+2+2+2+1$.

To see what to do in general, let us express this same answer in an alternate way. If we write out all the factorials involved in these $\binom{n}{k}$ values and so some fun cacellation, we get:

$$\frac{1}{2\cdot 6}\cdot\frac{13!}{3!10!}\cdot\frac{10!}{3!7!}\frac{7!}{2!5!}\cdot\frac{5!}{2!3!}\cdot\frac{3!}{2!1!}\frac{1!}{1!0!} = \frac{13!}{(3!3!2!2!2!1!)2!3!1!}.$$

All other terms in the numerator apart from 13! cancel out with some term in the denominator, and we have written the initial $2\cdot 6$ in the denominator in terms factorials as well and grouped all factorials in the denominator in a special way (writing 0! as 1!), which will become clear in a second. The question is: why does this final expression make sense as the correct answer, apart from saying simply that we derived it via cancellation? Think of it in the following way. We want to form subsets which look like:

$$\{\underbrace{\bullet\bullet\bullet}_{\text{size 3}}\}\{\underbrace{\bullet\bullet\bullet}_{\text{size 3}}\}\{\underbrace{\bullet\bullet}_{\text{size 2}}\}\{\underbrace{\bullet\bullet}_{\text{size 2}}\}\{\underbrace{\bullet\bullet}_{\text{size 2}}\}\{\underbrace{\bullet}_{\text{size 1}}\}$$

Imagine we list our 13 elements in some order in the 13 possible locations above:

$$\{\underbrace{a_1a_2a_3}_{\text{size 3}}\}\{\underbrace{a_4a_5a_6}_{\text{size 3}}\}\{\underbrace{a_7a_8}_{\text{size 2}}\}\{\underbrace{a_9a_{10}}_{\text{size 2}}\}\{\underbrace{a_{11}a_{12}}_{\text{size 2}}\}\{\underbrace{a_{13}}_{\text{size 1}}\}$$

There are 13! such listings, but many of these corresponds to the same set partition, so we must deal with over-counting. First, in each of the subsets of size 3, the elements within it could have been listed in 3! ways in total, so this is one source of over-counting; we thus divide by 3!3! to deal with this, one for each subset of size 3. Similarly, the elements within each subset of size 2 can be listed in 2! ways within the same subset, so we have over-counted by a factor of 2!2!2!. These factors give the

$$3!3!2!2!2!$$

in the parentheses in $\frac{13!}{(3!3!2!2!2!1!)2!3!1!}$. But now, we also have over-counting due to the fact that the two subsets of size 3 can themselves be arranged in 2! ways, and that the three subsets of size 2 can be arranged in 3! ways, which gives the 2!3! part outside the parentheses in the fraction above. The 1! terms come from the ways of rearranging the elements in the subset of size 1, and from the number of ways of rearranging the subsets of size 1 among themselves. Thus the number of set partitions corresponding to $13 = 3+3+2+2+2+1$ is indeed

$$\frac{13!}{\underbrace{3!3!2!2!2!1!}_{\text{rearrange elements within subsets}}\quad\underbrace{2!3!1!}_{\text{rearrange subsets of same size}}}$$

In general, say we take an integer partition of n into positive integers $k_1, k_2, \ldots, k_t$ with k_i appearing m_i times overall:

$$n = \underbrace{k_1+\cdots+k_1}_{m_1\text{ times}} + \underbrace{k_2+\cdots+k_2}_{m_2\text{ times}} + \cdots + \underbrace{k_t+\cdots+k_t}_{m_t\text{ times}}$$

Then the number of set partitions of $[n]$ whose subsets are of sizes given by these integers is

$$\frac{n!}{(k_1!)^{m_1}\cdots(k_t)^{m_t}\, m_1!\cdots m_t!}$$

We get this first arranging our subsets in a list—with those of size k_1 first, then of size k_2, and so on—with the elements of $[n]$ to be put into, next listing the elements of $[n]$ in $n!$ possible orders among these subsets, and finally we deal with over-counting by dividing by $k_i!$ for each of the m_i subsets of size k_i, which comes from the number of ways of arranging the elements within that subset, and dividing by $m_i!$ for the number of ways of rearranging the subsets of size k_i among each themselves. We will see a similar type of expression in a bit with our new material.

Permutations revisited. Up until now, when we've used permutations we've only thought about them in terms of rearrangements, so that a permutation of 123456 is just something like 345261. But to get a better sense of the role which permutations play in combinatorics and elsewhere more broadly, we should instead interpret them as *functions*. Namely, thinking of the procedure which turns 123456 into 345621 as a function which sends each term in the first list into the term in the second list which is in the same location:

$$\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 3 & 4 & 5 & 6 & 2 & 1 \end{array}$$

So, the permutation 345621 is the function which sends 1 to 3, 2 to 4, 3 to 5, 4 to 6, 5 to 2, and 6 to 1. Such a function bijective (one-to-one and onto), so a permutation of $[n]$ is thus a bijection function $[n] \to [n]$. This perspective is the way in which permutations are thought of elsewhere in mathematics, not simply as the result of some rearrangement, but as the actual process of performing that rearrangement.

The main benefit of this perspective is that it then makes sense to *compose* permutations, by composing them as functions. For instance, take the permutation $\sigma = 345621$ above and the permutation $\tau = 243165$, which in function form is

$$\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 4 & 3 & 1 & 6 & 5 \end{array}$$

The product $\tau\sigma$ is the function defined by first applying σ and then applying τ. So, in this case this product sends 1 to 3 since σ first sends 1 to 3 and then τ sends 3 to 3, $\tau\sigma$ sends 2 to 1 since σ first sends 2 to 4 and τ then sends 4 to 1, and so on we see first what σ does to an element and then what τ does to the result:

$$\tau\sigma = \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 3 & 1 & 6 & 5 & 4 & 2 \end{array}$$

So, $\tau\sigma$ is the permutation 316542 when written in single-line notation.

Cycles. But as we can already see, using this arrow notation to compute products can get kind of cumbersome and tedious to write out. Luckily we have a better notation available, which results from expressing permutations in terms of *cycles*. Consider again $\sigma = 345621$. In cycle notation we start with a first cycle beginning with 1:

$$\sigma = (1$$

The permutation σ sends 1 to 3, which we denote by writing 3 next to 1 in cycle form:

$$\sigma = (13$$

Next we determine where 3 goes: σ sends 3 to 5, so 5 comes next in cycle form:

$$\sigma = (135$$

In general, in cycle notation each element is sent to one which comes immediately after. Now, σ sends 5 to 2 so we get

$$\sigma = (1352$$

and 2 gets sent to 4, so

$$\sigma = (13524$$

Finally, 4 gets sent to 6, so

$$\sigma = (135246$$

and 6 gets sent back to 1, which indicate by closing this cycle:

$$\sigma = (135246).$$

We always interpret the final term in a cycle as being sent to the first term in that cycle. Hence the permutation $\sigma = 345621$ in cycle notation is $\sigma = (135246)$. We say that σ consists of a single cycle of length 6, or of a single 6-cycle. The point is that the numbers within this one cycle are all being "cycled" among themselves under this permutation.

Now, for $\tau = 243165$, we have 1 sent to 2, 2 sent to 4, and 4 sent back to 1, so this gives the 3-cycle (124) as part of τ. But this is not the end, since we still have to determine what happens to $3, 5, 6$. So, we start a new cycle with 3:

$$\tau = (124)(3$$

We have that τ sends 3 to itself, which thus makes up the 1-cycle (3):

$$\tau = (124)(3)($$

A third cycle begins with 5, which gets sent to 6, which gets sent to 5, so we get one final 2-cycle:

$$\tau = (124)(3)(56).$$

The point is that even though τ as a permutation of the six numbers $1, 2, 3, 4, 5, 6$, actually $1, 2, 4$ are being permuted (i.e. "cycled") among each other, 3 is only permuted among itself (so not being permuted at all), and $5, 6$ are permuted among each other. This is easier to see via cycle notation than any other notation. We say that τ then consists of one 3-cycle, one 1-cycle, and one 2-cycle.

The composition $\tau\sigma$ then looks like:

$$\tau\sigma = (124)(3)(56)(135246),$$

which we can re-express in terms of *disjoint* cycles as follows. We read from right to left, since we are fisrt applying σ and then τ. Starting with 1, the first cycle on the right (135246) sends 1 to 3. The next cycle (56) when reading right to left does nothing to 3, so we still have 3 after this step. (In general, we interpret a number which is missing from a cycle as one which is being sent to itself.) The next cycle 3 also does nothing to 3, so we still have 3 after this step, and the final cycle (124) again does nothing to 3, so we still get 3 as a result. Thus, overall under this entire composition 1 was sent to 3:

$$(124)(3)(56)(135246) = (13$$

Now we determine where 3 goes: (135246) sends 3 to 5, (56) next sends 5 to 6, and none of the remaining cycles (3) and (124) do anything further to 6, so overall 3 was sent to 6 under this product:

$$(124)(3)(56)(135246) = (136$$

Next, 6 is first sent to 1 under (135246), then (56) and (3) do nothing to 1, but the final cycle (124) sends 1 to 2, so overall 6 is sent to 2:

$$(124)(3)(56)(135246) = (1362$$

The number 2 is sent to 4 under (135246), which is eventually sent to 1 under the final (124), so this closes off this cycle:

$$(124)(3)(56)(135246) = (1362)($$

We start the next cycle by seeing what happens to a number we haven't dealt with yet, like 4, and keep going until we've exhausted all numbers:

$$(124)(3)(56)(135246) = (1362)(45).$$

Thus, the permutation $\tau\sigma = (1362)(45)$ consists of two cycles in disjoint cycle notation, one of length 4 and one of length 2. Note that even the original expression $\tau\sigma$ we had as

$$\tau\sigma = (124)(3)(56)(135246)$$

simply by sticking τ to the let of σ was *not* in disjoint cycle form since numbers appear more than once, but that we were able to write it in disjoint cycle form $\tau\sigma = (1362)(45)$ where each number only appears once via the procedure described above. Disjoint cycle form is what we'll mainly use when talking about properties of cycles in permutations.

Cycles of a given type. We first determine the number of permutations of [7] which consist of a single 7-cycle in disjoint cycle notation, which hence looks like

$$(a_1a_2a_3a_4a_5a_6a_7).$$

We can imagine listing the elements of [7] in 7! ways in this 7-cycle, but the point is that many of these listings describe the same permutation. For instance, (1234567) is the same permutation as (2345671), which is the same as (3456712), and so on, since each of these sends 1 to 2, 2 to 3, 3 to 4, 4 to 5, 5 to 6, 6 to 7, and 7 to 1. What matters is not how the elements are listed but rather where they are in relation to each other since this is what determines the permutation as a function. Thus, given any possible listing of elements in a 7-cycle, there are 6 *other* listings which describe the same permutation, so the 7! total listings we have over-counts by a factor of 7. Thus there are

$$\frac{7!}{7} = 6!$$

permutations which consist of a single 7-cycle. In general, for any n there are

$$\frac{n!}{n} = (n-1)!$$

permutations of $[n]$ which consist of a single n-cycle in disjoint cycle notation. (This number answers the problem of counting the number of ways of arranging n people around a round table where it matters not which specific seats they are in, but to whom they are adjacent.)

Next we determine the number of permutations of [13] which consist of three 3-cycles and two 2-cycles in disjoint cycle notation:

$$(\bullet\bullet\bullet)(\bullet\bullet\bullet)(\bullet\bullet\bullet)(\bullet\bullet)(\bullet\bullet)$$

List out all 13 elements in 13! ways. The numbers within the first 3-cycle can be arranged in 3 ways in total without changing the permutation, and similarly for the numbers within the second and third 3-cycle. Thus we have over-counted by a factor of $3 \cdot 3 \cdot 3 = 3^3$ in these. For the same reason, the elements within each 2-cycle can be listed in 2 ways without changing the permutation, so we over-count by a factor of 2^2. Finally, the three 3-cycles can be arranged in 3! ways among themselves without changing the permutation, and the two 2-cycles can be arranged in 2! ways, so we over-count by these factors also. (Notice the similarity between this and our Warm-Up problem.) Thus, the number of permutations of [13] which consist of three 3-cycles and two 2-cycles in disjoint cycle notation is

$$\frac{13!}{3^3 2^2 3! 2!}$$

In general, the number of permutations of $[n]$ which consist of a_1 1-cycles, a_2 2-cycles, a_3 3-cycles, and so on up until a_k k-cycles is:

$$\frac{n!}{1^{a_1} 2^{a_2} \cdots k^{a_k} a_1! a_2! \cdots a_k!}.$$

The i^{a_i} terms in the denominator come from the number of ways the terms within each i-cycle can be arranged without changing the permutation, and the $a_i!$ terms come from the number of ways of rearranging the i-cycles among themselves, which also does not change the permutation. Again, this is a very similar to the idea in the Warm-Up problem where we counted set partitions of a given type determined by an integer partition.

