NOTES ON OCT 18, 2012

MIKE ZABROCKI

Last time I said that when we have a combinatorial problem like

Find the number of solutions to the equation

$$x_1 + x_2 + x_3 + x_4 = n$$

where $x_i \geq 0$.

We can write down the generating function to this combinatorial problem $\frac{1}{(1-x)^4}$ and when we apply restrictions of the form $x_1 + x_2$ is even and $x_3 \leq 8$. In this case the generating function for the number of solutions to this equation is (which I will not justify because we have done a number similar problems)

$$= \frac{1}{2}\left(\frac{1}{(1-x)^2} + \frac{1}{(1+x)^2}\right)\frac{1-x^9}{1-x}\frac{1}{1-x}\ .$$

Most of the combinatorial problems that we can use this method on it will be possible to reduce them to a similar enumerative question.

There is another class of problems that is useful for the Mulitplication Principle of Exponential generating functions that I discussed last time. Consider problems like:

How many words (rearrangements of the letters) in the alphabet $\{a, b, c, d\}$ are there of length n?

Since our words are of length n, there are 4^n possible words with letters in $\{a, b, c, d\}$, each letter of the word has 4 choices. The exponential generating function for the number of these words is $\sum_{n\geq 0} 4^n \frac{x^n}{n!} = e^{4x}$. But what is kind of surprising is that I can also place restrictions on the letters and write down the exponential generating function for the sequence. Say that I consider the set of words

How many words are there in the alphabet $\{a, b, c, d\}$ such that there an even number of a's and b's (total) and at most 8 c's?

If we were to enumerate this using the multiplication principle and the addition priciple, then we would choose i spots from n for the a's and b's, choose a word in the a's and b's of length i, choose j of the remaining $n - i$ for the c's such that there are at most 8

c's, then the remaining $n-i-j$ spaces are where we place the d's. By the addition and multiplication principle of generating functions, we have

(1)

$$\sum_{i+j\leq n} \binom{n}{i} \left(\begin{matrix}\text{\# words length } i \text{ in } a \text{ and } b\\ \text{with an even \# } a\text{'s \&}b\text{'s}\end{matrix}\right) \binom{n-i}{j} \left(\begin{matrix}\text{\# words of length } j\\ \text{in } c \text{ with } \leq 8\ c\text{'s}\end{matrix}\right) \left(\begin{matrix}\text{\# words of length}\\ n-i-j \text{ in } d\end{matrix}\right)$$

If we combine the binomials $\binom{n}{i}$ and $\binom{n-i}{j}$ and note that it is equal to $\binom{n}{i,j,n-i-j} = \binom{n}{i}\binom{n-i}{j}$.

Last time I presented the multiplication principle of exponential generating functions. I will restate it here with multiple generating functions (while the last time it was a product of two).

Principle 1. *(The Multiplication Principle of Exponential Generating Functions) Let* $A_i(x) = \sum_{n\geq 0} a_n^{(i)} \frac{x^n}{n!}$*, then*

$$A_1(x)A_2(x)\cdots A_d(x) = \sum_{n\geq 0}\left(\sum_{i_1+i_2+\cdots+i_d=n}\binom{n}{i_1,i_2,\ldots,i_d} a_{i_1}^{(1)} a_{i_2}^{(2)}\cdots a_{i_d}^{(d)}\right)\frac{x^n}{n!}\ .$$

Alternatively the coefficient of $\frac{x^n}{n!}$ *in* $A_1(x)A_2(x)\cdots A_d(x)$ *is equal to*

$$\sum_{i_1+i_2+\cdots+i_d=n}\binom{n}{i_1,i_2,\ldots,i_d} a_{i_1}^{(1)} a_{i_2}^{(2)}\cdots a_{i_d}^{(d)}\ .$$

You should recognize that (1) is a special case of a coefficient of one of these coefficients. The expression in (1) is equal to the coefficient of $x^n/n!$ in the product

(2)

$$\left(\begin{matrix}\text{g.f. for words length } in \text{ in } a \text{ and } b\\ \text{with an even \# } a\text{'s \&}b\text{'s}\end{matrix}\right) \left(\begin{matrix}\text{g.f. for words of length } n\\ \text{in } c \text{ with } \leq 8\ c\text{'s}\end{matrix}\right) \left(\begin{matrix}\text{g.f. for words of length}\\ n \text{ in } d\end{matrix}\right)$$

Now I note that since there is precisely 1 word of length n using only the letter d then

$$\left(\begin{matrix}\text{g.f. for words of length}\\ n \text{ in } d\end{matrix}\right) = \sum_{n\geq 0}\frac{x^n}{n!} = e^x$$

