1. Basic enumeration and binomial coefficients

1. (A Course in Combinatorics, Problem 13A) We want to place the integers 1, 2, ..., \( r \) into a circular array with \( n \) positions so that they occur in order, clockwise, and such that consecutive integers (including the pair \((r, 1)\)) are not adjacent. Arrangements which are rotations of each other are considered the same. In how many ways can this be done?

Solution: Let \( a_1 \) be the number of empty spaces between 1 and 2, let \( a_2 \) be the number of empty spaces between 2 and 3 and so on, with \( a_r \) spaces between \( r \) and 1. Since we only care about configurations up to rotation, our configuration is determined by the \( r \)-tuple \((a_1, a_2, ..., a_r)\).

Since \( i \) and \( i+1 \) are not supposed to be adjacent, we must have \( a_i \geq 1 \). Set \( b_i = a_i - 1 \). We are supposed to have \( \sum a_i = n - r \), so \( \sum b_i = n - 2r \). From Theorem 13.1 in the book (or the corresponding result described in class), there are \( \binom{n-r-1}{r-1} \) such \( r \)-tuples.

2. (A Course in Combinatorics, Problem 13D) Consider the set \( S \) of all ordered \( k \)-tuples of subsets of \([n]\). What is \( \sum (A_1 \cup A_2 \cup \cdots \cup A_k) \)?

Solution: Let \( T \) be a subset of \([n]\) with \(|T| = r\). The number of \( k \)-tuples with \( T = A_1 \cup A_2 \cup \cdots \cup A_k \) is \((2^k - 1)^r\). (For each member \( x \) of \( r \), there are \( 2^k - 1 \) choices for which of the \( k \) sets \( A_i \) will contain \( x \), remembering that \( x \) must be in at least one of them.) So we want to evaluate

\[
\sum_T |T|(2^k - 1)^{|T|} = \sum_{r=0}^{n} \binom{n}{r} (2^k - 1)^r.
\]

Recall that

\[
\sum_{r=0}^{n} \binom{n}{r} x^r = (1 + x)^n \quad \text{and} \quad \sum_{r=0}^{n} r \binom{n}{r} x^r = nx(1 + x)^{n-1}.
\]

So \( \sum_{r=0}^{n} r \binom{n}{r} (2^k - 1)^r = n(2^k - 1)(2^k)^{n-1} = n(2^{kn} - 2^{k(n-1)}) \).

There is a slicker proof of this result; can you find it?

3. (A Course in Combinatorics, Problem 13H) Let \( A_n \) be the \( n \times n \) matrix whose \((i,j)\) entry is \( \binom{i}{j} \), with rows and columns numbered starting from 0. So, for example,

\[
A_5 = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 \\
1 & 2 & 1 & 0 & 0 \\
1 & 3 & 3 & 1 & 0 \\
1 & 4 & 6 & 4 & 1
\end{pmatrix}.
\]

Compute \( A_2^{-1} \), \( A_3^{-1} \) and \( A_4^{-1} \). Find and prove a formula for \( A_n^{-1} \).
Solution: Quick computations will suggest the conjecture that the inverse is \(((-1)^{i-j}\binom{i}{j})\). For example, 

\[
A_5^{-1} = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
-1 & 1 & 0 & 0 & 0 \\
1 & -2 & 1 & 0 & 0 \\
-1 & 3 & -3 & 1 & 0 \\
1 & -4 & 6 & -4 & 1 \\
\end{pmatrix}.
\]

So, let’s prove this. Let \(B_n\) be the matrix whose \((i,j)\) entry is \((-1)^{i-j}\binom{i}{j}\); we want to show that \(A_nB_n\) is the identity. Since a product of lower triangular matrices is lower triangular, \(A_nB_n\) vanishes above the diagonal and, since the diagonals of \(A_n\) and \(B_n\) are both 1, so is the diagonal of \(A_nB_n\). So the interesting task is to show that the entries below the diagonal vanish.

Let’s consider the entry in position \((i,k)\), with \(i > k\). This is

\[
\sum_j (-1)^{i-k} \binom{i}{j} \binom{j}{k}.
\]

There are a number of good ways to show this is zero. I’ll show how to get in from generating functions. Recall that \(\sum_j \binom{i}{j} x^{i-j} = (1 + x)^i\). And \(\sum_j (-1)^{i-j} \binom{j}{k} x^j = x^k/(1 + x)^{k+1}\). So our sum is the coefficient of \(x^i\) in \((1 + x)^i \times x^k/(1 + x)^{k+1}\).

This is \((1 + x)^{i-k-1} x^k\). Notice that the exponent is nonnegative, since \(i > k\). But the polynomial \((1 + x)^{i-k-1} x^k\) only has degree \(i - 1\). So the coefficient of \(x^i\) is zero, just as we wanted.