Unsigned Stirling numbers. We denote by $c(n, k)$ the number of permutations of $[n]$ which consist of k cycles overall in disjoint cycle notation. To be clear, k is the number of cycles used, and does not refer to the length of any one cycle. These are called the *unsigned Stirling numbers*, and will soon give lead us to the *Stirling numbers of the first kind*, which we will then relate to the previous type of Stirling number we had before.

For instance, let us compute $c(4, 2)$. In order for a permutation of [4] to be made up of 2 disjoint cycles overall, the only possibilities are to have a 1-cycle and a 3 cycle, or two 2-cycles:

$$(\bullet)(\bullet\bullet\bullet) \text{ or } (\bullet\bullet)(\bullet\bullet)$$

There are

$$\frac{4!}{1 \cdot 3} = 8$$

of the first type using the general formula we derived above for permutation with a_i i-cycles, and there are

$$\frac{4!}{2^2 2!} = 3$$

of the second type using the same formula. Hence we get $c(4, 2) = 8 + 3 = 11$. We can actually list them all out by brute-force:

$$(1)(234), (1)(243), (2)(134), (2)(143), (3)(124), (3)(142), (4)(123), (4)(132) \text{ of the first type, and}$$

$$(12)(34), (13)(24), (14)(23) \text{ of the second type.}$$

In general, however, it will be nice that an explicit formula is available.

Lecture 14: More on Stirling Numbers

Warm-Up 1. We compute the unsigned Stirling number $c(6,3)$. In order for a permutation of $[6]$ to decompose into three disjoint cycles, the possible lengths are:

4-cycle,1-cycle,1-cycle   3-cycle,2-cycle,1-cycle   2-cycle,2-cycle,2-cycle.

Based on the general formula we derived last time for the number of permutation with a_i cycles of length i, there are

$$\frac{6!}{4 \cdot 1^2 \cdot 1! \cdot 2!} = \frac{6!}{8} = 90$$

of the first type,

$$\frac{6!}{3 \cdot 2 \cdot 1 \cdot 1!1!1!} = \frac{6!}{6} = 5! = 120$$

of the second type, and

$$\frac{6!}{2^3 \cdot 3!} = \frac{6!}{48} = 15$$

of the third type. Thus $c(6,3) = 90 + 120 + 15 = 225$.

Warm-Up 2. We determine the number of permutations of $[n]$ (for $n \geq 3$) which have 1 and 2 as part of the same 3-cycle when written in disjoint cycle notation. First, there are $n-2$ ways of choosing the remaining element in the 3-cycle to which $1,2$ belonging, and once this is chosen there are 2 possible 3-cycles we can form with it and 1 and 2 as entries: $(12a)$ or $(1a2)$. So there are $2(n-2)$ possibilities for the 3-cycle. The remaining $n-3$ elements can then be permuted in anyway whatsoever, so there are $(n-3)!$ possibilities for these remaining elements. Hence we get

$$2(n-2)(n-3)! = 2(n-2)!$$

possible permutations with 1 and 2 as part of the same 3-cycle.

Stirling numbers via recursion. As with the Stirling number $S(n,k)$, we will not be able to find an expilict formula for $c(n,k)$ in general. We have such a formula in a few special cases: $c(n,1) = (n-1)!$ since here we need a single cycle of length n; $c(n,n) = 1$ since here we need n cycles of length 1; and $c(n,n-1) = S(n,n-1) = \binom{n}{2}$ since here we need a 2-cycle and $n-2$ 1-cycles, so it comes down to choosing the 2 elements which go into the 2-cycle. Of course, $c(n,k) = 0$ for $k > n$ since it is not possible to have more cycles than elements in $[n]$. In general, we have $S(n,k) \leq c(n,k)$ since both numbers come from partitioning $[n]$, only that in $c(n,k)$ we also take into account the order of elements within a subset, so there are more possibilities.

But, as with the case of $S(n,k)$, we can find a nice recursion which does work. We claim that

$$c(n,k) = c(n-1,k-1) + (n-1)c(n-1,k).$$

To see this we keep track of the cycle to which n belongs. If n makes up its own 1-cycle, then we need to break up the remaining $n-1$ elements into $k-1$ cycles since we need k cycles overall, and this can be done in $c(n-1,k-1)$ many ways. If n does not make up its own 1-cycle, then we can distribute $1,2,\ldots,n-1$ into k cycles in $c(n-1,k)$ ways, and put n into one of these existing cycles. The key observation is that there are $n-1$ possible locations for where n can go, one *after* each existing element in a decomposition of $[n-1]$ into k-cycles:

$$(a_1 \bullet a_2 \bullet)(a_3 \bullet a_4 \bullet a_5 \bullet) \cdots (a_{n-1} \bullet)$$

The point is that we don't need to worry about inserting n at the *start* of a cycle, since these are already covered by inserting n after the last term in that cycle:

$$(na_3a_4a_5) = (a_3a_4a_5n).$$

Thus, there are $(n-1)c(n-1,k)$ permutations where n is part of a cycle of length longer than 1, giving the required identity:

$$c(n,k) = \underbrace{c(n-1,k-1)}_{\text{contain }(n)} + \underbrace{(n-1)}_{\substack{\text{insert } n \text{ into} \\ \text{existing cycle}}} \underbrace{c(n-1,k)}_{\substack{\text{break up } [n-1] \\ \text{into } k \text{ cycles}}}$$

So, we could compute something like $c(6,3)$ using this as follows:

$$\begin{aligned} c(6,3) &= c(5,2) + 5c(5,3) \\ &= [c(4,1) + 4c(4,2)] + 5[c(4,2) + 4c(4,3)] \\ &= c(4,1) + 4[c(3,1) + 3c(3,2)] + \text{and so on,} \end{aligned}$$

but of course in this case there is the simpler way we used in the Warm-Up. For computations by a computer, however, the recursion is simpler to program.

Towards generating functions. But, the real reason why we care about the recursive formula for $c(n,k)$ is not so much to help us compute these numbers, but rather to help us find the *ordinary generating function* of the sequence $c(n,k)$, where k varies. The notion of a "generating function" is one we briefly mentioned when discussing integer partitions, and is one we'll come back to much more heavily soon enough, but here what we mean is the function

$$\sum_{k=1}^{n} c(n,k)x^k$$

Our goal is to determine more explicitly, without using a summation, what this function actually is. We claim that in fact

$$\sum_{k=1}^{n} c(n,k)x^k = x(x+1)(x+2)(x+3)\cdots(x+n-1),$$

so that $x(x+1)(x+2)\cdots(x+n-1)$ is the function (a polynomial in this case) which encodes $c(n,k)$ as the coefficient of x^k.

Indeed, we show that if we write the right side as a sum of powers of k:

$$x(x+1)(x+2)(x+3)\cdots(x+n-1) = \sum_{k=0}^{n} a_{n,k}x^k$$

then the coefficients $a_{n,k}$ which arise satisfy the same recursive formula as do the $c(n,k)$; since $c(n,0) = 0 = a_{n,0}$ ($a_{n,0}$ is the coefficient of $x^0 = 1$ in $x(x+1)\cdots(x+n-1)$, which is 0 since there is no constant term when multiplying out) and $c(1,1) = 1 = a_{1,1}$ ($a_{1,1}$ is the coefficient of x^1 in x) for all $n \geq 1$, satisfying the same recursion then implies that $c(n,k) = a_{n,k}$ in general. For the sake of notation set

$$P_n(x) = x(x+1)\cdots(x+n-1) = \sum_{k=0}^{n} a_{n,k}x^k.$$

Note that then

$$P_{n-1}(x) = x(x+1)\cdots(x+n-2) = \sum_{k=0}^{n-1} a_{n-1,k}x^k.$$

Hence

$$\begin{aligned}
P_n(x) &= (x+n-1)(x+n-2)\cdots(x+1)x \\
&= (x+n-1)P_{n-1}(x) \\
&= (x+n-1)\sum_{k=1}^{n} a_{n-1,k}x^k \\
&= x\sum_{k=1}^{n} a_{n-1,k}x^k + (n-1)\sum_{k=1}^{n} a_{n-1,k}x^k \\
&= \sum_{k=1}^{n} a_{n-1,k}x^{k+1} + (n-1)\sum_{k=1}^{n} a_{n-1,k}x^k \\
\sum_{k=1}^{n} a_{n,k}x^k &= \sum_{k=0}^{n-1} a_{n-1,k-1}x^k + (n-1)\sum_{k=1}^{n} a_{n-1,k}x^k.
\end{aligned}$$

We rewrote the first term on the right to start at $k = 0$ so that the exponent of x would be the same throughout. Comparing coefficients of x^k on both sides we get:

$$a_{n,k} = a_{n-1,k-1} + (n-1)a_{n-1,k}$$

which is the same recursion satisfied by $c(n,k)$. Thus we conclude that $a_{n,k} = c(n,k)$, so that

$$\sum_{k=1}^{n} c(n,k)x^k = x(x+1)(x+2)\cdots(x+n-1)$$

as claimed. Of course, this makes sense even if we start the sum on the left zero $k = 0$, since $c(n,0) = 0$ anyway:

$$\sum_{k=0}^{n} c(n,k)x^k = x(x+1)(x+2)\cdots(x+n-1)$$

Again, the point is that the right side is a single function which encodes $c(n,k)$ (for fixed n) as the coefficients in its power series expansion, which in this case is a finite series.

Stirling numbers of various kinds. With the equality derived above, we can finally see how these types of Stirling numbers relate to the ones we considered previously when discussing set partitions. First, make the substitution $x \mapsto -x$ in

$$\sum_{k=0}^{n} c(n,k)x^k = x(x+1)(x+2)\cdots(x+n-1)$$

to get:

$$\sum_{k=0}^{n} c(n,k)(-x)^k = (-x)(-x+1)(-x+2)\cdots(-x+n-1).$$

We can factor -1 out of each factor on the right side, so with n factors in total we get

$$\sum_{k=0}^{n} c(n,k)(-1)^k x^k = (-1)^n x(x-1)(x-2)\cdots(x-n+1).$$

Finally multiply through by $(-1)^n$ to get

$$\sum_{k=0}^{n} (-1)^{n-k} c(n,k) x^k = x(x-1)(x-2)\cdots(x+n-1) = (x)_n.$$

We define the *Stirling numbers of the first kind* $s(n,k)$ as $s(n,k) = (-1)^{n-k}c(n,k)$, so that the equality above becomes

$$\sum_{k=0}^{n} s(n,k)x^k = (x)_n.$$

The point is that this looks suspiciously similar to the equality

$$x^n = \sum_{k=0}^{n} S(n,k)(x)_k$$

we had for the Stirling numbers $S(n,k)$ of the second kind, so what exactly is going on here? The answer is that the two types of Stirling numbers $S(n,k)$ and $s(n,k)$ are in a sense *inverse* to one another, in that the $S(n,k)$ translate from expressions $(x)_k$ to expressions x^n and the $s(n,k)$ translate the other way around from x^k to $(x)_n$. To be precise, form the infinitely-sized matrix S with entries $S(n,k)$, where n is the row, and k is the column, both starting at 0):

$$S = \begin{bmatrix} S(0,0) & S(0,1) & S(0,2) & S(0,3) & \cdots \\ S(1,0) & S(1,1) & S(1,2) & S(1,3) & \cdots \\ S(2,0) & S(2,1) & S(2,2) & S(2,3) & \cdots \\ S(3,0) & S(3,1) & S(3,2) & S(3,3) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 0 & \cdots \\ 0 & 1 & 1 & 0 & \cdots \\ 0 & 1 & 3 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

and the infinitely-sized matrix s with entries $s(n,k)$:

$$s = \begin{bmatrix} s(0,0) & s(0,1) & s(0,2) & s(0,3) & \cdots \\ s(1,0) & s(1,1) & s(1,2) & s(1,3) & \cdots \\ s(2,0) & s(2,1) & s(2,2) & s(2,3) & \cdots \\ s(3,0) & s(3,1) & s(3,2) & s(3,3) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 0 & \cdots \\ 0 & -1 & 1 & 0 & \cdots \\ 0 & 2 & -3 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

The fact is that these two matrices are inverses of one another, meaning $sS = I$ and $Ss = I$ where I is an infinitely-sized identity matrix. The reason why this follows from the identities given above uses some concepts from abstract linear algebra, namely the notion of a "change of basis" matrix, and the fact that both $\{1, x, x^2, x^3 \ldots\}$ and $\{1, x, x(x-1), x(x-1)(x-2), \ldots\}$ are bases for the "vector space of all polynomials", but spelling all of this out is best left to a different course.

