Notes on generating functions

James Aspnes

December 13, 2010

1 Basics

The short version: A generating function represents objects of weight n with z^n, and adds all the objects you have up to get a sum $a_0z^0 + a_1z^1 + a_2z^2 + \dots$, where each a_n counts the number of different objects of weight n. If you are very lucky (or constructed your set of objects by combining simpler sets of objects in certain straightforward ways) there will be some compact expression that is expands to this horrible sum but is easier to right down. Such compact expressions are called **generating functions**, and manipulating them algebraically gives an alternative to actually knowing HowToCount.

1.1 A simple example

We are given some initial prefixes for words: `qu`, `s`, and `t`; some vowels to put in the middle: `a`, `i`, and `oi`; and some suffixes: `d`, `ff`, and `ck`, and we want to calculate the number of words we can build of each length.

One way is to generate all 27 words[1] and sort them by length:

```
sad sid tad tid
quad quid sack saff sick siff soid tack taff tick tiff toid
quack quaff quick quiff quoid soick soiff toick toiff
quoick quoiff
```

This gives us 4 length-3 words, 12 length-4 words, 9 length-5 words, and 2 length-6 words. This is probably best done using a computer, and becomes expensive if we start looking at much larger lists.

An alternative is to solve the problem by judicious use of algebra. Pretend that each of our letters is actually a variable, and that when we concatenate `qu`, `oi`, and `ck` to make `quoick`, we are really multiplying the variables using our usual notation. Then we can express all 27 words as the product (`qu`+`s`+`t`)(`a`+`i`+`oi`)(`d`+`ff`+`ck`). But we don't care about the exact set of words, we just want to know how many we get of each length.

[1] We are using *word* in the combinatorial sense of a finite sequence of letters (possibly even the empty sequence) and not the usual sense of a finite, nonempty sequence of letters that actually make sense.

So now we do the magic trick: we replace every variable we've got with a single variable z. For example, this turns `quoick` into $zzzzzz = z^6$, so we can still find the length of a word by reading off the exponent on z. But we can also do this before we multiply everything out, getting

$$\begin{aligned}
(zz + z + z)(z + z + zz)(z + zz + zz) &= (2z + z^2)(2z + z^2)(z + 2z^2) \\
&= z^3(2 + z)^2(1 + 2z) \\
&= z^3(4 + 4z + z^2)(1 + 2z) \\
&= z^3(4 + 12z + 9z^2 + 2z^3) \\
&= 4z^3 + 12z^4 + 9z^5 + 2z^6.
\end{aligned}$$

We can now read off the number of words of each length directly off the coefficients of this polynomial.

1.2 Why this works

In general, what we do is replace any object of **weight** 1 with z. If we have an object with weight n, we think of it as n weight-1 objects stuck together, i.e., z^n. Disjoint unions are done using addition as in simple counting: $z + z^2$ represents the choice between a weight-1 object and a weight-2 object (which might have been built out of 2 weight-1 objects), while $12z^4$ represents a choice between 12 different weight-4 objects. The trick is that when we multiply two expressions like this, whenever two values z^k and z^l collide, the exponents add to give a new value z^{k+l} representing a new object with total weight $k+l$, and if we have something more complex like $(nz^k)(mz^l)$, then the coefficients multiply to give $(nm)z^{k+l}$ different weight $(k + l)$ objects.

For example, suppose we want to count the number of robots we can build given 5 choices of heads, each of weight 2, and 6 choices of bodies, each of weight 5. We represent the heads by $5z^2$ and the bodies by $6z^5$. When we multiply these expressions together, the coefficients multiply (which we want, by the product rule) and the exponents add: we get $5z^2 \cdot 6z^5 = 30z^7$ or 30 robots of weight 7 each.

The real power comes in when we consider objects of different weights. If we add to our 5 weight-2 robot heads two extra-fancy heads of weight 3, and compensate on the body side with three new lightweight weight-4 bodies, our new expression is $(5z^2 + 2z^3)(3z^4 + 6z^5) = 15z^6 + 36z^7 + 12z^8$, giving a possible 15 weight-6 robots, 36 weight-7 robots, and 12 weight-8 robots. The rules for multiplying polynomials automatically tally up all the different cases for us.

This trick even works for infinitely-long polynomials that represent infinite series (such "polynomials" are called formal power series). Even though there might be infinitely many ways to pick three natural numbers, there are only finitely many ways to pick three natural numbers whose sum is 37. By computing an appropriate formal power series and extracting the coefficient from the z^{37} term, we can figure out exactly how many ways there are. This works best, of course, when we don't have to haul around an entire infinite series, but can instead represent it by some more compact function whose expansion gives the desired series. Such a function is called a **generating function**, and manipulating generating functions can be a powerful alternative to creativity in making combinatorial arguments.

1.3 Formal definition

Given a sequence $a_0, a_1, a_2, \ldots$, its **generating function** $F(z)$ is given by the sum

$$F(z) = \sum_{i=0}^{\infty} a_i z^i.$$

A sum in this form is called a **formal power series**. It is "formal" in the sense that we don't necessarily plan to actually compute the sum, and are instead using the string of z^i terms as a long rack to store coefficients on.

In some cases, the sum has a more compact representation. For example, we have

$$\frac{1}{1-z} = \sum_{i=0}^{\infty} z^i,$$

so $1/(1-z)$ is the generating function for the sequence $1, 1, 1, \ldots$. This may let us manipulate this sequence conveniently by manipulating the generating function.

Here's a simple case. If $F(z)$ generates some sequence a_i, what does sequence b_i does $F(2z)$ generate? The i-th term in the expansion of $F(2z)$ will be $a_i(2z)^i = a_i 2^i z^i$, so we have $b_i = 2^i a_i$. This means that the sequence $1, 2, 4, 8, 16, \ldots$ has generating function $1/(1-2z)$. In general, if $F(z)$ represents a_i, then $F(cz)$ represents $c^i a_i$.

What else can we do to F? One useful operation is to take its derivative with respect to z. We then have

$$\frac{d}{dz}F(z) = \sum_{i=0}^{\infty} a_i \frac{d}{dz} z^i = \sum_{i=0}^{\infty} a_i i z^{i-1}.$$

This *almost* gets us the representation for the series ia_i, but the exponents on the z's are off by one. But that's easily fixed:

$$z\frac{d}{dz}F(z) = z\sum_{i=0}^{\infty} a_i i z^{i-1} = \sum_{i=0}^{\infty} a_i i z^i.$$

So the sequence $0, 1, 2, 3, 4, \ldots$ has generating function

$$z\frac{d}{dz}\frac{1}{1-z} = \frac{z}{(1-z)^2},$$

and the sequence of squares $0, 1, 4, 9, 16, \ldots$ has generating function

$$z\frac{d}{dz}\frac{z}{(1-z)^2} = \frac{z}{(1-z)^2} + \frac{2z^2}{(1-z)^3}.$$

As you can see, some generating functions are prettier than others.