4. (Putnam, 1985, A-1) How many ordered triples of sets \((A, B, C)\) are there such that \(A \cup B \cup C = [n]\) and \(A \cap B \cap C = \emptyset\)?

Each element of \([n]\) must lie in at least one, but not all three, of \(A, B\) and \(C\). There are six possibilities (\(A\) but not \(B\) or \(C\), \(B\) but not \(A\) or \(C\) and so forth), so there are \(6^n\) possibilities altogether.

5. Mathematics for the analysis of Algorithms, C.1 In a bubble sort, one is given a list of \(n\) distinct numbers to sort into order. One first compares the first two numbers, interchanging them if they are out of order. One then compares the second and third, then the third and fourth and so on. After one has done \(n - 1\) comparisons, one returns to the start of the list and does it again.

Of the \(n!\) possible starting arrangements, how many will be correctly sorted at the end of the first pass? At the end of the first two passes? (I recommend you gather some data before you try to answer this.)

(Harder) In a cocktail-shaker sort, one first compares the first two numbers, then the second and third and so forth as before but, after comparing the \((n - 1)\)st and \(n\)th numbers, one backtracks to compare \((n - 2)\)nd and \((n - 1)\)st, then \((n - 3)\)rd and \((n - 2)\)nd and so forth. Let \(x_n\) be the number of arrangements that are sorted by one pass each way. Deduce the recurrence:

\[
x_n = x_{n-1} + x_{n-1} + 2x_{n-2} + 4x_{n-3} + 8x_{n-4} + 16x_{n-5} + \cdots.
\]

Solution More generally, we will show that, for \(n \geq k\), the number of permutations sorted in \(k\) passes of a bubble sort is \(1 \times 2 \times 3 \times \cdots \times k \times (k + 1)^{n-k} = k!(k + 1)^{n-k}\). (When \(n < k\), we will show that all permutations are sorted, so the count is \(n!\).)

Consider a permutation \(a_1a_2\cdots a_n\) of \([n]\), and let us consider whether it can be sorted in \(\leq k\) passes. Notice that \(a_n\) can only be moved forward \(k\) times. So, if \(a_n < n - k\), it is certainly impossible to sort in \(k\) passes.
Now suppose that \( a_n \geq n - k \). I claim that \( a_1a_2 \cdots a_n \) is sorted in \( \leq k \) passes if and only if \( a_1a_2 \cdots a_n \) is sorted in \( \leq k \) passes. Set \( a_n = n - t \) for \( t \leq k \). Let \( a_{n-1}^1 \cdots a_{n-1}^t \) be the result of the first pass of bubble sorting \( a_1 \cdots a_{n-1} \); let \( a_{n-1}^2 \cdots a_{n-1}^2 \) be the result of the second pass and so forth.

Then, when we bubble sort \( a_1a_2 \cdots a_n \), here is what happens. When the first pass sweeps through \( a_1 \cdots a_{n-1} \), we get \( a_1^1 \cdots a_{n-1}^1 a_n \). Then, when we make the final comparison between \( a_{n-1}^1 \) and \( a_n \), we have \( a_{n-1}^1 = n \), so they get swapped. So, after the first pass, we have \( a_1^1 \cdots a_{n-2}^1 (n - t)n \). Similarly, after the second pass we have \( a_1^2 \cdots a_{n-3}^2 (n - t)(n - 1)n \). And so forth until the end of the \( t \)-th pass, when we have \( a_1^t \cdots a_{n-t-1}^t (n - t)(n - t + 1) \cdots (n - 1)n \). Then, continuing for \( k - t \) more passes, we get \( a_1^k \cdots a_{n-t-1}^k (n - t)(n - t + 1) \cdots (n - 1)n \). So we see that \( a_1 \cdots a_n \) winds up sorted in \( \leq k \) passes if and only if \( a_n \geq n - k \) and \( a_1 \cdots a_{n-1} \) gets sorted in \( \leq k \) passes.

Let \( S_n^k \) be the number of \( k \)-pass-sortable permutations of \([n]\). We can describe such a permutation by choosing \( a_n \) to be one of the \( \min(k + 1, n) \) numbers \( \{ n - k, n - k + 1, \ldots, n \} \), and choosing a \( k \)-pass-sortable permutation of \([n - 1]\). So we get \( S_n^k = (k + 1)S_{n-1}^k \) for \( n \geq k + 1 \) and we deduce the formula above.

**The harder problem:**

Consider a permutation \( a_1a_2 \cdots a_n \). We will break into cases based on the value of \( a_1 \).

If \( a_1 = 1 \), then \( a_1 \cdots a_n \) is sorted in one back-and-forth pass if and only if \( a_2 \cdots a_n \) is. So that gives us \( x_{n-1} \) such permutations.