Lecture 15: More on Permutations

Warm-Up. Let $f(n)$ denote the number of permutations of n whose cube is the identity permutation, meaning permutations σ satisfying $\sigma^3 = (1)(2)(3)\cdots(n)$. We find a recursive formula for $f(n)$. The key point is that in order to have third power be the identity, the only possible cycles we can have are those of length 1 and those of length 3; no other length will give a cycle whose three-fold composition with itself is the identity. For instance, for a 2-cycle we get $(ab)^3 = (ab)$, and for something like a 4-cycle we get

$$(abcd)^3 = (adcb).$$

(In general, for a k-cycle, the powers which give the identity are those which are multiples of k.)

So, we need to determine recursively the number of permutations of n which consist only of 1-cyles and 3-cycles. Focus on where n goes: either into it's own 1-cycle or into a 3-cycle. In the first case, we are left with permutations of $[n-1]$ consisting still only of 1- and 3-cycles, and there are $f(n-1)$ many of these. If n goes into a 3-cycle, there are $\binom{n-1}{2} = \frac{(n-1)(n-2)}{2}$ ways of picking the other elements of this 3-cycle and 2 such 3-cycles which can be formed with these three elements, so there are

$$2\frac{(n-1)(n-2)}{2} = (n-1)(n-2)$$

possibilities for the 3-cycle containing n. The remaining $n-3$ elements have to be broken down into only 1- and 3-cycles, so there are $f(n-3)$ possibilities for these, giving $(n-1)(n-2)f(n-2)$ possibilities in total where n is part of a 3-cycle. Thus we get:

$$f(n) = f(n-1) + (n-1)(n-2)f(n-3)$$

as the required recursion. (We should probably assume $n \geq 4$ here.)

Even and odd permutations.

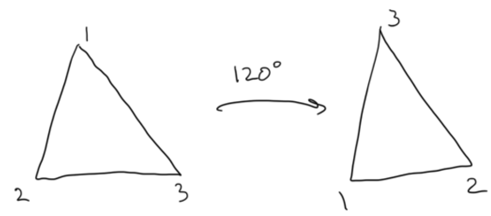
Lecture 16: Inclusion/Exlcusion

To come!

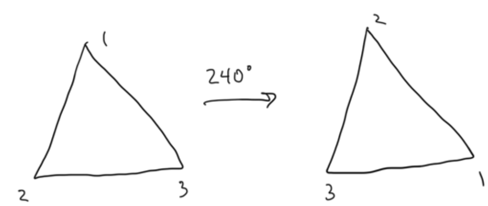
Lecture 17: Burnside's Lemma

Permutations as symmetries. Before we move on from permutations, we consider one final perspective as to what they allow us to do. None of this material is in the book, and indeed it falls within the real of things you study in more detail in a course in Abstract Algebra. But, I think this material is crucial to understanding the idea that permutations can manifest themselves more broadly than simply in terms of rearranging objects, which is quite important elsewhere. In the end, the specific result we look at, Burnside's Lemma, is a result about counting so that it does fall within combinatorics, but the point of covering this is really to give a sense of the larger role permutations play in mathematics.

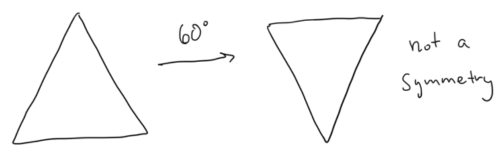
Take a triangle, say, and label the vertices $1, 2, 3, 4$. We consider the *rotational symmetries* of this triangle, which are transformations which when applied to this triangle result in the same triangle. For instance, we can rotate the triangle by $120°$:



Or, we can rotate by 240°:



Rotating by something like 60° is not a symmetry however since, although it results in a triangle, it does not result in the *same* triangle:



A triangle thus has three rotational symmetries: rotation by $120°, 240°$, and by $0°$, the last of which does nothing at all to the given triangle.

The key observation is that each of these rotations has the effect of permuting the labeled vertices, and we can characterize each rotation via this associated permutation. For instance, when rotating by 120° above, vertex 1 got sent to vertex 2, vertex 2 got sent to 3, 3 got sent to 2, and 3 got sent to 1, so the effect of this rotation on the vertices is given by the 3-cycle (123):

$$\text{rotation by } 120° = (123).$$

Similarly, rotation by 240° sends 1 to 3, 3 to 2, and 2 to 1, so this is given by the 3 cycle (132):

$$\text{rotation by } 240° = (132).$$

Rotation by 0° leaves every vertex as is, so this is the identity permutation:

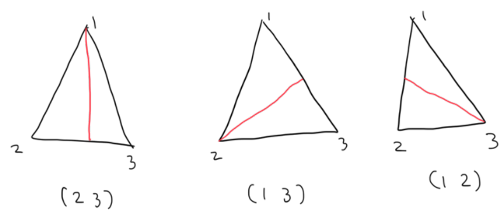
$$\text{rotation by } 0° = (1)(2)(3).$$

We say that these three rotations, or rather these three permutations, form the *rotational symmetry group* of the triangle, or the triangle's *group of rotations*.

Groups. And now a slight digression: we have used the term "group" here, without explaining what this term really means. The concept of a *group* is a crucial one in Abstract Algebra, and is essentially what you spend the entire first quarter of such a course studying. We will not give the most general definition here since we will only look at special examples, but it comes down to the following for our purposes: a *group* is a set of permutations which contains the identity permutation, has the property that the product of any two permutations in that set remains within that set, and has the property that the *inverse* of any permutation in that set is also in that set. (The inverse of a permutation is the permutation you compose it with in order to get the identity; for instance, (123) and (132) are inverses of one another since (123)(132) = (1)(2)(3).)

Again, groups in general can be defined in a more abstract way than this, but this is good enough for our purposes. Even then, you will never be asked to verify that some given candidate group is actually a group, since this will always be the case for the examples we will look at. So, you really don't have to know this definition at all in this course, but since we will use the term "group" a few times it makes sense to at least have an idea for what it means.

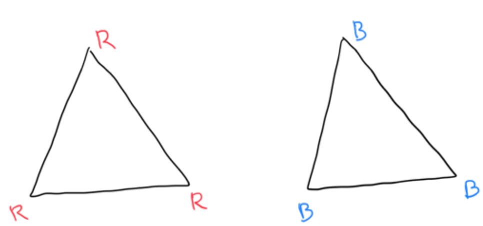
Back to permutations as symmetries. Let's come back to the triangle. In addition to rotations, we can also consider *reflections* as symmetries. For instance, we can reflect the triangle across a line which connects a vertex to the midpoint of the opposite side:



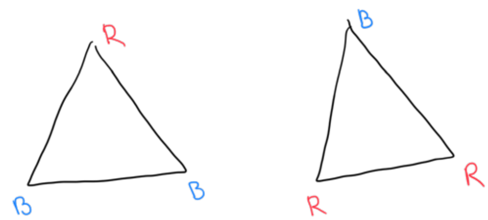
As in the case of rotations, we can also describe these reflections in terms of cycles, as we did above. Note that a reflection is its own inverse, which makes sense visually: performing the same reflection twice in a row is the same as doing nothing at all. The set of all three rotations and three reflections is then the full *symmetry group* of the triangle. In a course in Abstract Algebra, this set would be denoted by D_3 (D_n in general denotes the symmetry group of a regular n-gon) and is known as a *dihedral group*, but we will not use this notation nor terminology here.

Colorings up to symmetry. The point of introducing symmetry groups for us is that they give a way to group together things we want to consider as being the "same" for the purposes of counting. Imagine that we now color the vertices of our triangle with one of n colors. We declare two such colorings to be the same if one can be obtained from the other via some rotation, i.e. via an element of the rotational symmetry group of the triangle. We are thus interested in determining the number of colorings distinct *up to the action of this group*. ("Up to the action of this group" is just the shorthand way to say that two colorings are treated as being the same if you can get one from the other via an element of the group in question.)

For instance, consider the case of $n = 2$ colors, say red and blue. First, we can color every vertex red or every vertex blue:

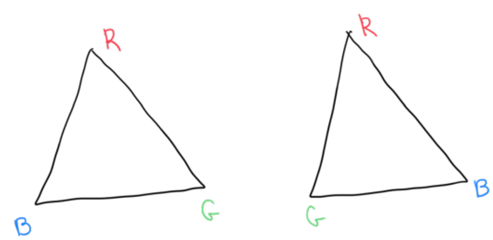


These are distinct since neither can be obtained from the other via rotations. Second, we can make two vertices one color and the third vertex the other color:



Now, any other coloring of vertices can be obtained from one of these via a rotation; for instance, the coloring where the top vertex is blue, the bottom left is red, and the bottom right is blue can be obtained from the triangle on the left in the picture directly above by rotating by 120°. So, up to rotational symmetry, there are only 4 colorings of the vertices of a triangle with 2 colors.

Say we use $n = 3$ colors: red, blue, and green. Again, we have three possibilities where all vertices have the same color. Now, for possibilities where only two colors are used, it comes down to picking one color for one vertex and one for the other two, but the specific vertex we pick to be of the first color does not matter since any other configuration can obtained from any other via a rotation. So, we have two possibilities when red is chosen as the color for the vertex: either the other two are blue or the other two are green. Similarly, we have two possibilities for having one blue vertex, and two for having one green vertex, so we get 6 total colorings which use only two colors. Finally, have two possibilities for using all three colors:



These are different from each other since neither can be obtain from the other via a rotation alone. Hence, we get

$$3 + 6 + 2 = 11$$

distinct colorings of the vertices of a triangle up to rotational symmetries when using 3 colors.

Why colorings? One issue to address is why we should care about colorings at all. After all, who really cares about coloring the vertices of a triangle, or of some other shape? The point is that, although we use the term "coloring" and visualize this literally in terms of coloring, the concept of a "color" in general can be thought of as characterizing any other shared property we might care about. For instance, maybe we have a collection of people and think of "color" as really meaning *age*; then we can ask for ways of permuting these people "up to age", where we treat two permutations as being the same if they result in people having the same age as at the start. The point is that the idea of coloring vertices of some "graph" in general is one which as broad interpretations, based on whatever characteristic we choose to treat as the one we mean by the word "color". So, these types of counting problems really are ubiquitous.

Burnside's Lemma. Counting distinct colorings by hand can get tedious fairly quickly, so we need a better way to do this. *Burnside's Lemma* is the result which gives us the answer in general. Suppose we color some number of objects using some number of colors, where we treat two colorings as being the same if one can be obtained from the other via an element of a permutation group G. The Burnside's Lemma says that:

$$\text{the number of distinct colorings up to elements of } G = \frac{1}{|G|} \sum_{\sigma \text{ in G}} (\text{fixed colorings of } \sigma)$$

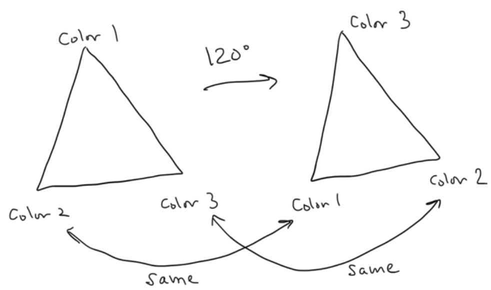
Here, $|G|$ denotes the size of G, the sum is taken as σ ranges throughout all elements of G, and a "fixed coloring" of σ is a coloring which σ leaves unchanged. So, Burnside's Lemma expresses the number of distinct colorings in terms of the number of colorings by each group element.