(We can also use integration to divide each term by i, but the details are messier.)

Another way to get the sequence $0, 1, 2, 3, 4, \ldots$ is to observe that it satisfies the recurrence:

- $a_0 = 0$.
- $a_{n+1} = a_n + 1 (\forall n \in \mathbb{N})$.

A standard trick in this case is to multiply each of the $\forall i$ bits by z^n, sum over all n, and see what happens. This gives $\sum a_{n+1}z^n = \sum a_n z^n + \sum z^n = \sum a_n z^n + 1/(1-z)$. The first term on the right-hand side is the generating function for a_n, which we can call $F(z)$ so we don't have to keep writing it out. The second term is just the generating function for $1, 1, 1, 1, 1, \ldots$. But what about the left-hand side? This is almost the same as $F(z)$, except the coefficients don't match up with the exponents. We can fix this by dividing $F(z)$ by z, after carefully subtracting off the a_0 term:

$$\begin{aligned}
(F(z) - a_0)/z &= \left(\sum_{n=0}^{\infty} a_n z^n - a_0\right)/z \\
&= \left(\sum_{n=1}^{\infty} a_n z^n\right)/z \\
&= \sum_{n=1}^{\infty} a_n z^{n-1} \\
&= \sum_{n=0}^{\infty} a_{n+1} z^n.
\end{aligned}$$

So this gives the equation $(F(z) - a_0)/z = F(z) + 1/(1-z)$. Since $a_0 = 0$, we can rewrite this as $F(z)/z = F(z) + 1/(1-z)$. A little bit of algebra turns this into $F(z) - zF(z) = z/(1-z) or F(z) = z/(1-z)^2$.

Yet another way to get this sequence is construct a collection of objects with a simple structure such that there are exactly n objects with weight n. One way to do this is to consider strings of the form a^+b^* where we have at least one a followed by zero or more b's. This gives n strings of length n, because we get one string for each of the 1 through n a's we can put in (an example would be abb, aab, and aaa for $n = 3$). We can compute the generating function for this set because to generate each string we must pick in order:

- One initial a. Generating function $= z$.
- Zero or more a's. Generating function $= 1/(1-z)$.
- Zero or more b's. Generating function $= 1/(1-z)$.

Taking the product of these gives $z/(1-z)^2$, as before.

This trick is useful in general; if you are given a generating function $F(z)$ for a_n, but want a generating function for $b_n = \sum_{k \le n} a_k$, allow yourself to pad each weight-k object out to weight n in exactly one way using $n-k$ junk objects, i.e. multiply $F(z)$ by $1/(1-z)$.

2 Some standard generating functions

Here is a table of some of the most useful generating functions.

$$\begin{aligned}
\frac{1}{1-z} &= \sum_{i=0}^{\infty} z^i \\
\frac{z}{(1-z)^2} &= \sum_{i=0}^{\infty} i z^i \\
(1+z)^n &= \sum_{i=0}^{\infty} \binom{n}{i} z^i = \sum_{i=0}^{n} \binom{n}{i} z^i \\
\frac{1}{(1-z)^n} &= \sum_{i=0}^{\infty} \binom{n+i-1}{i} z^i
\end{aligned}$$

Of these, the first is the most useful to remember (it's also handy for remembering how to sum geometric series). All of these equations can be proven using the binomial theorem.

3 More operations on formal power series and generating functions

Let $F(z) = \sum_i a_i z^i$ and $G(z) = \sum_i b_i z^i$. Then their sum $F(z) + G(z) = \sum_i (a_i + b_i) z^i$ is the generating function for the sequence $(a_i + b_i)$. What is their product $F(z)G(z)$?

To compute the i-th term of $F(z)G(z)$, we have to sum over all pairs of terms, one from F and one from G, that produce a z^i factor. Such pairs of terms are precisely those that have exponents that sum to i. So we have

$$F(z)G(z) = \sum_{i=0}^{\infty} \left(\sum_{j=0}^{i} a_j b_{j-i} \right) z^i.$$

As we've seen, this equation has a natural combinatorial interpretation. If we interpret the coefficient a_i on the i-th term of $F(z)$ as counting the number of "a-things" of weight i, and the coefficient b_i as the number of "b-things" of weight i, then the i-th coefficient of $F(z)G(z)$ counts the number of ways to make a combined thing of total weight i by gluing together an a-thing and a b-thing.

As a special case, if $F(z) = G(z)$, then the i-th coefficient of $F(z)G(z) = F^2(z)$ counts how many ways to make a thing of total weight i using two "a-things", and $F^n(z)$ counts how many ways (for each i) to make a thing of total weight i using n "a-things". This gives us an easy combinatorial proof of a special case of the binomial theorem:

$$(1+x)^n = \sum_{i=0}^{\infty} \binom{n}{i} x^i.$$

Think of the left-hand side as the generating function $F(x) = 1 + x$ raised to the n-th power. The function F by itself says that you have a choice between one weight-0 object or one weight-1 object. On the right-hand side the i-th coefficient counts how many ways you can put together a total of i weight-1 objects given n to choose from—so it's $\binom{n}{i}$.

4 Counting with generating functions

The product formula above suggests that generating functions can be used to count combinatorial objects that are built up out of other objects, where our goal is to count the number of objects of each possible non-negative integer "weight" (we put "weight" in scare quotes because we can make the "weight" be any property of the object we like, as long as it's a non-negative integer—a typical choice might be the size of a set, as in the binomial theorem example above). There are five basic operations involved in this process; we've seen two of them already, but will restate them here with the others.

Throughout this section, we assume that $F(z)$ is the generating function counting objects in some set A and $G(z)$ the generating function counting objects in some set B.

4.1 Disjoint union

Suppose $C = A \cup B$ and A and B are disjoint. Then the generating function for objects in C is $F(z) + G(z)$.

Example: Suppose that A is the set of all strings of zero or more letters x, where the weight of a string is just its length. Then $F(z) = 1/(1-z)$, since there is exactly one string of each length and the coefficient a_i on each z^i is always 1. Suppose that B is the set of all strings of zero or more letters y and/or z, so that $G(z) = 1/(1-2z)$ (since there are now 2^i choices of length-i strings). The set C of strings that are either (a) all x's or (b) made up of y's, z's, or both, has generating function $F(z) + G(z) = 1/(1-z) + 1/(1-2z)$.

4.2 Cartesian product

Now let $C = A \times B$, and let the weight of a pair $(a, b) \in C$ be the sum of the weights of a and b. Then the generating function for objects in C is $F(z)G(z)$.

Example: Let A be all-x strings and B be all-y or all-z strings, as in the previous example. Let C be the set of all strings that consist of zero or more x's followed by zero or more y's and/or z's. Then the generating function for C is $F(z)G(z) = \frac{1}{(1-z)(1-2z)}$.