Since there is one word of length n in the letters c unless $n > 8$, then

$$\left(\begin{matrix}\text{g.f. for words of length } n\\ \text{in } c \text{ with } \leq 8\ c\text{'s}\end{matrix}\right) = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \cdots + \frac{x^8}{8!}$$

Now if we insist that there are an even number of a's and b's then the there are 4 words of length 2 (aa, ab, ba, bb), there are 16 words of length 4 ($aaaa$, $aaab$, $aaba$, $\ldots$, $bbbb$). In general, the number of words of length n is 2^n if n is even and 0 if n is odd, hence the exponential generating function is equal to

$$\left(\begin{matrix}\text{g.f. for words length } in \text{ in } a \text{ and } b\\ \text{with an even \# } a\text{'s \&}b\text{'s}\end{matrix}\right) = 1+4\frac{x^2}{2!}+16\frac{x^4}{4!}+64\frac{x^6}{6!}+\cdots = \frac{1}{2}\left(e^{2x}-e^{-2x}\right) = cosh(2x)$$

Therefore putting this together with (2) we have that the coefficient of $x^n/n!$ in

$$cosh(2x)\left(1+\frac{x}{1!}+\frac{x^2}{2!}+\cdots+\frac{x^8}{8!}\right)e^x$$

is equal to the number of words in the alphabet $\{a, b, c, d\}$ such that there an even number of a's and b's (total) and at most 8 c's.

For example for the words of length 1 there is only c and d, for the words of length 2 we can have aa, bb, ab, ba, cc, cd, dc, dd so there are 8 words of length 2. For words of length 3 we can have caa, aca, aac, cbb, bcb, bbc, cab, acb, abc, cba, bca, bac, another 12 with a,b and ds and then 8 more are words in c and d (32 in total). In total there are We should then see that the series expands as $1 + 2\frac{x}{1!} + 8\frac{x^2}{2!} + 32\frac{x^3}{3!} + \cdots$. I will check this on the computer to show you how it is done.

```
sage: taylor(exp(x)*cosh(2*x)*sum(x^n/factorial(n) for n in range(9)),x,0,4)
16/3*x^4 + 16/3*x^3 + 4*x^2 + 2*x + 1
```

In general we have that ordinary generating functions used for counting problems that can be reduced to integer sum problems and exponential generating functions are useful for enumerating problems that can be reduced to enumerating words. It is also sometimes said that ordinary generating functions are good for enumerating "unlabeled" objects and exponential generating functions are good for enumerating "labeled" objets. This is a vague rule and hard to tell why this might be correct until we come with more examples of uses for ordinary and exponential generating functions. For example, we looked at the exponential generating function for the number of set partitions of n and this was e^{e^x-1} (this is a "labeled" object), we also started to look at partitions and ordinary generating functions.

We also talked about generating functions for partitions. I had given some of the definitions of partitions last time and I restated them. A partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_\ell)$ of n is a sequence of positive integers whose sum is n with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_\ell$. The size of a partition is the sum of the entries $\lambda_1 + \lambda_2 + \cdots + \lambda_\ell = n$. The length of the partition is ℓ, the number of entries in the sequence.

It is difficult to give the generating function for the number partitions of n in this form because we have this condition that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_\ell$, while we know how to give the generating function for the number of non negative integer solutions to $x_1+x_2+\cdots+x_r = n$ (with potentially other conditions), but there is a way of transforming the partitions into solutions to a similar system of equations.

Let $m_i(\lambda) =$ the number of parts of λ of size i (the number of $\lambda_d = i$. Then the size of the partition λ is equal to $n = \lambda_1 + \lambda_2 + \cdots + \lambda_\ell = 1m_1(\lambda) + 2m_2(\lambda) + 3m_3(\lambda) + \cdots$.

For example say that I wanted to compute the size of $(5, 2, 1, 1, 1)$. It is $10 = 5+2+1+1+1$, but since $m_1(5, 2, 1, 1, 1) = 3$, $m_2(5, 2, 1, 1, 1) = 1$, $m_3(5, 2, 1, 1, 1) = 0$, $m_4(5, 2, 1, 1, 1) = 0$, $m_5(5, 2, 1, 1, 1) = 1$ and the rest of the $m_i(5, 2, 1, 1, 1) = 0$ for $i > 5$ so the size of the partition is $1m_1(5, 2, 1, 1, 1) + 2m_2(5, 2, 1, 1, 1) + 3m_3(5, 2, 1, 1, 1) + 4m_4(5, 2, 1, 1, 1) + 5m_5(5, 2, 1, 1, 1) = 1 \cdot 3 + 2 \cdot 1 + 3 \cdot 0 + 4 \cdot 0 + 5 \cdot 1 = 3 + 2 + 5 = 10$.