In this course we will take this result for granted. The proof belongs to a course in Abstract Algebra, where the proper setting is in the context of *group actions*. In fancier group-theoretic language, Burnside's Lemma gives a formula for the number of *orbits* of a group action in terms of the sizes of the *fixed sets* of that action, whatever all these terms actually mean.

Example. Let us return to the general problem of coloring the vertices of a triangle with n colors up to rotational symmetries. The permutation group in this case is

$$G = \{0^\circ, 120^\circ, 240^\circ\} = \{(1)(2)(3), (123), (132)\}.$$

so $|G| = 3$. Now, we start with n^3 total ways of coloring the vertices of a triangle using n colors. All of these colorings are fixed by the identity permutation (i.e. rotation by 0°), so 0° has n^3 fixed colorings. Now, if a coloring is to be fixed by a 120° rotation, we can see that all vertices must be the same color:



Hence, there are n colorings which are fixed by 120°, coming from the number of ways to choose the one color we will use for all vertices. Similarly, for 240°, all vertices must be the same color in order to be fixed, so there are n fixed colorings in this case also. Hence we get that the number of distinct colorings of the vertices of a triangle with n colors up to rotational symmetries is

$$\frac{1}{3}(\underbrace{n^3}_{\text{fixed by } 0^\circ} + \underbrace{n}_{\text{fixed by } 120^\circ} + \underbrace{n}_{\text{fixed by } 240^\circ}) = \frac{n^3 + 2n}{3}.$$

When $n = 3$, this does give $\frac{1}{3}(3^3 + 2(3)) = \frac{33}{3} = 11$, agreeing with the number we found by brute-force earlier.

Suppose now we that we allow reflections as symmetries as well, so that

$$G = \{(1)(2)(3), (123), (132), (12)(3), (13)(2), (1)(24)\}.$$

Here we have $|G| = 6$. The first three terms in the sum we get from Burnside's Lemma are the same as they were for rotations alone. Now, each reflection has n^2 fixed colorings: it is only required that the two vertices which are exchanged be the same color, so the third vertex can be any color whatsoever. (Note the benefit of expressing these operations in terms of cycles: the identity has n^3 fixed colorings since it is made up of 3 cycles overall, each rotation has n^1 fixed colorings since they each consist of a single cycle, and the reflections have n^2 fixed colorings each since each consists of 2 cycles overall.) Thus we get

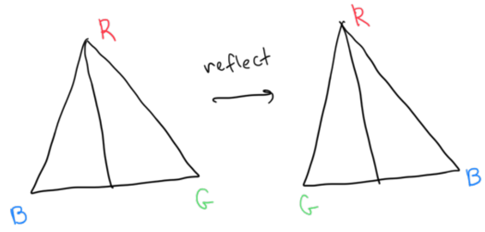
$$\frac{1}{6}(n^3+n+n+n^2+n^2+n^2) = \frac{n^3+3n^2+2n}{6}$$

distinct colorings under the full symmetry group, which includes reflections.

Note that when $n = 3$ this latter number is

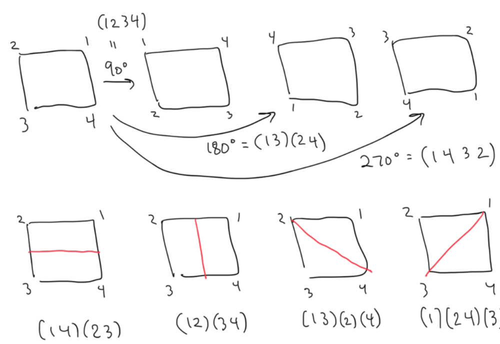
$$\frac{1}{6}(3^3+3(3)^2+2(3)) = \frac{27+27+6}{6} = \frac{60}{6} = 10,$$

which is one less than the 11 we got when considering rotations alone. So, what are the colorings which are distinct under rotations by become the same if we allow the reflections? They are the ones where we color all vertices a different color:



Neither of these can be obtained from the other under a rotation, but they can be obtained from each other under a reflection, so there should be one less distinct coloring if we allow relfections.

Another example. Now we consider the same problem for the vertices of a square. First we describe the symmetries. It will help to label the vertices $1, 2, 3, 4$. There are four rotations, by angles $0°, 90°, 180°, 270°$ and there are four reflections, across horizontal/vertical lines or diagonals:



First take G to consist of the rotations alone. The identity rotation $0°$ has all n^4 possible colorings as fixed colorings. Each of $90° = (1234)$ and $270° = (1432)$ have n fixed colorings since for each of these all vertices are required to be the same color. (Again, note the relation between the exponent of n which occurs and the number of cycles used to describe that permutation.) But, under rotation by $180° = (13)(24)$, only vertices directly opposite one another must be of the same color, so we get n^2 fixed colorings here. Thus, the number of distinct colorings under rotational symmetries is

$$\frac{1}{4}(\underbrace{n^4}_{\text{fixed by }0°} + \underbrace{n}_{\text{fixed by }90°} + \underbrace{n^2}_{\text{fixed by }180°} + \underbrace{n}_{\text{fixed by }270°}) = \frac{n^4+n^2+2n}{4}.$$

If we also allow reflections, then $|G| = 8$. Reflections across the horizontal or vertical line have n^2 fixed colorings, since only the two corners being exchanged in each case are required to be of the same color. But reflections across one of the diagonals have n^3 fixed colorings since the two vertices being left unchanged can be any color, and the two which are exchanged must be the same color. Thus we get:

$$\frac{1}{8}(\underbrace{n^4+n+n^2+n}_{\text{rotations}}+\underbrace{n^2+n^2+n^3+n^3}_{\text{reflections}}) = \frac{n^4+2n^3+3n^2+2n}{8}$$

distinct colorings of the vertices of a square up to all symmetries of the square.

Lecture 18: Ordinary Generating Functions

Warm-Up. We determine the number of ways of distributing 6 identical balls into 3 identical boxes, where we allow for the possibility that some boxes might be empty. Of course, this is a type of problem we considered earlier when discussing integer partitions, only that now we allow a term to be 0. Thus, we are asking for the number of *weak partitions* of 6 into 3 parts. We can list them all out directly as follows:

$$6+0+0, 5+1+0, 4+2+0, 4+1+1, 3+3+0, 3+2+1, 2+2+2,$$

so there are 7 ways of doing what's being asked.

However, the point here is not really to derive this number via a brute-force computation, but rather to see how we can use Burnside's Lemma to approach this and similar questions. The point is that this is not a problem where we are talking about "coloring" anything in a literal sense, but if we interpret "color" in the correct way we will see that this is susceptible to a Burnside's Lemma approach, hinting at the multitude of ways in which Burnside's Lemma can be used apart from literal coloring problems.

Label our three boxes $1, 2, 3$. By "color" let us mean the number of balls in a given box, so a "coloring" of the boxes is a distribution of balls into the boxes. Saying that the balls are identical means that we don't care about which box happens to have, say, 4 balls, which has 2, and which has 0, and this we can interpret by saying that we only care about "coloring" up to a permutation of the boxes; if we have $4, 2, 0$ balls in boxes $1, 2, 3$ respectively, then the numbers obtained by any permutation of these three, say $2, 0, 4$ into boxes $1, 2, 3$ respectively, should be treated as being the same distribution. So, we seek to determine the number of distinct "colorings" up to all possible permutations of the boxes, and phrased this way it is clear that Burnside's Lemma should be applicable. Let the permutation group in question be

$$G = \{(1)(2)(3), (12)(3), (13)(2), (1)(23), (123), (132)\}$$

which hence considers all permutations of our three boxes. We thus get

$$\substack{\text{\# of ways of} \\ \text{distributing the balls} \\ \text{up to permutations}} = \frac{1}{6}\sum_{\sigma \text{ in } G} (\text{\# of distributions fixed by } \sigma).$$

First we consider the identity permutation $(1)(2)(3)$, which leaves all boxes alone. A "coloring" $a+b+c$ (meaning a balls into the first box, b into the second, and c into the third) is fixed under this permutation if we get the exact same coloring after performing that permutation. But in this case, we permute nothing, so any possible values of a, b, c give a fixed distribution. So, the number of fixed distributions (or colorings) is the number of ways of distributing 6 identical balls into 3 *distinct* boxes (distinct since we are not permutating anything in this case), which is hence the number of weak compositions of 6 into 3 parts. So, there are

$$\binom{6+3-1}{3-1} = \binom{8}{2} = 28$$

colorings fixed by the identity permutation.

Next, for each permutation where we fix one box and exchange two (so, $(12)(3), (13)(2)$, and $(1)(23)$), the boxes being permuted must have the same number of balls in order to give a fixed coloring. For instance, if we distribute according to $a+b+c$ and we exchange the first two boxes, we get $b+a+c$, so in order for this to be the same as $a+b+c$ we need $a=b$. Thus there are 4 possibilities for each of these types of permutations: the two boxes being exchanged get 0 balls and the other gets 6, the two boxes being exchanged get 1 ball and the other 4, the two get 2 balls and the other 2 as well, or the two gets 3 balls and the other 0.

Finally, for (123) and (132) we need all boxes to have the same number of balls in order to be fixed: with $a+b+c$ we permute to get $c+a+b$ in the first case and $b+c+a$ in the second, and either way to be the same as $a+b+c$ we need $a=b=c$. Thus each of these permutations only has one fixed coloring, namely $2+2+2$. Hence the number of ways of distributing the balls up to permutations of the boxes is:

$$\frac{1}{6}(\underbrace{28}_{\text{identity}} + \underbrace{4+4+4}_{\text{2-cycle,1-cycle}} + \underbrace{1+1}_{\text{3-cycle}}) = \frac{1}{6}(42) = 7,$$

exactly as we determined earlier by brute-force. You can imagine that if we distribute n identical balls into k identical boxes with n and k much larger than 6 and 3 respectively, listing out all the possibilities by brute-force will become almost impossible, but this Burnside's approach has a better hope of working out.

Where we are. And so ends the second "chunk" of the course, where we focused on developing a few specific types of counts, namely compositions, set and integer partitions, cycles in permutations, and some related concepts. Now we move on to our final topic: generating functions. This topic will provide a new perspective on many things we've already looked at, and will allow us to solve both some new problems and some old problems in a simpler way. A fair warning: in my experience this is where the difficulty of the course seems to jump, maybe not by a huge amount but definitely by a noticeable amount. To key to working with generating functions is, of course, to understand what they allow us to do, which boils down to expressing an infinite amount of data via a single function.

Example. Here is a first type of application. Define a sequence of numbers by setting

$$a_0 = 1 \text{ and } a_{n+1} = 3a_n + 2^n \text{ for } n \geq 0.$$

Our goal is to find a closed form formula for a_n, meaning an exact, non-recursive expression. If we work out a few of these numbers we get:

$$a_0 = 1, a_1 = 3(1) + 2^0 = 4, a_2 = 3(4) + 2^1 = 14, a_3 = 3(14) + 2^2 = 46, a_4 = 3(46) + 2^3 = 146,$$

but there is no immediately-recognizable pattern to these values. So, we need a better technique than simply guessing at an answer and then trying to justify it via, say, induction.

Here is what we do. First we multiply our recursive identity $a_{n+1} = 3a_n + 2^n$ through by x^{n+1}:

$$a_{n+1}x^{n+1} = 3a_n x^{n+1} + 2^n x^{n+1}$$

This is actually not just a single identity, but a whole collection of identities, one for each $n \geq 0$. So, we can add up both sides over all values of $n \geq 0$ to get:

$$\sum_{n=0}^{\infty} a_{n+1}x^{n+1} = \sum_{n=0}^{\infty} 3a_n x^{n+1} + \sum_{n=0}^{\infty} 2^n x^{n+1}.$$

The point is that in order for this in order for two infinite sums like these to be equal, the coefficient of x^n on both sides must agree for all n; on the left side this coefficient is a_{n+1} whereas on the right it is $3a_n + 2^n$, so we recover the recursive identity $a_{n+1} = 3a_n + 2^n$ for our sequence via the equality among series given above.