4.3 Repetition

Now let C consists of all finite sequences of objects in A, with the weight of each sequence equal to the sum of the weights of its elements (0 for an empty sequence). Let $H(z)$ be the generating function for C. From the preceding rules we have

$$H = 1 + F + F^2 + F^3 + \ldots = \frac{1}{1-F}.$$

This works best when $H(0) = 0$; otherwise we get infinitely many weight-0 sequences. It's also worth noting that this is just a special case of substitution (see below), where our "outer" generating function is $1/(1-z)$.

4.3.1 Example: (0|11)∗

Let $A = \{0, 11\}$, and let C be the set of all sequences of zeros and ones where ones occur only in even-length runs. Then the generating function for A is $z + z^2$ and the generating function for C is $1/(1 - z - z^2)$. We can extract exact coefficients from this generating function using the techniques below.

4.3.2 Example: sequences of positive integers

Suppose we want to know how many different ways there are to generate a particular integer as a sum of positive integers. For example, we can express 4 as 4, $3+1$, $2+2$, $2+1+1$, $1+1+1+1$, $1+1+2$, $1+2+1$, or $1+3$, giving 8 different ways.

We can solve this problem using the repetition rule. Let $F = z/(1-z)$ generate all the positive integers. Then

$$\begin{aligned} H &= \frac{1}{1-F} \\ &= \frac{1}{1-\frac{z}{1-z}} \\ &= \frac{1-z}{(1-z)-z} \\ &= \frac{1-z}{1-2z}. \end{aligned}$$

We can get exact coefficients by observing that

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

This means that there is 1 way to express 0 (the empty sum), and 2^{n-1} ways to express any larger value n (e.g. $2^{4-1} = 8$ ways to express 4).

Once we know what the right answer is, it's not terribly hard to come up with a combinatorial explanation. The quantity 2^{n-1} counts the number of subsets of an $(n-1)$-element set. So imagine that we have $n-1$ places and we mark some subset of them, plus add an extra mark at the end; this might give us a pattern like `XX-X`. Now for each sequence of places ending with a mark we replace it with the number of places (e.g. `XX-X` $= 1, 1, 2$, `X--X-X---X` $= 1, 3, 2, 4$). Then the sum of the numbers we get is equal to n, because it's just counting the total length of the sequence by dividing it up at the marks and the adding the pieces back together. The value 0 doesn't fit this pattern (we can't put in the extra mark without getting a sequence of length 1), so we have 0 as a special case again.

If we are very clever, we might come up with this combinatorial explanation from the beginning. But the generating function approach saves us from having to be clever.

4.4 Pointing

This operation is a little tricky to describe. Suppose that we can think of each weight-k object in A as consisting of k items, and that we want to count not only how many weight-k objects there are, but how many ways we can produce a weight-k object where one of its k items has a special mark on it. Since there are k different items to choose for each weight-k object, we are effectively multiplying the count of weight-k objects by k. In generating function terms, we have

$$H(z) = z \frac{d}{dz} F(z).$$

Repeating this operation allows us to mark more items (with some items possibly getting more than one mark). If we want to mark n distinct items in each object (with distinguishable marks), we can compute

$$H(z) = z^n \frac{d^n}{dz^n} F(z),$$

where the repeated derivative turns each term $a_i z^i$ into $a_i i(i-1)(i-2)\dots(i-n+1)z^{i-n}$ and the z^n factor fixes up the exponents. To make the marks indistinguishable (i.e., we don't care what order the values are marked in), divide by $n!$ to turn the extra factor into $\binom{i}{n}$.

(If you are not sure how to take a derivative, look at HowToDifferentiate.)

Example: Count the number of finite sequences of zeros and ones where exactly two digits are underlined. The generating function for {0,1} is $2z$, so the generating function for sequences of zeros and ones is $F = 1/(1-2z)$ by the repetition rule. To mark two digits with indistinguishable marks, we need to compute

$$\frac{1}{2}z^2 \frac{d^2}{dz^2} \frac{1}{1-2z} = \frac{1}{2}z^2 \frac{d}{dz} \frac{2}{(1-2z)^2} = \frac{1}{2}z^2 \frac{8}{(1-2z)^3} = \frac{4z^2}{(1-2z)^3}.$$

4.5 Substitution

Suppose that the way to make a C-thing is to take a weight-k A-thing and attach to each its k items a B-thing, where the weight of the new C-thing is the sum of the weights of the B-things. Then the generating function for C is the composition $F(G(z))$.

Why this works: Suppose we just want to compute the number of C-things of each weight that are made from some single specific weight-k A-thing. Then the generating function for this quantity is just $(G(z))^k$. If we expand our horizons to include all a_k weight-k A-things, we have to multiply by a_k to get $a_k(G(z))^k$. If we further expand our horizons to include A-things of all different weights, we have to sum over all k:

$$\sum_{k=0}^{\infty} a_k (G(z))^k.$$

But this is just what we get if we start with $F(z)$ and substitute $G(z)$ for each occurrence of z, i.e. if we compute $F(G(z))$.

4.5.1 Example: bit-strings with primes

Suppose we let A be all sequences of zeros and ones, with generating function $F(z) = 1/(1-2z)$. Now suppose we can attach a single or double prime to each 0 or 1, giving $0'$ or $0''$ or $1'$ or $1''$, and we want a generating function for the number of distinct primed bit-strings with n attached primes. The set $\{', ''\}$ has generating function $G(z) = z + z^2$, so the composite set has generating function $F(z) = 1/(1-2(z+z^2)) = 1/(1-2z-2z^2)$.

4.5.2 Example: (0—11)* again

The previous example is a bit contrived. Here's one that's a little more practical, although it involves a brief digression into **multivariate generating functions**. A multivariate generating function $F(x, y)$ generates a series $\sum_{ij} a_{ij} x^i y^j$, where a_{ij} counts the number of things that have i x's and j y's. (There is also the obvious generalization to more than two variables). Consider the multivariate generating function for the set {0,1}, where x counts zeros and y counts ones: this is just $x + y$. The multivariate generating function for sequences of zeros and ones is $1/(1-x-y)$ by the repetition rule. Now suppose that each 0 is left intact but each 1 is replaced by 11, and we want to count the total number of strings by length, using z as our series variable. So we substitute z for x and z^2 for y (since each y turns into a string of length 2), giving $1/(1-z-z^2)$. This gives another way to get the generating function for strings built by repeating 0 and 11.