Now let \( a_1 = t \), with \( t > 1 \). Suppose that \( a_1, a_2, \ldots, a_{t-1} \) are \( < t \) while \( a_t > t \). While the sorting is going forward, \( t \) moves forward \( t - 1 \) times. When the sorting comes back the other way, \( t \) either moves 0 or 1 more times. So we can only have a perfect sorting if \( i \) is either \( t - 1 \) or \( t \). We break into cases depending on whether \( i \) is \( t - 1 \) or \( t \).

Suppose that \( i = t \). So the sorting process starts off moving \( t \) forward to make \( a_2a_3 \cdots a_t a_{t+1} \cdots a_n \). Then, we apply a single back-and-forth pass to \( a_{t+1} \cdots a_n \). Finally, we apply a single bubble pass to \( a_2a_3 \cdots a_t \). There are \( x_{n-t} \) ways for \( a_{t+1} \cdots a_n \) to be sorted in one back-and-forth pass, and (by the previous part of the problem) \( 2^t - 2 \) ways for \( a_2a_3 \cdots a_t \) to be sorted in one bubble pass. So this contributes \( 2^{t-2}x_{n-t} \) to the sum.

Suppose now that \( i = t - 1 \). Let \( u \) be the element of \( \{ 1, 2, \ldots, t-1 \} \) that is not among \( \{ a_2, a_3, \ldots, a_{t-1} \} \). Then the first part of the sort gets us to \( a_2a_3 \cdots a_{t-1}a_t \cdots a_n \). The next part is a single back-and-forth pass along \( a_t \cdots a_n \). After this is done, the \( t \)-th position is occupied by \( u \). Then \( u \) is swapped with \( t \). Finally, we have a single bubble pass on \( a_2a_3 \cdots a_{t-1}u \). There are \( 2^t - 2 \) ways for this final bubble pass to work. We might think that if there are \( x_{n-t+1} \) ways for \( a_t \cdots a_n \) to be sorted in a single pass, but be careful! We only want to count such ways where \( a_t \) is not \( u \); the ones where \( a_t = u \) are counted in the previous case. The number of cases that we must be careful not to count is precisely the number of ways to have \( a_t = u \) and have \( a_{t+1} \cdots a_n \) be sorted in a single back-and-forth pass, which is \( x_{n-t} \). All in all, this case contributes \( 2^{t-2}(x_{n-t+1} - x_{n-t}) \).

Adding it all up,

\[
x_n = x_{n-1} + [x_{n-3} + (x_{n-2} - x_{n-2})] + 2[x_{n-4} + (x_{n-3} - x_{n-4})] + 4[x_{n-5} + (x_{n-4} - x_{n-5})] \cdots .
\]

Simplifying a little, we get the claim.

### 2. Linear Recurrences

**6.** You may want to use a computer for this one: Amber likes to choose pronounceable passwords. This means that her passwords are made of uppercase letters of the alphabet and they never contain
two (or more) consecutive vowels nor three (or more) consecutive consonants\textsuperscript{1} For example, one of her passwords might be \textsc{radsotanked}. Let $a_n$ be the number of passwords of length $n$ that Amber can choose. Find $\sum a_n x^n$. The quantity $a_n$ grows like $\alpha^n$ for some $\alpha$: find it. Beth composes her passwords out of uppercase letters as well, but obey no other restrictions. How long should Amber choose a password, in order to have as many options as Beth gets from an ten character password?

\textbf{Solution:}

There are a couple of ways to do this, here is one of the faster ones. Every one of Amber’s passwords either begins with 0, 1 or 2 consonants, followed by a sequence of blocks of the form $VC$ or $VCC$, and finally either stopping or adding on a terminal vowel. So the generating function for the number of passwords of length $n$ is

\[ (1 + 21x + 21^2x^2)[1 + (5 \cdot 21x^2 + 5 \cdot 21^2x^3) + (5 \cdot 21^2x^2 + 5 \cdot 21^2x^3) + \cdots ](1 + 5x) \]

\[ = (1 + 5x)(1 + 21x + 441x^2) \]

\[ 1 - 105x^2 - 2205x^3. \]

The denominator $1 - 105x^2 - 2205x^3$ factors as $(1 - \alpha_1 x)(1 - \alpha_2 x)(1 - \alpha_3 x)$, where $\alpha_1 \approx 15.67$, and $\alpha_2$ and $\alpha_3$ are $\approx -7.84 \pm 8.90i$. The root $\alpha_1$ is the largest of the three, so $x_n$ grows approximately like $\alpha_1^n$.

Beth has $26^{10}$ passwords of length 10. So, roughly, we want to have $\alpha_1^n = 26^{10}$, or $n = 10 \log 26/ \log \alpha_1 \approx 11.8$.