Ordinary Generating Functions. The *ordinary generating function* of a sequence a_n is the function $F(x)$ defined by the power series

$$F(x) = \sum_{n=0}^{\infty} a_n x^n.$$

That is, $F(x)$ is the function whose power series coefficients are precisely the terms in the sequence a_n. The upshot is that we can encode the entire sequence a_n via one single function, and that by studying this one function we can hope to understanding properties of the sequence a_n which might have been difficult to do without the aid of the generating function. The term "ordinary" refers to the specific type of series we're looking at; later we will talk about *exponential generating functions*, where we'll look at a slightly modified series.

For example, we have

$$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \cdots = \sum_{n=0}^{\infty} x^n$$

as the standard geometric series. The coefficients of the powers of x on the right are all 1, so would we say that $\frac{1}{1-x}$ is the ordinary generating function of the sequence $1, 1, 1, 1, \ldots$ of all 1's. If we replace x by, say, $2x$, we get

$$\frac{1}{1-2x} = \sum_{n=0}^{\infty} (2x)^n = \sum_{n=0}^{\infty} 2^n x^n,$$

so that $\frac{1}{1-2x}$ is the ordinary generating function of the sequence 2^n. Again, this means simply that $\frac{1}{1-2x}$ is the function which reproduces the sequence 2^n via its power series coefficients.

Back to example. So, let $F(x) = \sum_{n=0}^{\infty} a_n x^n$ be the ordinary generating function of the sequence we're considering, namely the one defined recursively by $a_0 = 1$ and $a_{n+1} = 3a_n + 2^n$ for $n \geq 0$. We have so far rephrased this recursive identity in terms of the series equality

$$\sum_{n=0}^{\infty} a_{n+1}x^{n+1} = \sum_{n=0}^{\infty} 3a_n x^{n+1} + \sum_{n=0}^{\infty} 2^n x^{n+1}.$$

Now the goal is to interpret this equality in terms of the generating function $F(x)$, and eventually see what function $F(x)$ is concretely. The left side is

$$a_1 x + a_2 x^2 + a_3 x^3 + \cdots$$

which is the same as

$$F(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \cdots$$

except that the a_0 term is missing, so the left side of our series equality is $F(x) - a_0$:

$$F(x) - a_0 = \sum_{n=0}^{\infty} 3a_n x^{n+1} + \sum_{n=0}^{\infty} 2^n x^{n+1}.$$

The first on the right:

$$\sum_{n=0}^{\infty} 3a_n x^{n+1}$$

can be written as

$$3x \sum_{n=0}^{\infty} a_x^n$$

and so is equal to $3xF(x)$. The point here was that in order to relate this sum to the one defining $F(x)$, we needed to have terms of the form $a_n x^n$ where the index on a_n matches the exponent of x^n, and to obtain this we factored out 3 and one x. The remaining term on the right

$$\sum_{n=0}^{\infty} 2^n x^{n+1}$$

can be written as

$$x \sum_{n=0}^{\infty} 2^n x^n = x \sum_{n=0}^{\infty} (2x)^n,$$

which equals the function $x\left(\frac{1}{1-2x}\right) = \frac{x}{1-2x}$.

Thus, the series equality

$$\sum_{n=0}^{\infty} a_{n+1} x^{n+1} = \sum_{n=0}^{\infty} 3a_n x^{n+1} + \sum_{n=0}^{\infty} 2^n x^{n+1}$$

becomes the function equality

$$F(x) - a_0 = 3xF(x) + \frac{x}{1-2x}, \text{ or } F(x) - 1 = 3xF(x) + \frac{x}{1-2x}$$

after we use $a_0 = 1$. From this equality we can now explicitly solve for $F(x)$, thereby obtaining

$$F(x) = \frac{1}{1-3x} + \frac{x}{(1-3x)(1-2x)}$$

as the explicit ordinary generating function of the sequence a_n we are considering. Now, if we can compute the explicit power series representation of this function, we can read off the explicit formula for a_n we're after as the coefficient of x^n. The first part of $F(x)$ is a standard geometric series:

$$\frac{1}{1-3x} = \sum_{n=0}^{\infty} 3^n x^n, \text{ so } F(x) = \sum_{n=0}^{\infty} 3^n x^n + \frac{x}{(1-3x)(1-2x)}.$$

For the second term, we use a *partial fraction decomposition*:

$$\frac{x}{(1-3x)(1-2x)} = \frac{1}{1-3x} - \frac{1}{1-2x}.$$

(This comes from writing $\frac{x}{(1-3x)(1-2x)} = \frac{A}{1-3x} + \frac{B}{1-2x}$ for to-be-determined A and B, and then cross multiplying to find the necessary values of A and B.) Now each remaining term is a geometric series:

$$\frac{1}{1-3x} - \frac{1}{1-2x} = \sum_{n=0}^{\infty} 3^n x^n - \sum_{n=0}^{\infty} 2^n x^n,$$

and so altogether we get:

$$\begin{aligned} F(x) &= \frac{1}{1-3x} + \frac{x}{(1-3x)(1-2x)} \\ &= \frac{1}{1-3x} + \frac{1}{1-3x} - \frac{1}{1-2x} \\ &= \sum_{n=0}^{\infty} 3^n x^n + \sum_{n=0}^{\infty} 3^n x^n - \sum_{n=0}^{\infty} 2^n x^n \\ &= \sum_{n=0}^{\infty} (2 \cdot 3^n - 2^n) x^n. \end{aligned}$$

Hence, our generating function should be given by this final series, but at the same time it should be given by our original definition:

$$\sum_{n=0}^{\infty} (2 \cdot 3^n - 2^n) x^n = F(x) = \sum_{n=0}^{\infty} a_n x^n.$$

Two power series are equal precisely when they have the same coefficients, so we get that

$$a_n = 2 \cdot 3^n - 2^n$$

is the closed formula for the terms in our sequence a_n. You can verify that this does give the correct values we computed previously: $a_0 = 1, a_1 = 4, a_2 = 14, a_3 = 46, a_4 = 146$. The overarching idea, again, is to encode an entire sequence via a single power series function, and then to study that function in order to derive information about the sequence.

Another example. Here is one more example. Consider the sequence defined by

$$a_0 = 0, \ a_{n+1} = a_n + 1 \text{ for } n \geq 0.$$

Now in this case, the explicit formula for a_n is easy to come by: we start with 0 and add 1 each time, so by the time we get to a_n we will have added 1 to itself n times, and hence $a_n = n$. But, let us see how we get this same value via a generating function approach.

Set $F(x) = \sum_{n=0}^{\infty} a_n x^n$. After multiplying the recursive definition of a_n through by x^{n+1} and summing over the possible values of n we get:

$$\sum_{n=0}^{\infty} a_{n+1} x^{n+1} = \sum_{n=0}^{\infty} a_n x^{n+1} + \sum_{n=0}^{\infty} x^{n+1}.$$

The left side is $F(x) - a_0$, the first term on the right (after factoring out an x) is $xF(x)$, and the final term (after factoring out x and using a geometric series) is $\frac{x}{1-x}$:

$$F(x) - a_0 = xF(x) + \frac{x}{1-x}.$$

Using $a_0 = 0$, we can solve for $F(x)$ to get

$$F(x) = \frac{x}{(1-x)^2}$$

as the explicit generating function of our sequence.

Now, to express this as a power series, we can start with the standard geometric series:

$$\frac{1}{1-x} = \sum_{n=0}^{\infty} x^n,$$

take a derivative:

$$\frac{1}{(1-x)^2} = \sum_{n=1}^{\infty} nx^{n-1},$$

and multiply by x:

$$\frac{x}{(1-x)^2} = \sum_{n=1}^{\infty} nx^n.$$

Thus our generating function is

$$F(x) = \frac{x}{(1-x)^2} = \sum_{n=1}^{\infty} nx^n,$$

and so we get $a_n = n$ as the coefficient of x^n, as expected. Note that a_0 does equal 0 here, since there is no constant term in this power series—i.e. we get the same series if we start indexing at $n = 0$ instead of $n = 1$.

Lecture 19: Products of Generating Functions

Warm-Up. We find the ordinary generating function for the sequence of Fibonacci numbers defined by

$$f_0 = 0, \; f_1 = 1, \text{ and } f_{n+2} = f_{n+1} + f_n \text{ for all } n \geq 0.$$

Let $F(x) = \sum_{n=0}^{\infty} f_n x^n$ be the generating function we want. We multiply the recursive identity above through by x^{n+2} and sum over the values of n to get:

$$\sum_{n=0}^{\infty} f_{n+2}x^{n+2} = \sum_{n=0}^{\infty} f_{n+1}x^{n+2} + \sum_{n=0}^{\infty} f_n x^{n+2}.$$

(We multiplied through by x^{n+2} in order to get the largest index which appears in the recursive identity, which is $n+2$, match with the corresponding exponent.) The left side is almost the series for $F(x)$, except it is missing the terms corresponding to $n = 0$ and $n = 1$ in the definition of $F(x)$:

$$\sum_{n=0}^{\infty} f_{n+2}x^{n+2} = F(x) - f_0 - f_1 x.$$

The first term on the right in the equality above is $x(F(x) - f_0)$, and the final term is $x^2F(x)$, so we get the identity:

$$F(x) - f_0 - f_1 x = x(F(x) - f_0) + x^2F(x), \text{ or } F(x) - x = xF(x) + x^2F(x)$$

after we use the initial values $f_0 = 0$ and $f_1 = 1$. Thus we get

$$F(x) = \frac{x}{1 - x - x^2}$$

as the ordinary generating function of the Fibonacci numbers.

Now, after factoring the denominator, you could go further and write this in terms of a partial fraction decomposition, and then use a geometric series to write each resulting term as a series, with the aim of determining explicitly what the coefficient of x^n will be in this power series represetnation, since this will then give an explicit closed formula for the n-th Fibonacci number. We will not do this here since the algebra is a bit tedious (although nothing about it is difficult), but you will work it out on a problem in discussion section. (Actually, the problem will in discussion we be a shifted version of this sequence, where the 0-th term will be 1 instead of 0 as it is here, which will lead to a slightly simpler partial fraction decomposition to work with.) The answer, in case you've never seen an explicit formula for the n-th Fibonacci numbers before, is:

$$f_n = \frac{1}{\sqrt{5}} \left(\frac{1 + \sqrt{5}}{2} \right)^n - \frac{1}{\sqrt{5}} \left(\frac{1 - \sqrt{5}}{2} \right)^n.$$

This approach via generating functions is truly quite powerful indeed!

Other known generating functions. As a reminder, let us summarize now some generating functions we saw previously, now that we can put them in the proper context:

$$\sum_{n=0}^{\infty} p(n) x^n = \prod_{k=1}^{\infty} \frac{1}{1 - x^k}$$

is the ordinary generating function for the sequence $p(n)$ of integer partitions, and

$$\sum_{n=0}^{n} c(n,k) x^k = x(x+1) \cdots (x+n-1)$$

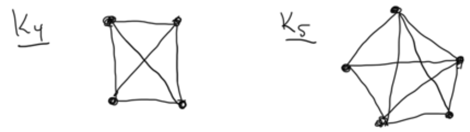
is the ordinary generating function for the sequence $c(n,k)$ of unsigned Stirling numbers with n fixed and k varying. Note that since $c(n,k) = 0$ for $n < k$, this latter finite sum can indeed be viewed as an infinite series:

$$\sum_{n=0}^{\infty} c(n,k) x^k = x(x+1) \cdots (x+n-1),$$

only one where all coefficients beyond x^n are zero. We will see later how we can interpret other numbers we've seen (Stirling numbers of the second kind $S(n,k)$ and derangements $D(n)$ for instance) in terms of generating functions as well.

An extra, fun problem. And now allow me to go off on a bit of a tangent and look at a problem which, although not so relevant for what we're doing now, does make use of the generating function for $c(n,k)$ and is one which ties some different concepts together. In the end, this is a problem which makes use of Burnside's Lemma, cycles in permutations, generating functions, $\binom{n}{k}$ notation, and compositions, which is why it's worth mentioning. You should not expect something like this on an exam, however, but it does give a sense of how none of what we've been doing exists in isolation—it's all connected!