5 Generating functions and recurrences

What makes generating functions particularly useful for algorithm analysis is that they directly solve recurrences of the form $T(n) = aT(n-1) + bT(n-2) + f(n)$ (or similar recurrences with more T terms on the right-hand side), provided we have a generating function $F(z)$ for $f(n)$. The idea is that there exists some generating function $G(z)$ that describes the entire sequence of values $T(0), T(1), T(2), \dots$, and we just need to solve for it by restating the recurrence as an equation about G. The left-hand side will just turn into G. For the right-hand side, we need to shift $T(n-1)$ and $T(n-2)$ to line up right, so that the right-hand side will correctly represent the sequence $T(0), T(1), aT(0) + aT(1) + F(2)$, etc. It's not hard to see that the generating function for the sequence $0, T(0), T(1), T(2), \dots$ (corresponding to the $T(n-1)$ term) is just $zG(z)$, and similarly the sequence $0, 0, T(1), T(2), T(3), \dots$ (corresponding to the $T(n-2)$ term) is $z^2G(z)$. So we have (being very careful to subtract out extraneous terms at for $i = 0$ and $i = 1$):

$$G = az(G - T(0)) + bz^2G + (F - f(0) - zf(1)) + T(0) + zT(1),$$

and after expanding F we can in principle solve this for G as a function of z.

5.1 Example: A Fibonacci-like recurrence

Let's take a concrete example. The Fibonacci-like recurrence

$$T(n) = T(n-1) + T(n-2), T(0) = 1, T(1) = 1,$$

becomes

$$G = (zG - z) + z^2G + 1 + z.$$

(here $F = 0$).

Solving for G gives

$$G = 1/(1 - z - z^2).$$

Unfortunately this is not something we recognize from our table, although it has shown up in a couple of examples. (Exercise: *Why* does the recurrence $T(n) = T(n-1) + T(n-2)$ count the number of strings built from 0 and 11 of length n?) In the next section we show how to recover a closed-form expression for the coefficients of the resulting series.

6 Recovering coefficients from generating functions

There are basically three ways to recover coefficients from generating functions:

1. Recognize the generating function from a table of known generating functions, or as a simple combination of such known generating functions. This doesn't work very often but it is possible to get lucky.

2. To find the k-th coefficient of $F(z)$, compute the k-th derivative $d^k/dz^k F(z)$ and divide by $k!$ to shift a_k to the z^0 term. Then substitute 0 for z. For example, if $F(z) = 1/(1-z)$ then $a_0 = 1$ (no differentiating), $a_1 = 1/(1-0)^2 = 1$, $a_2 = 1/(1-0)^3 = 1$, etc. This usually only works if the derivatives have a particularly nice form or if you only care about the first couple of coefficients (it's particularly effective if you only want a_0).

3. If the generating function is of the form $1/Q(z)$, where Q is a polynomial with $Q(0) \neq 0$, then it is generally possible to expand the generating function out as a sum of terms of the form $P_c/(1 - z/c)$ where c is a root of Q (i.e. a value such that $Q(c) = 0$). Each denominator P_c will be a constant if c is not a repeated root; if c is a repeated root, then P_c can be a polynomial of degree up to one less than the multiplicity of c. We like these expanded solutions because we recognize $1/(1 - z/c) = \sum_i c^{-i} z^i$, and so we can read off the coefficients a_i generated by $1/Q(z)$ as an appropriately weighted some of c_1^{-i}, c_2^{-i}, etc., where the c_j range over the roots of Q.

Example: Take the generating function $G = 1/(1 - z - z^2)$. We can simplify it by factoring the denominator: $1 - z - z^2 = (1 - az)(1 - bz)$ where $1/a$ and $1/b$ are the solutions to the equation $1 - z - z^2 = 0$; in this case $a = (1 + \sqrt{5})/2$, which is approximately 1.618 and $b = (1 - \sqrt{5})/2$, which is approximately -0.618. It happens to be the case that we can always expand $1/P(z)$ as $A/(1 - az) + B(1 - bz)$ for some constants A and B whenever P is a degree 2 polynomial with constant coefficient 1 and distinct roots a and b, so

$$G = A/(1 - az) + B/(1 - bz),$$

and here we can recognize the right-hand side as the sum of the generating functions for the sequences $A \cdot a^i$ and $B \cdot b^i$. The $A \cdot a^i$ term dominates, so we have that $T(n) = \Theta(a^n)$, where a is approximately 1.618. We can also solve for A and B exactly to find an exact solution if desired.

A rule of thumb that applies to recurrences of the form $T(n) = a_1T(n-1) + a_2T(n-2) + \ldots a_kT(n-k) + f(n)$ is that unless f is particularly large, the solution is usually exponential in $1/x$, where x is the smallest root of the polynomial $1 - a_1z$-$a_2z^2 \ldots$-a_kz^k. This can be used to get very quick estimates of the solutions to such recurrences (which can then be proved without fooling around with generating functions).

Exercise: What is the exact solution if $T(n) = T(n-1) + T(n-2) + 1$? Or if $T(n) = T(n-1) + T(n-2) + n$?

6.1 Partial fraction expansion and Heaviside's cover-up method

There is a nice trick for finding the numerators in a partial fraction expansion. Suppose we have

$$\frac{1}{(1-az)(1-bz)} = \frac{A}{1-az} + \frac{B}{1-bz}.$$

Multiply both sides by $1 - az$ to get

$$\frac{1}{1-bz} = A + \frac{B(1-az)}{1-bz}.$$

Now plug in $z = 1/a$ to get

$$\frac{1}{1-b/a} = A + 0.$$

We can immediately read off A. Similarly, multiplying by $1 - bz$ and then setting $1 - bz$ to zero gets B. The method is known as the "cover-up method" because multiplication by $1 - az$ can be simulated by covering up $1 - az$ in the denominator of the left-hand side and all the terms that don't have $1 - az$ in the denominator in the right hand side.

The cover-up method will work in general whenever there are no repeated roots, even if there are many of them; the idea is that setting $1 - qz$ to zero knocks out all the terms on the right-hand side but one. With repeated roots we have to worry about getting numerators that aren't just a constant, so things get more complicated. We'll come back to this case below.

6.1.1 Example: A simple recurrence

Suppose $f(0) = 0, f(1) = 1$, and for $n \geq 2$, $f(n) = f(n-1) + 2f(n-2)$. Multiplying these equations by z^n and summing over all n gives a generating function

$$F(z) = \sum_{n=0} \infty f(n)z^n = 0 \cdot z^0 + 1 \cdot z^1 + \sum_{n=2}^{\infty} f(n-1)z^n + \sum_{n=2}^{\infty} 2f(n-2)z^n.$$

With a bit of tweaking, we can get rid of the sums on the RHS by converting them into copies of F:

$$\begin{aligned}
F(z) &= z + \sum_{n=2}^{\infty} f(n-1)z^n + 2\sum_{n=2}^{\infty} f(n-2)z^n \\
&= z + \sum_{n=1}^{\infty} f(n)z^{n+1} + 2\sum_{n=0}^{\infty} f(n)z^{n+2} \\
&= z + z\sum_{n=1}^{\infty} f(n)z^n + 2z^2\sum_{n=0}^{\infty} f(n)z^n \\
&= z + z(F(z) - f(0)z^0) + 2z^2F(z) \\
&= z + zF(z) + 2z^2F(z).
\end{aligned}$$

Now solve for $F(z)$ to get $F(x) = \frac{\{}{z}\}\{1-z-2z^2\} = \frac{\{}{z}\}\{(1+z)(1-2z)\} = z\left(\frac{\{}{A}\}\{1+z\} + \frac{\{}{B}\}\{1-2z\}\right)$, where we need to solve for A and B.

