SOME OPERATORS FOR GENERATING FUNCTIONS

Suppose we have a collection of objects of various sizes. We want to build various configurations of the objects, like sequences of them, sets of them, etc. What size means depends on the problem, but it should have the property that if we group together N objects of sizes n_i, $i = 1, \ldots, N$, then the size of the collection should be $n_1 + \cdots + n_N$.

Let $A(x) = a_1x + a_2x^2 + a_3x^3 + \ldots$ be the OGF of the objects. We call A the *inventory GF* since it records the objects available to us (like we have a giant warehouse of these objects, and so this tells us what's available from there). Then it turns out that for certain configurations built from our inventory, we can apply some standard operators to A to build the GF counting them according to size. Many times one can go further to get explicit formulas, estimate how many sequences there are as a function of size, etc. But sometimes just building the GF is best one can hope for. The power here is that we can translate certain specifications directly into constructions, as if we were building a program by plugging together various modules. Even very complicated constructions can be easily handled.

Sequences. Sequences of length N (i.e. ordered tuples of length N) are counted by A^N. If we want to count all sequences, including the length 0 sequence, we use the operator

$$\mathrm{Seq}(A) = \frac{1}{1-A} = 1 + A + A^2 + A^{3+\cdots}.$$

To interpret the right hand side, read + as *or*. So it reads length 0 sequences or length 1 sequences or length 2 sequences ... This is an example of composition of GFs that we already studied.

For example, suppose we have 2×2 in two colors, and 3×3 flags in four colors. We want to put them on an n-foot flagpole. How many ways are there to do this? Answer: the inventory GF is $A = 2x^2 + 4x^3$. We want

$$\begin{aligned}\mathrm{Seq}(A) &= \frac{1}{1-(2x^2+4x^3)} \\ &= 1 + 2x^2 + 4x^3 + 4x^4 + 16x^5 + 24x^6 + 48x^7 + 112x^8 + 192x^9 + \cdots.\end{aligned}$$

Multisets. If we want multisets built from A, we use

$$\mathrm{Mult}(A) = \prod_{n\geq 1} \frac{1}{(1-x^n)^{a_n}}.$$

Notice that this isn't a composition. It's a different kind of transform of a GF.

For example, partitions are multisets of positive integers. The "size" of an integer is just the integer itself, and the so the total size is the number we're partitioning. So the inventory function is

$$A = x + x^2 + x^3 + \cdots \tag{0.1}$$

(because we have one integer of each size), and we have

$$\mathrm{Mult}(A) = \prod_{n\geq 1} \frac{1}{(1-x^n)^1},$$

which we know is the product formula for the parition GF

$$1 + x + 2x^2 + 3x^3 + 5x^4 + 7x^5 + 11x^6 + 15x^7 + 22x^8 + \cdots$$

If we have n different colors of the integer n, the inventory function is

$$A = x + 2x^2 + 3x^3 + 4x^4 + \cdots \tag{0.2}$$

and

$$\begin{aligned}\mathrm{Mult}(A) &= \prod_{n\geq 1} \frac{1}{(1-x^n)^n} \\ &= 1 + x + 3x^2 + 6x^3 + 13x^4 + 24x^5 + 48x^6 + 86x^7 + 160x^8 + \cdots\end{aligned}$$

(These are also called *plane partitions.*)

Sets. To build sets of elements from A, we use

$$\mathrm{Sets}(A) = \prod_{n\geq 1} (1+x^n)^{a_n}.$$

For instance, partitions with distinct parts are $\mathrm{Sets}(A)$ with A as in (0.1):

$$1 + x + x^2 + 2x^3 + 2x^4 + 3x^5 + 4x^6 + 5x^7 + 6x^8 + 8x^9 + 10x^{10} + 12x^{11} + \cdots$$

If each integer n comes in n colors, use (0.2):

$$1 + x + 2x^2 + 5x^3 + 8x^4 + 16x^5 + 28x^6 + 49x^7 + 83x^8 + 142x^9 + \cdots$$

Cycles. If we want to arrange objects into cycles (meaning we do sequences but consider two to be equivalent if we can cyclically permute one into the other), we use the operator (the most complicated of these)

$$\mathrm{Cyc}(A) = \sum_{k\geq 1} \frac{\varphi(k)}{k} \log \frac{1}{1-A(x^k)}.$$

To compute this

- φ is the Euler phi function
- The logarithm expression has the GF
$$\log \frac{1}{1-z} = z + \frac{z^2}{2} + \frac{z^3}{3} + \frac{z^4}{4} + \ldots$$
- $A(x^k)$ means we take A and substitute x^k for x.

For example, suppose we have three colors of beads, each of which has mass 1g, and we want to make different necklaces. The inventory is $A(x) = 3x$ (here size is the mass) and then

$$\begin{aligned}\mathrm{Cyc}(A) &= \sum_{k\geq 1} \frac{\varphi(k)}{k} \log \frac{1}{1-3x^k} \\ &= 3x + 6x^2 + 11x^3 + 24x^4 + 51x^5 + 130x^6 + \ldots\end{aligned}$$

If we have 3 types of 1g beads, 4 types of 2g beads, 5 types of 3g beads, then $A = 3x + 4x^2 + 5x^3$ and

$$\text{(0.3)}\quad \mathrm{Cyc}(A) = \sum_{k\geq 1} \frac{\varphi(k)}{k} \log \frac{1}{1 - 3x^k - 4x^{2k} - 5x^{3k}}$$
$$= 3x + 10x^2 + 28x^3 + 85x^4 + 272x^5 + 970x^6 + 3443x^7 + \cdots$$

Combining operations. We can use compositions of these operators to build some really complicated things. Suppose for instance we want to make necklaces out of partitions, then we want to arrange the necklaces into sequences. The size in this case is the sum of all the numbers represented by the little partitions in the final configuration. Then we start with

$$A = x + 2x^2 + 3x^3 + 5x^4 + 7x^5 + 11x^6 + 15x^7 + 22x^8 + 30x^9 + 42x^{10} + \cdots$$

which counts partitions (but doesn't include the constant term . . . that would be a basic object of size 0, and if we have that we don't have finitely many configurations of a fixed total size). Then we do

$$\mathrm{Seq}(\mathrm{Cyc}(A))$$
$$= 1+x+4x^2+13x^3+45x^4+150x^5+515x^6++1745x^7+5962x^8+20326x^9+\cdots$$

If we want to do sequences of partitions arranged into cycles:

$$\mathrm{Cyc}(\mathrm{Seq}(A) - 1)$$
$$= x+4x^2+12x^3+40x^4+126x^5+429x^6+1422x^7+4861x^8+16599x^9+\cdots$$

(we have to eliminate the empty sequence before we assemble into cycles). If we want cycles of partitions, then those arranged into cycles

$$\mathrm{Cyc}(\mathrm{Cyc}(A))$$
$$= x + 4x^2 + 10x^3 + 30x^4 + 79x^5 + 245x^6 + 709x^7 + 2192x^8 + 6709x^9 + \cdots$$

Try computing some of these coefficients by hand. Up to x^3 is usually enough to see what's going on and is doable.