Recall that K_n denotes the *complete graph on n vertices*, which is the graph formed by taking n vertices and drawing an edge between each pair of vertices; in other words, take a regular n-gon and draw in all possible diagonals:



This is something we briefly look at in some examples during the second week. We color the vertices of K_n using x colors, only that we declare two such colorings to be the "same" if one can be obtained from the other via a permutation of the vertices. For instance, if we color one vertex red, another blue, and all the rest green, this will be the same coloring as if we had started by coloring some different vertex red, some other one blue, and all the rest green since this latter coloring can be obtained from the former by permuting the vertices in a correct way. We want to determine the number of distinct colorings up to such permutations.

By Burnside's Lemma, with $n!$ permutations in total, the number of distinct colorings is:

$$\frac{1}{n!}\sum_{\sigma}(\text{colorings fixed by } \sigma).$$

Now, if we take a permutation like $(123)(45)(6789)$ in the $n = 9$ case, all vertices which occur in the same cycle must be of the same color in order to be fixed, meaning that $1, 2, 3$ have the same color, $4, 5$ have the same color, and $6, 7, 8, 9$ have the same color. Thus in this case there are x^3 possible fixed colorings, one x for each cycle. In general, for a permutation with k cycles overall, there are x^k fixed colorings for the same reason: all vertices within the same cycle must have the same color. So, there are $c(n, 1)$ permutations with x^1 fixed colorings, $c(n, 2)$ with x^2 fixed colorings, and in general

$$c(n, k) \text{ permutations with } x^k \text{ fixed colorings.}$$

Thus Burnside's Lemma gives the number of distinct colorings as

$$\frac{1}{n!}\sum_{k=1}^{\infty} c(n,k)x^k$$

However, we know how to evaluate this using the generating function for $c(n, k)$! We get:

$$\#\text{ of distinct colorings} = \frac{1}{n!}\sum_{k=1}^{\infty} c(n,k)x^k = \frac{1}{n!}x(x+1)\cdots(x+n-1).$$

Now, the resulting expression can be written in terms of factorials as:

$$\frac{1}{n!}x(x+1)\cdots(x+n-1) = \frac{1}{n!}\frac{(x+n-1)!}{(n-1)!},$$

and so is precisely the same as $\binom{x+n-1}{n-1}$. But this is the number of weak compositions of $x + n - 1$ into $n - 1$ parts, so finally we get

$$\begin{matrix}\#\text{ of distinct colorings of the} \\ \text{vertices of } K_n \text{ with } x \text{ colors}\end{matrix} = \begin{matrix}\#\text{ of weak compositions of} \\ x+n-1 \text{ into } n-1 \text{ parts}\end{matrix}$$

Seeing what we get such a nice answer, we should ask whether it makes sense, in the sense of whether we could have seen that this is the correct answer without making use of Burnside's lemma? But of course we can: in the end, all that matter is how many vertices are colored "red", how many are colored "blue" and so on, so that we can rephrase this problem as the one of distributing our n vertices into the x boxes representing the colors; the boxes are distinct since the colors matter, and it is possible for a color to not be used at all, which is why we get weak compositions. So, the point of this problem wasn't really to get the answer, but rather to see how it be approached in a way which ties different concepts together.

A new problem. And now we come back to things which are more relevant. Say we have n people arranged in a line. We want to split these people into two teams while maintaining their order, meaning that the first team consists of some number of people from the start of the line and the second consists of the remaining people, but we cannot have the first team for instance consisting of the first person and the last person without also taking all people in between; in other words, the teams are formed by simply splitting the line in two somewhere. We choose a captain from the first team, and two co-captains from the second team, with no restrictions on the team sizes apart from the fact that the first team must have at least one person and second at least two, due to the required captain and co-captains.

We ask for the total number of ways in which this entire process can be carried out; let c_n denote the number of ways of doing so. First, we can express c_n as a sum over the possible size k of the first team. If this first team has k people, where $1 \leq k \leq n-2$, then there are k ways of picking its captain. This leaves $n-k$ people ($n-k$ is at least 2) for the second team, and so there are $\binom{n-k}{2}$ ways of picking the two co-captains from this second team. This gives

$$c_n = \sum_{k=1}^{n-2} k \binom{n-k}{2}$$

as an expression for c_n. Of course, it would be nice to have something more explicit, which will come from finding an explicit form of the generating function for c_n:

$$C(x) = \sum_{n=3}^{\infty} c_n x^n = \sum_{n=3}^{\infty} \left(\sum_{k=1}^{n-2} k \binom{n-k}{2} \right) x^n.$$

Note that we start this series at $n = 3$ since $c_0 = c_1 = c_2 = 0$ because in these cases there are not enough people available to form the required teams.

Products of generating functions. We'll come back to the problem at hand in a second, but let us first compute something important before moving on. Given two power series, we want to figure out how to write their product as a new power series:

$$\left(\sum_{n=0}^{\infty} a_n x^n \right) \left(\sum_{n=0}^{\infty} b_n x^n \right) = \sum_{n=0} \underline{\ ?\ } \, x^n.$$

That is, if we multiply out the expression

$$(a_0 + a_1 x + a_2 x^2 + \cdots)(b_0 + b_1 x + b_2 x^2 + \cdots)$$

how do we describe the resulting coefficients? The resulting constant term is $a_0 b_0$. Now, there are two ways to get x^1: a_0 times $b_1 x$ and $a_1 x$ times b_0:

$$(a_0 + a_1 x + a_2 x^2 + \cdots)(b_0 + b_1 x + b_2 x^2 + \cdots) = a_0 b_0 + (a_0 b_1 + a_1 b_0) x + \cdots$$

There are three ways to get x^2: a_0 times b_2x^2, a_1x times b_1x, and a_2x^2 times b_0:

$$(a_0 + a_1x + \cdots)(b_0 + b_1x + \cdots) = a_0b_0 + (a_0b_1 + a_1b_0)x + (a_0b_2 + a_1b_1 + a_2b_0)x^2 + \cdots$$

And so on, in general the coefficient of x^n is $a_0b_n + a_1b_{n-1} + \cdots + a_nb_0$. Thus:

$$\left(\sum_{n=0}^{\infty} a_n x^n\right)\left(\sum_{n=0}^{\infty} b_n x^n\right) = \sum_{n=0}^{\infty}\left(\sum_{k=0}^{n} a_k b_{n-k}\right) x^n$$

is the required product.

The point is that if $A(x)$ is the ordinary generating function for a sequence a_n, and $B(x)$ is the ordinary generating function for a sequence b_n, then the computation above shows that their product $A(x)B(x)$ is the ordinary generating function for the sequence $c_n = \sum_{k=0}^{n} a_k b_{n-k}$. As we saw in the problem we were just looking at, such sums c_n arise in counting problems formed by making a sequence of successive choices (i.e. pick a captain, then pick two co-captains) while maintaining order, and so the result is that the generating function for such a sequence of choices is obtained by multiplying the functions for the individual choices together:

$$\begin{pmatrix}\text{generating function}\\ \text{for doing thing } A\end{pmatrix}\begin{pmatrix}\text{generating function}\\ \text{for doing thing } B\end{pmatrix} = \begin{matrix}\text{generating function for}\\ \text{doing thing } A \text{ to a first “chunk”}\\ \text{and thing } B \text{ to a second “chunk”}\end{matrix}$$

In our motivating problem, "thing A" refers to picking a captain for a team, and "thing B" refers to picking two co-captains for a team. In general, whenever we have a problem phrased in terms of making a sequence of successive choices (i.e. do something to a first "chunk", something to a second "chunk", something to a third, and so on) which maintain order, there is a way to interpret it in terms of products of ordinary generating functions.

You might ask: what if we don't want to maintain order, and allow the possibility that the "chunks" above can come from any of our n objects without regard to any ordering? The answer to such problems will come from multiplying *exponential generating functions* together, which we'll get to soon enough.

Back to our problem. Recall that we ended up with the generating function

$$C(x) = \sum_{n=3}^{\infty} c_n x^n = \sum_{n=3}^{\infty}\left(\sum_{k=1}^{n-2} k\binom{n-k}{2}\right) x^n$$

for the number of ways c_n of splitting n people into teams while maintain their order, picking a captain from the first and two co-captains from the second. But now we know that the resulting expression is precisely the product of two individual generating functions:

$$\sum_{n=3}^{\infty}\left(\sum_{k=1}^{n-2} k\binom{n-k}{2}\right) x^n = \left(\sum_{k=1}^{\infty} kx^k\right)\left(\sum_{\ell=2}^{\infty}\binom{\ell}{2}x^\ell\right).$$

On the right, $\sum kx^k$ is the generating function for the number of ways of picking a captain from a team (thing A), and $\sum \binom{\ell}{2}x^\ell$ is the generating function for the number of ways of picking two co-captains from a team (thing B), and so this is a reflection of the fact that their product is the generating function for the number of ways doing thing A to a first chunk and thing B to a second chuck, which is precisely the problem at hand.

Now, we have:

$$\sum_{k=1}^{\infty} kx^k = x\left(\sum_{k=0}^{\infty} x^k\right)' = x\left(\frac{1}{1-x}\right)' = \frac{x}{(1-x)^2}, \text{ and}$$

$$\sum_{\ell=2}^{\infty} \binom{\ell}{2} x^\ell = \sum_{\ell=2}^{\infty} \frac{\ell(\ell-1)}{2} x^\ell = \frac{x^2}{2}\left(\sum_{\ell=0}^{\infty} x^\ell\right)'' = \frac{x^2}{2}\left(\frac{1}{1-x}\right)'' = \frac{x^2}{(1-x)^3}.$$

Thus, the generating function for our sequence c_n (do thing A and then thing B while maintaining order) is

$$\underbrace{\frac{x}{(1-x)^2}}_{\text{thing } A} \underbrace{\frac{x^2}{(1-x)^3}}_{\text{thing } B} = \frac{x^3}{(1-x)^5}.$$

So, if we can determine how to write this final function as an explicit series, we will find an explicit formula for c_n as the coefficient of x^n. This will involve computing more derivatives of $\frac{1}{1-x}$, which will do next time.

Lecture 20: More on Products

Warm-Up 1. Suppose n people are arranged in a line. We distribute one of two hats (red or blue) to the first however many of them (maintaining order, but no restriction on how many), and one of three coats (brown, black, gray) to the rest. In how many ways can this be done? One possible answer is

$$\sum_{k=0}^{n} 2^k 3^{n-k}$$

where we sum over the number of people who receive hats: for k people there are 2^k possibilities with two hats, and then 3^{n-k} possibilities for distributing three coats among the remaining $n-k$ people. But, we want to find a more explicit closed formula for this number.

This calls for a product of generating functions, since we do one thing (distribute hats) to a first "chunk" and then do a second thing (distribute coats) to a second "chunk". If $F(x)$ is the generating function for distributing two hats and $G(x)$ the generating function for distributing three cots, then $F(x)G(x)$ is the generating function for the problem at hand. We have:

$$F(x) = \sum_{n=0}^{\infty} 2^n x^n = \frac{1}{1-2X}$$

since there are 2^n ways of distributing two hats to n people, and

$$G(x) = \sum_{n=0}^{\infty} 3^n x^n = \frac{1}{1-3x}$$

since there are 3^n ways of distributing three coats to n people. Again, the point is that all the possible ways of combining both of these operations (give first two people hats and the rest coats, or first 5 people hats and the rest coats, etc) are encoded in the fact that we will take a product, so we don't have to consider these combinations ourselves. We thus get

$$F(x)G(x) = \frac{1}{(1-2x)(1-3x)}$$

as the generating function for our problem. Using partial fractions we get:

$$\begin{aligned}\frac{1}{(1-2x)}\frac{1}{(1-3x)} &= -\frac{2}{1-2x}+\frac{3}{1-3x}\\ &= -2\sum_{n=0}^{\infty}2^n x^n + 3\sum_{n=0}^{\infty}3^n x^n\\ &= \sum_{n=0}^{\infty}(3^{n+1}-2^{n+1})x^n.\end{aligned}$$

Thus, there are $3^{n+1}-2^{n+1}$ ways of distributing our hats and coats to n people. For instance, with $n=1$ person, either that one person can get hats in 2 ways, or coats in 3 ways, giving $2+3=5$ possibilities, which does equal 3^2-2^2; with $n=2$ people, either both can get hats in $2\cdot 2=4$ ways, both can get coats in $3\cdot 3=9$ ways, or the first can get hats in 2 ways the second coats in 3 ways, giving $4+9+2\cdot 3=19$ possibilities, which is 3^3-2^3.