We can do this directly, or we can use the cover-up method. The cover-up method is easier. Setting $z = -1$ and covering up $1+z$ gives $A = 1/(1-2(-1)) = 1/3$. Setting $z = 1/2$ and covering up $1-2z$ gives $B = 1/(1+z) = 1/(1+1/2) = 2/3$. So we have

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

This gives $f(0) = 0$ and, for $n \geq 1$, $f(n) = \frac{\{}{2}^{n-(-1)}n\}\{3\}$. It's not hard to check that this gives the same answer as the recurrence.

6.1.2 Example: Coughing cows

Let's count the number of strings of each length of the form `(M)*(O|U)*(G|H|K)*` where `(x|y)` means we can use x or y and * means we can repeat the previous parenthesized expression 0 or more times (see WikiPedia's article on regular expressions).

We start with a sequence of 0 or more M's. The generating function for this part is our old friend $1/(1-z)$. For the second part, we have two choices for each letter, giving $1/(1-2z)$. For the third part, we have $1/(1-3z)$. Since each part can be chosen independently of the other two, the generating function for all three parts together is just the product:

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

Let's use the cover-up method to convert this to a sum of partial fractions. We have

$$\begin{aligned}
\frac{1}{(1-z)(1-2z)(1-3z)} &= \frac{\left(\frac{1}{(1-2)(1-3)}\right)}{1-z} + \frac{\left(\frac{1}{(1-\frac{1}{2})(1-\frac{3}{2})}\right)}{1-2z} + \frac{\left(\frac{1}{(1-\frac{1}{3})(1-\frac{2}{3})}\right)}{1-3z} \\
&= \frac{\frac{1}{2}}{1-z} + \frac{-4}{1-2z} + \frac{\frac{9}{2}}{1-3z}.
\end{aligned}$$

So the exact number of length-n sequences is $(1/2) - 4 \cdot 2^n + (9/2) \cdot 3^n$. We can check this for small n:

n	Formula	Strings
0	$1/2 - 4 + 9/2 = 1$	()
1	$1/2 - 8 + 27/2 = 6$	M, O, U, G, H, K
2	$1/2 - 16 + 81/2 = 25$	$MM, MO, MU, MG, MH, MK, OO, OU, OG, OH, OK, UO, UU, UG, UH, UK, G$
3	$1/2 - 32 + 243/2 = 90$	(exercise)$\ddot\smile$

6.1.3 Example: A messy recurrence

Let's try to solve the recurrence $T(n) = 4T(n-1) + 12T(n-2) + 1$ with $T(0) = 0$ and $T(1) = 1$.

Let $F = \sum T(n) z^n$.

Summing over all n gives

$$\begin{aligned}
F = \sum_{n=0}^{\infty} T(n) z^n &= T(0)z^0 + T(1)z^1 + 4\sum_{n=2}^{\infty} T(n-1)z^n + 12\sum_{n=2}^{\infty} T(n-2)z^n + \sum_{n=2}^{\infty} 1 \cdot z^n \\
&= z + 4z\sum_{n=1}^{\infty} T(n)z^n + 12z^2\sum_{n=0}^{\infty} T(n)z^n + z^2\sum_{n=0}^{\infty} z^n \\
&= z + 4z(F - T(0)) + 12z^2 F + \frac{z^2}{1-z} \\
&= z + 4zF + 12z^2 F + \frac{z^2}{1-z}.
\end{aligned}$$

Solving for F then gives

$$F = \frac{\left(z + \frac{z^2}{1-z}\right)}{1 - 4z - 12z^2}.$$

We want to solve this using partial fractions, so we need to factor $(1-4z-12z^2) = (1+2z)(1-6z)$. This gives

$$\begin{aligned}
F &= \frac{\left(z+\frac{z^2}{1-z}\right)}{(1+2z)(1-6z)} \\
&= \frac{z}{(1+2z)(1-6z)} + \frac{z^2}{(1-z)(1+2z)(1-6z)}. \\
&= z\left(\frac{1}{(1+2z)(1-6(-\frac{1}{2}))} + \frac{1}{(1+2(\frac{1}{6}))(1-6z)}\right) \\
&\quad +z^2\left(\frac{1}{(1-z)(1+2)(1-6)} + \frac{1}{(1-(-\frac{1}{2}))(1+2z)(1-6(-\frac{1}{2}))} + \frac{1}{(1-\frac{1}{6})(1+2(\frac{1}{6}))(1-6z)}\right) \\
&= \frac{\frac{1}{4}z}{1+2z} + \frac{\frac{3}{4}z}{1-6z} + \frac{-\frac{1}{15}z^2}{1-z} + \frac{\frac{1}{6}z^2}{1+2z} + \frac{\frac{9}{10}z^2}{1-6z}.
\end{aligned}$$

From this we can immediately read off the value of $T(n)$ for $n \geq 2$:

$$\begin{aligned}
T(n) &= \frac{1}{4}(-2)^{n-1} + \frac{3}{4}6^{n-1} - \frac{1}{15} + \frac{1}{6}(-2)^{n-2} + \frac{9}{10}6^{n-2} \\
&= -\frac{1}{8}(-2)^n + \frac{1}{8}6^n - \frac{1}{15} + \frac{1}{24}(-2)^n + \frac{1}{40}6^n \\
&= \frac{3}{20}6^n - \frac{1}{12}(-2)^n - \frac{1}{15}.
\end{aligned}$$

Let's check this against the solutions we get from the recurrence itself:

n	$T(n)$
0	0
1	1
2	$1+4\cdot 1+12\cdot 0 = 5$
3	$1+4\cdot 5+12\cdot 1 = 33$
4	$1+4\cdot 33+12\cdot 5 = 193$

We'll try $n=3$, and get $T(3) = (3/20)\cdot 216 + 8/12 - 1/15 = (3\cdot 3\cdot 216 + 40 - 4)/60 = (1944+40-4)/60 = 1980/60 = 33$.

To be extra safe, let's try $T(2) = (3/20)\cdot 36 - 4/12 - 1/15 = (3\cdot 3\cdot 36 - 20 - 4)/60 = (324-20-4)/60 = 300/60 = 5$. This looks good too.

The moral of this exercise? Generating functions can solve ugly-looking recurrences exactly, but you have to be very very careful in doing the math.