If we look at all partitions this way we can say that all partitions are the number of solutions to the equations

$$m_1 + 2m_2 + 3m_3 + \cdots = n \tag{3}$$

with $m_i \geq 0$. Now we have phrased this question in terms of non-negative integer solutions equations and we can say that the generating function for the number of partitions of n is equal to the generating function for the number of solutions to equation (3). The generating function for the number of non-negative integer solutions to equation (3) is equal to the product of the generating functions for the number of non-negative integer solutions to $im_i = n$ over all possible $i \geq 1$. We know that the generating function for the number of non-negative solutions to the equation $im_i = n$ is equal to $\frac{1}{1-x^i}$, therefore the generating function for the number of partitions of n is equal to

$$\prod_{i \geq 1} \frac{1}{1-x^i} \ .$$

There is something a little odd about this formula because I am taking an infinite product. But because I can calculate the coefficient of x^n in this generating function by only taking the product of $\prod_{i=1}^{n} \frac{1}{1-x^i}$ (because the rest of the terms of the form $\frac{1}{1-x^{n+r}}$ for $r > 0$ don't affect the exponent of x^n), then I consider this a 'good' formula even though it seems to involve an infinite product. Since the calculation of any finite piece is finite and we can work with it (although carefully to ensure that any finite term of the series can always be computed in a finite number of steps).

Notice that if I want to compute the first 11 terms of series I just need to multiply the first 10 products together and so I can use sage to expand the series and sage also has functions which allow me to count the number of partitions of n. You should note in the code below the command `range(a,b)` are the integers i such that $a \leq i < b$ and `range(b)` are the integers $0 \leq i < b$.

```
sage: taylor(prod(1/(1-x^i) for i in range(1,11)),x,0,10)
42*x^10 + 30*x^9 + 22*x^8 + 15*x^7 + 11*x^6 + 7*x^5 + 5*x^4 + 3*x^3 + 2*x^2 + x + 1
sage: [Partitions(n).cardinality() for n in range(0,11)]
[1, 1, 2, 3, 5, 7, 11, 15, 22, 30, 42]
```

We then considered odd partitions, that is, partitions where all entries in the parts are odd. The number of odd partitions of n is equal to the number of non-negative integer solutions to the equation:

$$1m_1 + 3m_3 + 5m_5 + \cdots = n.$$

Leaving out the argument this time (because it seems we have done it so many times), the generating function for the number of odd partitions of n is equal to

$$\prod_{i \geq 0} \frac{1}{1-x^{2i+1}}$$

Again I can use sage to compute both the taylor series for the first 10 or so terms and use it to count the number of odd partitions of n. In the following snippit of code, I compute the Taylor series for the generating function and I also compute the partitions of n and then I restrict (filter) them so that I look at the ones where all entries are odd.

```
sage: taylor(prod(1/(1-x^(2*i+1)) for i in range(0,5)),x,0,10)
10*x^10 + 8*x^9 + 6*x^8 + 5*x^7 + 4*x^6 + 3*x^5 + 2*x^4 + 2*x^3 + x^2 + x + 1
sage: [Partitions(n).filter(lambda x: all(mod(v,2)==1 for v in x)).cardinality()
...: for n in range(0,11)]
[1, 1, 1, 2, 2, 3, 4, 5, 6, 8, 10]
```

Next we looked at strict partitions or partitions with distinct parts. A partition is called *strict* if there is at most one part of any given size (or otherwise stated, no parts are repeated). If we phrase this in terms of solutions to equations we would consider equations of the form

$$m_1 + 2m_2 + 3m_3 + \cdots = n$$

with $0 \leq m_i \leq 1$. The restriction that the parts are distinct (or the partition is strict) imposes the condition that m_i is either 0 or 1 since m_i represents the number of parts of size i. Again, without further explation the generating function for the number of solutions to these equations is

$$\prod_{i\geq 1}(1 + x^i)$$

Again I can use sage to calculate both the series and the number of such partitions. This time I looked in the documentation in order to find the number of partitions of n with distinct parts and it said the command is: `Partitions(n, max_slope=-1).cardinality()`.

```
sage: taylor(prod(1+x^i for i in range(1,11)),x,0,10)
10*x^10 + 8*x^9 + 6*x^8 + 5*x^7 + 4*x^6 + 3*x^5 + 2*x^4 + 2*x^3 + x^2 + x + 1
sage: [Partitions(n, max_slope=-1).cardinality() for n in range(11)]
[1, 1, 1, 2, 2, 3, 4, 5, 6, 8, 10]
```

Hmmm, I wonder if there is a connection between the number of strict partitions and the number of odd partitions?

Then I gave you a worksheet (which I will attach).