With such a nice expression $3^{n+1}-2^{n+1}$ for the answer, it makes sense to ask whether we can see this is the correct value without the use of generating functions. ***FINISH***

Warm-Up 2. We finish the motivating problem from last time, forming two teams and picking a captain for the first and two co-captains for the second, by expressing the generating function we obtained

$$\frac{x^3}{(1-x)^5}$$

as a power series. Recall that this function came from multiplying the generating function for picking a captain:

$$\sum_{n=0}^{\infty} nx^n = \frac{x}{(1-x)^2}$$

with the generating function for picking two co-captains:

$$\sum_{n=0}^{\infty}\binom{n}{2}x^n = \sum_{n=0}^{\infty}\frac{n(n-1)}{2}x^n = \frac{x^2}{(1-x)^3}.$$

To build up to $\frac{1}{(1-x)^5}$ we need four derivatives of $\frac{1}{1-x}$:

$$\begin{aligned}\text{1-st derivative: }\ \frac{1}{(1-x)^2} &= \sum_{n=1}^{\infty} nx^{n-1}\\ \text{2-nd derivative: }\ \frac{2}{(1-x)^3} &= \sum_{n=2}^{\infty} n(n-1)x^{n-2}\\ \text{3-rd derivative: }\ \frac{2\cdot 3}{(1-x)^4} &= \sum_{n=3}^{\infty} n(n-1)(n-2)x^{n-3}\\ \text{4-th derivative: }\ \frac{2\cdot 3\cdot 4}{(1-x)^5} &= \sum_{n=4}^{\infty} n(n-1)(n-2)(n-3)x^{n-4}.\end{aligned}$$

Thus

$$\frac{x^3}{(1-x)^5} = \frac{x^3}{4!}\frac{2\cdot 3\cdot 4}{(1-x)^5} = \sum_{n=4}^{\infty}\frac{n(n-1)(n-2)(n-3)}{4!}x^{n-1},$$

and if we reindex so that the exponent on x is n we get:

$$\sum_{n=3}^{\infty} \frac{(n+1)n(n-1)(n-2)}{4!} x^n = \sum_{n=3}^{\infty} \binom{n+1}{4} x^n.$$

Hence, the number of ways splitting into two teams, picking a captain from the first and two co-captains from the second is $\binom{n+1}{4}$ when $n \geq 3$, and 0 for $n = 0, 1, 2$.

Here is another way to see that this is correct, without using generating functions. Suppose we draw our people as dots, and indicate where the teams are split by a bar. Then we also put a bar before the captain of the first team, and a bar after each co-captain of the second team. Then we are choosing the 4 locations for bars from $n + 1$ possible spots (one before each dot and including the spot after the last dot), which can be done in $\binom{n+1}{4}$ many ways.

More products. Here is a simple example which requires a product of three generating functions. We want the generating function for the number of ways of making n dollars using only $1, 5$, and 10 dollar bills. For instance, we can make 5 dollars in two ways (use five 1's or one 5) so we should have a $2x^5$ term in our generating function; since we can make 10 dollars in 4 ways (ten 1's, five 1's and one 5, two 5's, one 10), we need $4x^{10}$ in our generating function:

$$1 + \cdots + 2x^5 + \cdots + 4x^{10} + \cdots$$

If we draw the bills we'll use as dots, then we are picking some chunk at the start to represent the 1 dollar bills used, a next chunk to represent the 5 dollar bills used, and a final chunk to represent the 10 dollar bills. Phrasing it this way shows that the ordinary generating function we want is the product of the ordinary generating functions for using only 1 dollar bills, or only 5 dollar bills, or only 10 dollar bills.

If we only use 1 dollar bills, there is 1 way of making n dollars for any n (even if we consider $n = 0$ since there is 1 way of using no 1 dollar bills), so the generating function for this choice is

$$1 + x + x^2 + x^3 + \cdots = \sum_{n=0}^{\infty} x^n = \frac{1}{1-x}.$$

For using only 5 dollar bills, we can only make n dollars if n is a multiple of 5, and in that case there is only 1 way of making $n = 5k$ dollars using only 5 dollar bills. So the generating function for this choice is

$$1 + x^5 + x^{10} + \cdots = \sum_{n=0}^{\infty} (x^5)^n = \frac{1}{1-x^5}.$$

(Again, something like x^7 doesn't appear, or has coefficient 0, since there are 0 ways of making 7 dollars with only 5 dollar bills.) Similarly, with only 10 dollar bills we can only make a multiple of 10 dollars, so we get the function

$$1 + x^{10} + x^{20} + \cdots = \sum_{n=0}^{\infty} (x^{10})^n = \frac{1}{1-x^{10}}.$$

Thus the generating function for our problem is

$$\frac{1}{(1-x)(1-x^5)(1-x^{10})},$$

so that the number of ways of making n dollars is the coefficient of x^n obtained when expanding this as a poser series. Once again, the fact that we can combine these different choices, say we can make 10 dollars in 4 ways, is encoded in the way which products of series work.

Revisiting partitions. And now having developed the idea of taking products of generating functions, we can recast the generating function for the sequence of integer partitions in a new light. Recall that this function was:

$$\sum_{n=0}^{\infty} p(n)x^n = (1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots)\cdots ,$$

which using the fact that $1 + x^k + x^{2k} + \cdots = \frac{1}{1-x^k}$ can be more succinctly written as

$$\sum_{n=0}^{\infty} p(n)x^n = \prod_{k=1}^{\infty} \frac{1}{1 - x^k}.$$

Previously we derived this by interpreting the x^{mk} term in the k-th factor $1 + x^k + x^{2k} + \cdots$ of this product to represent the idea of taking "m copies of the integer k" in a partition of n, and through products of such terms with exponents adding to n we recover all integer partitions of n.

But now we can give another derivation as follows. Consider the problem of counting the number of partitions of n which consist only of 1's. This has generating function

$$1 + x + x^2 + x^3 + \cdots = \frac{1}{1 - x}$$

so, no matter what n is, there is only one way of expressing n as a sum of 1's, so that the coefficient of x^n should be 1. Now, for partitions of n which consist only of 2's, the generating function is

$$1 + x^2 + x^4 + \cdots = \frac{1}{1 - x^2}$$

since in this case there is 1 way of writing n as a sum of 2's if n is even, and 0 ways if n is odd, except that for $n = 0$ there is one way if we interpret the "empty sum" as being zero. (This is a convention.) In general, the generating function for the number of ways of writing n as a sum of k's is

$$1 + x^k + x^{2k} + \cdots = \frac{1}{1 - x^k}$$

since this is only possible (in one way) if n is a multiple of k. Thus, if we interpret the problem of constructing a partition as a sequence of choices:

choose how many 1's to use, choose how many 2's to use, how many 3's, etc

while maintaining order (meaning we will not choose 3's first, and then 1's, and then 6's, but we will choose only in an increasing order), then what we know about products of generating functions says that the generating function for integer partitions should be

(function for partitions of all 1's)(function for partitions of all 2's)(function for all 3's) $\cdots$,

which is indeed thus

$$\frac{1}{1 - x} \cdot \frac{1}{1 - x^2} \cdot \frac{1}{1 - x^3} \cdots = \prod_{k=1}^{\infty} \frac{1}{1 - x^k}$$

Distinct = odd. Now we can finally justify a fact we briefly mentioned when first discussion partitions, that the number of partitions of n into distinct parts is equal to the number of partitions of n into odd parts:

$$p_{distinct}(n) = p_{odd}(n).$$

This was mentioned under the heading "A property still to come" in Lecture 12, and at that point we were not able to give a justification, since it required more than just manipulations of Young diagrams. But now with the machinery of generating functions we can give a proof.

To show that two sequences such as $p_{distinct}(n)$ and $p_{odd}(n)$ are equal, it suffices to show that their generating functions are equal:

$$\sum_{n=0}^{\infty} p_{distinct}(n)x^n = \sum_{n=0}^{\infty} p_{odd}(n)x^n.$$

After all, two power series are equal if and only if they have the same coefficients, so this equality does imply the one we want: $p_{distinct}(n) = p_{odd}(n)$. First we consider the left side. The number of partitions of n into distinct parts which are all equal to k is

$$\begin{cases} 1 & \text{if } n = 0 \text{ or } n = k \\ 0 & \text{otherwise} \end{cases}$$

The $n = 0$ case is simply a convention, and the point is that we cannot get for example $n = 2k$ since this requires multiple k's, and so we wouldn't have distinct parts. Thus the generating function for the number of ways of writing n as a sum of at most one k is

$$1 + x^k.$$

If we make this choice for $k = 1$, then $k = 2$, then $k = 3$, and so on, we get that the generating function for $p_{distinct}(n)$ is

$$\sum_{n=0}^{\infty} p_{distinct}(n)x^n = (1 + x)(1 + x^2)(1 + x^3)\cdots = \prod_{k=1}^{\infty}(1 + x^k)$$

Now, as we have seen, the generating function for the number of ways of writing n as a sum of 1's is $\frac{1}{1-x}$, as sum of 3's is $\frac{1}{1-x^3}$, and in general as a sum of integers which are all $2k+1$ is $\frac{1}{1-x^{2k+1}}$. Thus the generating function for $p_{odd}(n)$ is the product of these:

$$\sum_{n=0}^{\infty} p_{odd}(n)x^n = \prod_{\ell \text{ odd}}^{\infty} \frac{1}{1 - x^{\ell}}$$

and we claim that this is the same as the one for $p_{distinct}(n)$. Indeed, we rewrite the $p_{odd}(n)$ function in a more clever way:

$$\prod_{\ell \text{ odd}}^{\infty} \frac{1}{1 - x^{\ell}} = \frac{\prod_{k=1}^{\infty} \frac{1}{1-x}}{\prod_{k=1}^{\infty} \frac{1}{1-x^{2k}}}.$$

The point is that the right side looks like

$$\frac{(1 + x + x^2 + \cdots)(1 + x^2 + x^4 + \cdots)(1 + x^3 + x^6 + \cdots)(1 + x^4 + x^8 + \cdots)\cdots}{(1 + x^2 + x^4 + \cdots)(1 + x^4 + x^8 + \cdots)\cdots}$$

so that all the factors which involve taking powers of x^k for k even cancel out of this fraction, so that we are only left with those factors where we take powers of x^{odd}, which is what

$$\prod_{\ell \text{ odd}}^{\infty} \frac{1}{1-x^\ell}$$

equals. And now we do some algebraic manipulation, and use the fact that $1-x^{2k} = (1-x^k)(1+x^k)$ along the way:

$$\prod_{\ell \text{ odd}}^{\infty} \frac{1}{1-x^\ell} = \frac{\prod_{k=1}^{\infty} \frac{1}{1-x}}{\prod_{k=1}^{\infty} \frac{1}{1-x^{2k}}} = \prod_{k=1}^{\infty} \frac{\frac{1}{1-x^k}}{\frac{1}{1-x^{2k}}} = \prod_{k=1}^{\infty} \frac{1-x^{2k}}{1-x^k} = \prod_{k=1}^{\infty}(1+x^k).$$

Thus we get

$$\sum_{n=0}^{\infty} p_{distinct}(n)x^n = \prod_{k=1}^{\infty}(1+x^k) = \prod_{\ell \text{ odd}}^{\infty} \frac{1}{1-x^\ell} = \sum_{n=0}^{\infty} p_{odd}(n)x^n$$

as desired so $p_{distinct}(n) = p_{odd}(n)$ for all n. Boom! There are actually ways of showing this without using generating functions, but the generating function approach is much, much cleaner.

Lecture 21: Compositional Formula

Warm-Up 1. (This was actually a problem in discussion section, but the TA slightly misinterpreted the intent: he assumed that the groups could be formed using *any* of the days in *any* order, whereas I intended for the order to be maintained. So, here is the solution I had in mind; the TA's interpretation of the problem is better approaching using products of *exponential generating functions*, which we'll begin to look at next time.)

Take a stretch of n days and split them into three parts while maintaining their order. We then pick any number of days to be holidays from the first group, and odd number of holidays from the second, and an even number from the third. In how many ways can this be done? (We allow for the possibility that the first and third groups have zero days.)