6.2 Partial fraction expansion with repeated roots

Let $a_n = 2a_{n-1} + n$, with some constant a_0. We'd like to find a closed-form formula for a_n.

As a test, let's figure out the first few terms of the sequence:

$$\begin{aligned}
a_0 &= a_0\\
a_1 &= 2a_0 + 1\\
a_2 &= 4a_0{+}2{+}2{=}\ 4a_0 + 4\\
a_3 &= 8a_0{+}8{+}3{=}\ 8a_0 + 11\\
a_4 &= 16a_0 + 22{+}4{=}\ 16a_0 + 26
\end{aligned}$$

The a_0 terms look nice (they're $2^n a_0$), but the 0, 1, 4, 11, 26 sequence doesn't look like anything familiar. So we'll find the formula the hard way.

First we convert the recurrence into an equation over generating functions and solve for the generating function F:

$$\begin{aligned}
\sum a_n z^n &= 2\sum a_{n-1} z^n + \sum n z^n + a_0\\
F &= 2zF + \frac{z}{(1-z)^2} + a_0\\
(1-2z)F &= \frac{z}{(1-z)^2} + a_0\\
F &= \frac{z}{(1-z)^2(1-2z)} + \frac{a_0}{1-2z}.
\end{aligned}$$

Observe that the right-hand term gives us exactly the $2^n a_0$ terms we expected, since $1/(1-2z)$ generates the sequence 2^n. But what about the left-hand term? Here we need to apply a partial-fraction expansion, which is simplified because we already know how to factor the denominator but is complicated because there is a repeated root.

We can now proceed in one of two ways: we can solve directly for the partial fraction expansion, or we can use an extended version of Heaviside's cover-up method that handles repeated roots using differentiation. We'll start with the direct method.

6.2.1 Solving for the PFE directly

Write

$$\frac{1}{(1-z)^2(1-2z)} = \frac{A}{(1-z)^2} + \frac{B}{1-2z}$$

.

We expect B to be a constant and A to be of the form $A_1 z + A_0$.

To find B, use the technique of multiplying by $1-2z$ and setting $z = 1/2$:

$$\frac{1}{(1-\frac{1}{2})^2} = \frac{A \cdot 0}{(1-z)^2} + B.$$

So $B = 1/(1-1/2)^2 = 1/(1/4) = 4$.

We can't do this for A, but we can solve for it after substituting in $B = 4$:

$$\begin{aligned}
\frac{1}{(1-z)^2(1-2z)} &= \frac{A}{(1-z)^2} + \frac{4}{1-2z} \\
1 &= A(1-2z) + 4(1-z)^2 \\
A &= \frac{1-4(1-z)^2}{1-2z} \\
&= \frac{1-4+8z-4z^2}{1-2z} \\
&= \frac{-3+8z-4z^2}{1-2z} \\
&= \frac{-(1-2z)(3-2z)}{1-2z} \\
&= 2z-3.
\end{aligned}$$

So we have the expansion

$$\frac{1}{(1-z)^2(1-2z)} = \frac{2z-3}{(1-z)^2} + \frac{4}{1-2z},$$

from which we get

$$\begin{aligned}
F &= \frac{z}{(1-z)^2(1-2z)} + \frac{a_0}{1-2z} \\
&= \frac{2z^2-3z}{(1-z)^2} + \frac{4z}{1-2z} + \frac{a_0}{1-2z}.
\end{aligned}$$

If we remember that $1/(1-z)^2$ generates the sequence $x_n = n+1$ and $1/(1-2z)$ generates $x_n = 2^n$, then we can quickly read off the solution (for large n):

$$a_n = 2(n-1) - 3n + 4 \cdot 2^{n-1} + a_0 \cdot 2^n = 2^n a_0 + 2^{n+1} - 2 - n$$

which we can check by plugging in particular values of n and comparing it to the values we got by iterating the recurrence before.

The reason for the "large n" caveat is that $z^2/(1-z)^2$ doesn't generate precisely the sequence $x_n = n-1$, since it takes on the values $0, 0, 1, 2, 3, 4, \ldots$ instead of $-1, 0, 1, 2, 3, 4, \ldots$. Similarly, the power series for $z/(1-2z)$ does not have the coefficient $2^{n-1} = 1/2$ when $n = 0$. Miraculously, in this particular example the formula works for $n = 0$, even though it shouldn't: $2(n-1)$ is -2 instead of 0, but $4 \cdot 2^{n-1}$ is 2 instead of 0, and the two errors cancel each other out.

6.2.2 Solving for the PFE using the extended cover-up method

It is also possible to extend the cover-up method to handle repeated roots. Here we choose a slightly different form of the partial fraction expansion:

$$\frac{1}{(1-z)^2(1-2z)} = \frac{A}{(1-z)^2} + \frac{B}{1-z} + \frac{C}{1-2z}.$$

Here A, B, and C are all constants. We can get A and C by the cover-up method, where for A we multiply both sides by $(1-z)^2$ before setting $z = 1$; this gives $A = 1/(1-2) = -1$ and $C = 1/(1-\frac{1}{2})^2 = 4$. For B, if we multiply both sides by $(1-z)$ we are left with $A/(1-z)$ on the right-hand side and a $(1-z)$ in the denominator on the left-hand side. Clearly setting $z = 1$ in this case will not help us.

The solution is to first multiply by $(1-z)^2$ as before but then take a derivative:

$$\begin{aligned}
\frac{1}{(1-z)^2(1-2z)} &= \frac{A}{(1-z)^2} + \frac{B}{1-z} + \frac{C}{1-2z} \\
\frac{1}{1-2z} &= A + B(1-z) + \frac{C(1-z)^2}{1-2z} \\
\frac{d}{dz}\frac{1}{1-2z} &= \frac{d}{dz}\left(A + B(1-z) + \frac{C(1-z)^2}{1-2z}\right) \\
\frac{2}{(1-2z)^2} &= -B + \frac{-2C(1-z)}{1-2z} + \frac{2C(1-z)^2}{(1-2z)^2}
\end{aligned}$$

Now if we set $z = 1$, every term on the right-hand side except $-B$ becomes 0, and we get $-B = 2/(1-2)^2$ or $B = -2$.

Plugging A, B, and C into our original formula then gives

$$\frac{1}{(1-z)^2(1-2z)} = \frac{-1}{(1-z)^2} + \frac{-2}{1-z} + \frac{4}{1-2z},$$

and thus

$$F = \frac{z}{(1-z)^2(1-2z)} + \frac{a_0}{1-2z} = z\left(\frac{-1}{(1-z)^2} + \frac{-2}{1-z} + \frac{4}{1-2z}\right) + \frac{a_0}{1-2z}.$$

From this we can read off (for large n):

$$a_n = 4 \cdot 2^{n-1} - n - 2 + a_0 \cdot 2^n = 2^{n+1} + 2^n a_0 - n - 2.$$

We believe this because it looks like the solution we already got.

7 Asymptotic estimates

We can simplify our life considerably if we only want an asymptotic estimate of a_n (see AsymptoticNotation). The basic idea is that if a_n is non-negative for sufficiently large n and $\sum a_n z^n$ converges for some fixed value z, then a_n must be $o(z^{-n})$ in the limit. (Proof: otherwise, $a_n z^n$ is at least a constant for infinitely many n, giving a divergent sum.) So we can use the **radius of convergence** of a generating function $F(z)$, defined as the largest value r such that $F(z)$ is defined for all (complex) z with $|z| < r$, to get a quick estimate of the growth rate of F's coefficients: whatever they do, we have $a_n = O(r^{-n})$.

For generating functions that are **rational functions** (ratios of polynomials), we can use the partial fraction expansion to do even better. First observe that for $F(z) = \sum f_i z^n = 1/(1-az)^k$, we have $f_n = \binom{k+n-1}{n} a^n = \frac{(n+k-1)(n+k-2)\dots(k-1)}{(k-1)!} a^n = \Theta(a^n n^{k-1})$. Second, observe that the numerator is irrelevant: if $1/(1-az)^k = \Theta(a^n n^{k-1})$ then $bz^m/(1-az)^{k-1} = b\Theta(a^{n-m}(n-m)^{k-1}) = ba^{-m}(1-m/n)^{k-1}\Theta(a^n n^{k-1}) = \Theta(a^n n^{k-1})$, because everything outside the Θ disappears into the constant for sufficiently large n. Finally, observe that in a partial fraction expansion, the term $1/(1-az)^k$ with the largest coefficient a (if there is one) wins in the resulting asymptotic sum: $\Theta(a^n) + \Theta(b^n) = \Theta(a^n)$ if $|a| > |b|$. So we have:

Theorem 1. *Let $F(z) = \sum f_n z^n = P(z)/Q(z)$ where P and Q are polynomials in z. If Q has a root r with multiplicity k, and all other roots s of Q satisfy $|r| < |s|$, then $f_n = \Theta((1/r)^n n^{k-1})$.*

The requirement that r is a unique minimal root of Q is necessary; for example, $F(z) = 2/(1-z^2) = 1/(1-z) + 1/(1+z)$ generates the sequence $0, 2, 0, 2, \dots$, which is *not* $\Theta(1)$ because of all the zeros; here the problem is that $1 - z^2$ has two roots with the same absolute value, so for some values of n it is possible for them to cancel each other out.

A root in the denominator of a rational function F is called a **pole**. So another way to state the theorem is that the asymptotic value of the coefficients of a rational generating function is determined by the smallest pole.

More examples:

$F(z)$	Smallest pole	Asymptotic value
$1/(1-z)$	1	$\Theta(1)$
$1/(1-z)^2$	$1, multiplicity 2$	$\Theta(n)$
$1/(1-z-z^2)$	$(\sqrt{5}-1)/2 = 2/(1+\sqrt{5})$	$\Theta(((1+\sqrt{5})/2)^n)$
$1/((1-z)(1-2z)(1-3z))$	$1/3$	$\Theta(3^n)$
$(z+z^2(1-z))/(1-4z-12z^2)$	$1/6$	$\Theta(6^n)$
$1/((1-z)^2(1-2z))$	$1/2$	$\Theta(2^n)$

In each case it may be instructive to compare the asymptotic values to the exact values obtained earlier on this page.

8 Recovering the sum of all coefficients

Given a generating function for a convergent series $\sum_i a_i z^i$, we can compute the sum of all the a_i by setting z to 1. Unfortunately, for many common generating functions setting $z = 1$ yields $0/0$ (if it yields something else divided by zero then the series diverges). In this case we can recover the correct sum by taking the limit as z goes to 1 using *L'Hôpital's rule*, which says that $\lim_{x\to c} f(x)/g(x) = \lim_{x\to c} f'(x)/g'(x)$ when the latter limit exists and either $f(c) = g(c) = 0$ or $f(c) = g(c) = \infty$.[2]

[2] The justification for doing this is that we know that a finite sequence really has a finite sum, so the "singularity" appearing at $z = 1$ in e.g. $\frac{1-z^{n+1}}{1-z}$ is an artifact of the generating-

8.1 Example

Let's derive the formula for $1 + 2 + \ldots + n$. We'll start with the generating function for the series $\sum_{i=0}^{n} z^i$, which is $(1 - z^n + 1)/(1 - z)$. Applying the $z\frac{d}{dz}$ method gives us

$$\begin{aligned}
\sum_{i=0}^{n} iz^i &= z\frac{d}{dz}\frac{1-z^{n+1}}{1-z} \\
&= z\left(\frac{1}{(1-z)^2} - \frac{(n+1)z^n}{1-z} - \frac{z^{n+1}}{(1-z)^2}\right) \\
&= \frac{z-(n+1)z^{n+1}+nz^{n+2}}{(1-z)^2}.
\end{aligned}$$

Plugging $z = 1$ into this expression gives $(1-(n+1)+n)/(1-1) = 0/0$, which does not make us happy. So we go to the hospital—twice, since one application of L'Hôpital's rule doesn't get rid of our $0/0$ problem:

$$\begin{aligned}
\lim_{z\to 1}\frac{z-(n+1)z^{n+1}+nz^{n+2}}{(1-z)^2} &= \lim_{z\to 1}\frac{1-(n+1)^2z^n+n(n+2)z^{n+1}}{-2(1-z)} \\
&= \lim_{z\to 1}\frac{-n(n+1)^2z^{n-1}+n(n+1)(n+2)z^n}{2} \\
&= \frac{-n(n+1)^2+n(n+1)(n+2)}{2} \\
&= \frac{-n^3-2n^2-n+n^3+3n^2+2n}{2} \\
&= \frac{n^2+n}{2} = \frac{n(n+1)}{2},
\end{aligned}$$

which is our usual formula. Gauss's childhood proof is a lot quicker, but the generating-function proof is something that we could in principle automate most of the work using a computer algebra system, and it doesn't require much creativity or intelligence. So it might be the weapon of choice for nastier problems where no clever proof comes to mind.

More examples of this technique can be found on the BinomialCoefficients page, where the binomial theorem applied to $(1 + x)^n$ (which is really just a generating function for $\sum \binom{n}{i} z^i$) is used to add up various sums of binomial coefficients.

9 A recursive generating function

Let's suppose we want to count binary trees with n internal nodes. We can obtain such a tree either by (a) choosing an empty tree (g.f.: $z^0 = 1$); or (b) choosing a root with weight 1 (g.f. $1 \cdot z^1 = z$), since we can choose it in exactly

function representation rather than the original series—it's a "removable singularity" that can be replaced by the limit of $f(x)/g(x)$ as $x \to c$.

one way), and two subtrees (g.f. $= F^2$ where F is the g.f. for trees). This gives us a recursive definition

$$F = 1 + zF^2.$$

Solving for F using the quadratic formula gives

$$F = \frac{1 \pm \sqrt{1-4z}}{2z}.$$

That $2z$ in the denominator may cause us trouble later, but let's worry about that when the time comes. First we need to figure out how to extract coefficients from the square root term.