The ordinary generating function for the number of ways in which this can be done can be obtained by taking the product of the generating functions for each choice separately. With a first group of n days, there are 2^n ways of picking some number of them (viewed as a subset of the entire group) as holidays, so this choice has generating function

$$A(x) = \sum_{n=0}^{\infty} 2^n x^n = \frac{1}{1-2x}$$

For the second group, which requires at least 1 day since we want an odd holidays chosen, we use the fact we derived a while ago that a set of n elements has 2^{n-1} subsets which are of odd size (the same as the number of subsets of even size), so the generating function for this choice

$$B(x) = \sum_{n=1}^{\infty} 2^{n-1} x^n = x\sum_{n=1}^{\infty} 2^{n-1} x^{n-1} = x\sum_{n=0}^{\infty} 2^n x^n = \frac{x}{1-2x}$$

Finally, using the fact that a subset of n elements has 2^{n-1} subsets of even size when $n \geq 1$ and 1 such subset when $n = 0$, the generating function for the third choice is:

$$C(x) = 1 + \sum_{n=1}^{\infty} 2^{n-1} x^n = 1 + \frac{x}{1-2x} = \frac{1-x}{1-2x}$$

The thus the ordinary generating function for number of ways in which our groups of holidays can be chosen is

$$A(x)B(x)C(x) = \frac{1}{1-2x} \cdot \frac{x}{1-2x} \cdot \frac{1-x}{1-2x} = \frac{x}{(1-2x)^3} - \frac{x^2}{(1-2x)^3}$$

To express this as a power series, we first express $\frac{1}{(1-2x)^3}$ by differentiating $\frac{1}{1-2x}$:

$$\frac{1}{1-2x} = \sum_{n=0}^{\infty} 2^n x^n, \quad \frac{2}{(1-2x)^2} = \sum_{n=1}^{\infty} n2^n x^{n-1}, \quad \frac{8}{(1-2x)^3} = \sum_{n=2}^{\infty} n(n-1)2^n x^{n-2}$$

Hence

$$\frac{1}{(1-2x)^3} = \sum_{n=2}^{\infty} n(n-1)2^{n-3}x^{n-2},$$

so

$$\frac{x}{(1-2x)^3} = \sum_{n=2}^{\infty} n(n-1)2^{n-3}x^{n-1} = \sum_{n=1}^{\infty}(n+1)n2^{n-2}x^n$$

and

$$\frac{x^2}{(1-2x)^3} = \sum_{n=2}^{\infty} n(n-1)2^{n-3}x^n$$

Hence the ordinary generating function for our problem as a series looks like:

$$\sum_{n=1}^{\infty}(n+1)n2^{n-2}x^n - \sum_{n=2}^{\infty} n(n-1)2^{n-3}x^n = x + \sum_{n=2}^{\infty} n(n+3)2^{n-3}x^n,$$

so the number of ways of choosing the required holidays from a stretch of n days is $n(n+3)2^{n-3}$.

Warm-Up 2. We find the ordinary generating function of the Catalan numbers. For this we will use the Catalan recursion we derived back in the second week:

$$C_n = C_0C_{n-1} + C_1C_{n-2} + \cdots + C_{n-2}C_1 + C_{n-1}C_0 \text{ for } n \geq 1$$

where $C_0 = 1$. Set $C(x) = \sum_{n=0}^{\infty} C_n x^n$ for the generating function we want. The key point is that we can interpret

$$C_n = \sum_{k=0}^{n-1} C_k C_{n-1-k}$$

in terms of a product of generating functions, namely $C(x)$ with itself. Write $C(x)$ as follows:

$$C(x) = \sum_{n=0}^{\infty} C_n x^n = C_0 + \sum_{n=1}^{\infty} C_n x^n = 1 + \sum_{n=1}^{\infty} \left(\sum_{k=0}^{n-1} C_k C_{n-1-k} \right) x^n$$

where we use $C_0 = 1$ and the recursive expression given above for C_n, which is valid for $n \geq 1$. The second term is the following product:

$$\sum_{n=1}^{\infty} \left(\sum_{k=0}^{n-1} C_k C_{n-1-k} \right) x^n = \left(\sum_{n=0}^{\infty} C_n x^n \right) \left(\sum_{n=1}^{\infty} C_{n-1} x^n \right)$$

where we use $\left(\sum a_n x^n\right)\left(\sum b_n x^n\right) = \sum\left(\sum a_k b_{n-k}\right) x^n$, and the two factors on the right respectively are $C(x)$ and

$$\sum_{n=1}^{\infty} C_{n-1}x^n = x\sum_{n=1} C_{n-1}x^{n-1} = x\sum_{n=0}^{\infty} C_n x^n = xC(x)$$

Thus the identity we had above for $C(x)$ becomes

$$C(x) = 1 + C(x)[xC(x)] = 1 + xC(x)^2.$$

Now, treating this as a quadratic with "variable" $C(x)$, the quadratic formula gives:

$$C(x) = \frac{1 \pm \sqrt{1-4x}}{2x}$$

To decide which of $\pm$ we need, we consider what the corresponding series should look like. The power series expansion of $\sqrt{1-4x}$ should look like

$$\sqrt{1-4x} = 1 + (\text{higher order terms in } x)$$

since the constant term should be the value of the function at $x = 0$. Thus, the series for $C(x)$ should look like

$$C(x) = \frac{1 \pm [1 + \text{higher order terms in } x]}{2x}$$

If we use $+$, we get a constant term of 2 in the numerator, and hence a term of the form $\frac{2}{2x} = \frac{1}{x}$ in the series expansion, which does not give an appropriate power series since a power series should only use non-negative exponents on x. But with the $-$ choice we get

$$\frac{1 - [1 + \text{higher order terms}]}{2x} = \frac{\text{terms with } x^k \text{ where } k \geq 1}{2x} = \text{terms with } x^k \text{ where } k \geq 0,$$

which is a valid power series. Thus the ordinary generating function for the Catalan numbers is

$$C(x) = \frac{1 - \sqrt{1-4x}}{2x}$$

It is possible to figure out how to write this as a power series explicitly, as shown in the book. The key points come in expanding $(1-4x)^{1/2}$ as a series, which can be done using a *binomial series*, which is an analog of the binomial formula

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}, \ n \text{ a postitive integer}$$

for powers n which can be fractional. We skipped over binomial series in our course, but the book talks about them back in Chapter 4. The upshot is that using a binomial series expansion, we can derive the fact that

$$C(x) = \frac{1-\sqrt{1-4x}}{2x} = \sum_{n=0}^{\infty} \frac{1}{n+1}\binom{2n}{n} x^n,$$

so that $C_n = \frac{1}{n+1}\binom{2n}{n}$ is the n-th Catalan number, which is a formula we found alternatively by counting paths back in Lecture 6.

Picking captains revisited. Let us return to a type of problem we considered previously, that of picking captains from various teams. But now, we place no restriction on the number of teams we'll have, so to be clear here is our problem: take n people arranged in a line, split them up into some number of teams while maintaining their order, and then pick a captain from each team. We want to understand the number of ways in which this can be done.

For instance, if we are going to have only one team, the ordinary generating function should be

$$A(x) = \sum_{n=1}^{\infty} nx^n = \frac{x}{(1-x)^2}$$

since for a group of n people being put into 1 team, there are n ways of picking the captain. But, we could also have two teams, in which case the number of ways of splitting into two teams and picking a captain from each has generating function

$$A(x) \cdot A(x) = A(x)^2,$$

since we "do thing A" (i.e. pick a captain) on the first team and then the same "thing A" on the second; the fact that we can have teams of varying size is encoded in the product formula for ordinary generating functions.

But, we don't stop here! We could have 3 teams, for which the generating function is $A(x)^3$, or 4 teams for which the generating function if $A(x)^4$, and so on. In general, the generating function for picking a captain from each of k teams is $A(x)^k$ where $A(x)$ is the function we derived above when we had only one team to consider. Thus, if we consider all possible numbers of teams (when k varies), we get the generating function:

$$1 + \underbrace{A(x)}_{\text{1 team}} + \underbrace{A(x)^2}_{\text{2 teams}} + \underbrace{A(x)^3}_{\text{3 teams}} + \cdots = \sum_{n=0}^{\infty} A(x)^n.$$

(We are using the convention that there is 1 way of doing what we want when we have $n = 0$ people, which is: "do notnhing".) But, we can express this function more concretely, and the result is called the *compositional formula*.

Compositional formula. Suppose $A(x)$ is the ordinary generating function for the number of ways of doing "thing A" on one group of objects. Then, the number of ways to line up n objects, split them into some number of groups while maintaining order, and do "thing A" on each group individually has ordinary generating function

$$\sum_{n=0}^{\infty} A(x)^n = \frac{1}{1-A(x)}$$

which comes from substituting $A(x)$ in for x in the standard geometric series $\sum_{n=0}^{\infty} x^n = \frac{1}{1-x}$. The point is that we need only work out the generating function for a single group, since the possibility of having more than group is encoded within the compositional formula itself.

Back to picking captains. Going back to our captain-picking problem, we had

$$A(x) = \frac{x}{(1-x)^2}$$

as the ordinary generating function for picking a captain from a single team. Thus, by the compositional formula we get

$$\frac{1}{1-A(x)} = \frac{1}{1-\frac{x}{(1-x)^2}} = \frac{(1-x)^2}{(1-x)^2 - x} = \frac{x^2-2x+1}{x^2-3x+1}$$

as the generating function for picking captains from any numbers of teams formed by splitting people lined up into any number of parts, while maintaining order. It is possible to figure out how to express this function as a power series explicitly in order to find precisely the number of ways of doing what we want. We won't go through that here since the algebra gets messy and is not so important for our purposes, but you can see it worked out in the book. It turns out that:

$$\frac{x^2 - 2x + 1}{x^2 - 3x + 1} = \sum_{n=0}^{\infty} \frac{1}{\sqrt{5}} (\alpha_+^n - \alpha_-^n) x^n$$

where $\alpha_\pm = \frac{1}{2}(3 \pm \sqrt{5})$, so $\frac{1}{\sqrt{5}}(\alpha_+^n - \alpha_-^n)$ is the number of ways of forming teams from n people lined up (order maintained) and picking a captain from each.

Revisiting compositions. We now use the compositional formula to give an alternate derivation of the number of compositions of n. Of course, we know from weeks ago that the answer is 2^{n-1}, but the point here is to see how to approach this via generating functions.

A composition of n can consist of more than one part, but the point is that we essentially only need to consider those compositions which consist of a single part, since the fact that we can have multiple parts is handled by the compositional formula itself. (The compositional formula applies since "order" is maintained in the process of constructing compositions: take n "stars" lined up and then insert "bars" in some of the gaps to indicate the parts we want, without modifying the order of the stars). So, let $A(x)$ be the ordinary generating function for the number of compositions of n which consist of a single part:

$$A(x) = \sum_{n=1}^{\infty} (\text{\# compositions of } n \text{ with single part}) x^n.$$

But this coefficient is always 1 for $n \geq 1$, since for any such n there is only 1 composition of n into a single part. Thus

$$A(x) = \sum_{n=1}^{\infty} x^n = \frac{x}{1-x}.$$

Hence the ordinary generating function for the number of all compositions of n is:

$$\frac{1}{1 - A(x)} = \frac{1}{1 - \frac{x}{1-x}} = \frac{1-x}{1-2x} = \frac{1}{1-2x} - \frac{x}{1-2x}$$

Notice how great this is: we don't have to consider having multiple parts ourselves since that is handled by the compositional approach, and the case of having only one part is easy to work out by hand!

So, if we work out the series expression for this generating function, we can read off the number of compositions of n as the coefficient of x^n. We have:

$$\frac{1}{1-2x} - \frac{x}{1-2x} = \sum_{n=0}^{\infty} 2^n x^n - \sum_{n=0}^{\infty} 2^n x^{n+1} = \sum_{n=0}^{\infty} 2^n x^n - \sum_{n=1}^{\infty} 2^{n-1} x^n = 1 + \sum_{n=1}^{\infty} (2^n - 2^{n-1}) x^n$$

so for $n \geq 1$ the number of compositions of n is indeed $2^n - 2^{n-1} = 2^{n-1}$ as expected. Huzzah!