The binomial theorem says

$$\sqrt{1-4z} = (1-4z)^{1/2} = \sum_{n=0}^{\infty} \binom{1/2}{n} (-4z)^n.$$

For $n \geq 1$, we can expand out the $\binom{1/2}{n}$ terms as

$$\begin{aligned}
\binom{1/2}{n} &= \frac{(1/2)_{(n)}}{n!} \\
&= \frac{1}{n!} \cdot \prod_{k=0}^{n-1} (1/2 - k) \\
&= \frac{1}{n!} \cdot \prod_{k=0}^{n-1} \frac{1-2k}{2} \\
&= \frac{(-1)^n}{2^n n!} \cdot \prod_{k=0}^{n-1} (2k-1) \\
&= \frac{(-1)^n}{2^n n!} \cdot \frac{\prod_{k=1}^{2n-2} k}{\prod_{k=1}^{n-1} 2k} \\
&= \frac{(-1)^n}{2^n n!} \cdot \frac{(2n-2)!}{2^{n-1}(n-1)!} \\
&= \frac{(-1)^n}{2^{2n-1}} \cdot \frac{(2n-2)!}{n!(n-1)!} \\
&= \frac{(-1)^n}{2^{2n-1}(2n-1)} \cdot \frac{(2n-1)!}{n!(n-1)!} \\
&= \frac{(-1)^n}{2^{2n-1}(2n-1)} \cdot \binom{2n-1}{n}.
\end{aligned}$$

For $n = 0$, the switch from the big product of odd terms to $(2n-2)!$ divided by the even terms doesn't work, because $(2n-2)!$ is undefined. So here we just use the special case $\binom{1/2}{0} = 1$.

Now plug this nasty expression back into F to get

$$\begin{aligned}
F &= \frac{1 \pm \sqrt{1-4z}}{2z} \\
&= \frac{1}{2z} \pm \frac{1}{2z} \sum_{n=0}^{\infty} \binom{1/2}{n} (-4z)^n \\
&= \frac{1}{2z} \pm \left(\frac{1}{2z} + \frac{1}{2z} \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{2^{2n-1}(2n-1)} \binom{2n-1}{n} (-4z)^n \right) \\
&= \frac{1}{2z} \pm \left(\frac{1}{2z} + \frac{1}{2z} \sum_{n=1}^{\infty} \frac{(-1)^{2n-1} 2^{2n}}{2^{2n-1}(2n-1)} \binom{2n-1}{n} z^n \right) \\
&= \frac{1}{2z} \pm \left(\frac{1}{2z} + \frac{1}{2z} \sum_{n=1}^{\infty} \frac{-2}{(2n-1)} \binom{2n-1}{n} z^n \right) \\
&= \frac{1}{2z} \pm \left(\frac{1}{2z} - \sum_{n=1}^{\infty} \frac{1}{(2n-1)} \binom{2n-1}{n} z^{n-1} \right) \\
&= \frac{1}{2z} \pm \left(\frac{1}{2z} - \sum_{n=0}^{\infty} \frac{1}{(2n+1)} \binom{2n+1}{n+1} z^{n} \right) \\
&= \sum_{n=0}^{\infty} \frac{1}{(2n+1)} \binom{2n+1}{n+1} z^n \\
&= \sum_{n=0}^{\infty} \frac{1}{n+1} \binom{2n}{n} z^n.
\end{aligned}$$

Here we choose minus for the plus-or-minus to get the right answer and then do a little bit of tidying up of the binomial coefficient.

We can check the first few values of $f(n)$:

n	$f(n)$
0	$\binom{0}{0} = 1$
1	$(1/2)\binom{2}{1} = 1$
2	$(1/3)\binom{4}{2} = 6/3 = 2$
3	$(1/4)\binom{6}{3} = 20/4 = 5$

and these are consistent with what we get if we draw all the small binary trees by hand.

The numbers $\frac{1}{n+1}\binom{2n}{n}$ show up in a lot of places in combinatorics, and are known as the Catalan numbers.

10 Summary of operations on generating functions

The following table describes all the nasty things we can do to a generating function. Throughout, we assume $F = \sum f_k z^k$, $G = \sum g_k z^k$, etc.

Operation	**Effect on generating functions**	**Effect on coefficients**	**Combinatorial**
Coefficient extraction	$f_k = (1/k!)d^k/dz^k F(z)\|_{z=0}$		Find the number
Sum of all coefficients	$F(1)$	Computes $\sum f_k$	Count the total
Shift right	$G = zF$	$g_k = f_{k-1}$	Add 1 to the wei
Shift left	$G = z^{-1}(F - F(0))$	$g_k = f_{k+1}$	Subtract 1 from
Pointing	$G = z\frac{d}{dz}F$	$g_k = kf_k$	A G-thing is an
Sum	$H = F + G$	$h_k = f_k + g_k$	Every H-thing is
Product	$H = FG$	$h_k = \sum_i f_i g_{k-i}$	Every H-thing c
Composition	$H = F \circ G$	$H = \sum f_k G^k$	To make an H-th
Repetition	$G = 1/(1 - F)$	$G = \sum F^k$	A G-thing is a se

11 Variants

The **exponential generating function** or **egf** for a sequence $a_0, \dots$ is given by $F(z) = \sum a_n z^n/n!$. For example, the egf for the sequence $1, 1, 1, \dots$ is $e^z = \sum z^n/n!$. Exponential generating functions admit a slightly different set of operations from ordinary generating functions: differentiation gives left shift (since the factorials compensate for the exponents coming down), multiplying by z gives $b_n = na_{n+1}$, etc. The main application is that the product $F(z)G(z)$ of two egf's gives the sequence whose n-th term is $\sum (nchoosek) a_k b_{n-k}$; so for problems where we want that binomial coefficient in the convolution (e.g. when we are building weight n objects not only by choosing a weight-k object plus a weight-$(n-k)$ object but also by arbitrarily rearranging their unit-weight pieces) we want to use an egf rather than an ogf. We won't use these in CS202, but it's worth knowing they exist.

A **probability generating function** or **pgf** is essentially an ordinary generating function where each coefficient a_n is the probability that some random variable equals n. See RandomVariables for more details.

12 Further reading

RosenBook discusses some basic facts about generating functions in §7.4. ConcreteMathematics gives a more thorough introduction. Herbert Wilf's book *generatingfunctionology*, which can be downloaded from the web, will tell you more about the subject than you probably want to know.

See this page for very detailed notes on partial fraction expansion.