As a check on this, I had Mathematica compute the exact answer, with the command $\text{Series}[(1 + 5 x) (1 + 21 x + 441 x^2)/(1 - 105 x^2 - 2205 x^3), \{x,0,13\}]$. It turns out that the coefficients of $x^{11}$ and $x^{12}$ are $2.96 \times 10^{13}$ and $4.62 \times 10^{14}$, nicely flanking $26^{10} \approx 1.41 \times 10^{14}$.

Of course, please remember the message of http://xkcd.com/792/. Good passwords are useless if you reuse them!

\textbf{7.} Let $F_n$ be the Fibonacci numbers, defined by $F_0 = 0$, $F_1 = 1$ and $F_n = F_{n-1} + F_{n-2}$. Show that there are constants $a$, $b$ and $c$ such that $F_0 + F_1 + \cdots + F_n = aF_n + bF_{n-1} + c$. For elegance points, prove this without going to the trouble of computing $(a,b,c)$.

\textbf{Solution:} The generating function for Fibonacci numbers is

\[ \frac{x}{1 - x - x^2}. \]

The generating function for sums of Fibonacci numbers is

\[ \frac{x}{(1 - x - x^2)(1 - x)}. \]

By partial fraction expansion, there are some $A$, $B$ and $C$ such that

\[ \frac{x}{(1 - x - x^2)(1 - x)} = \frac{Ax + B}{1 - x - x^2} + \frac{C}{1 - x}. \]

We have $\sum_{k=0}^n F_k = AF_n + BF_{n+1} + C$.

\textbf{8.} (Concrete Mathematics, Exercise 7.7) Solve the recurrence:

\[ g_0 = 1 \]
\[ g_n = g_{n-1} + 2g_{n-2} + \cdots + ng_0 \]

\textbf{Solution:} Let $G(z) = \sum g_n z^n$. Recall that $\sum k z^k = z/(1 - z)^2$. So

\[ G(z) = 1 + \frac{z}{(1 - z)^2}G(z). \]

\textsuperscript{1}For simplicity, treat $Y$ as a consonant.
Rearranging,
\[ G(z) = \frac{(1-z)^2}{1-3z+z^2} = 1 + \frac{z}{1-3z+z^2}. \]
For \( n \) greater than 0, partial fraction expansion then gives
\[ G_n = \frac{1}{\sqrt{5}} \left( \tau_n^+ - \tau_n^- \right) \]
where \( \tau_{\pm} = \left( 3 \pm \sqrt{5} \right)/2 \). You might recognize the \( g \)'s as the even Fibonacci's.

9. \textit{(Concrete Mathematics, Exercise 7.23)} In how many ways can a \( 2 \times 2 \times n \) tower be built out of \( 1 \times 1 \times 2 \) bricks?

\textbf{Solution:} Let \( a_n \) be the number of \( 2 \times 2 \times n \) towers. Let \( b_n \) be the number of such towers with a single brick shaped notch removed from the top. We have
\[
\begin{align*}
a_n &= 2a_{n-1} + a_{n-2} + 4b_{n-1} \\
b_n &= a_{n-1} + b_{n-1}
\end{align*}
\]
with initial conditions \( a_0 = 1, \ b_0 = 0 \).

We get
\[
\begin{align*}
A(x) &= 2xA(x) + x^2A(x) + 4xB(x) \\
B(x) &= xA(x) + xB(x)
\end{align*}
\]
Solving the linear equations,
\[ A(x) = \frac{1-x}{(1+x)(1-4x+x^2)}. \]
Expanding in partial fractions, we get the answer
\[ \frac{2 + \sqrt{3}}{6} (2 + \sqrt{3})^n + \frac{2 - \sqrt{3}}{6} (2 - \sqrt{3})^n + \frac{1}{3} (-1)^n. \]

10. In problem 10, you derived the recursion
\[ x_n = x_{n-1} + x_{n-1} + 2x_{n-2} + 4x_{n-3} + 8x_{n-4} + 16x_{n-5} + \cdots \]
with \( x_n = 1 \). Find a closed form for \( x_n \).

\textbf{Solution:} Let \( F(z) \) be the generating function \( \sum x_n z^n \). (Somehow, \( X(z) \) just feels wrong.) So
\[ F(z) = zF(z) + (z + 2z^2 + 4z^3 + 8z^4 + \cdots) F(z) + 1 = (z + \frac{z}{1-2z}) F(z) + 1. \]
We derive that
\[ F(z) = \frac{1 - 2z}{1 - 4z + 2z^2}. \]
Let \( \sigma_{\pm} = 2 \pm \sqrt{2} \), so \( 1 - 4z + 2z^2 = (1 - \sigma_+ z)(1 - \sigma_- z) \). Then
\[ F(z) = \frac{1}{2} \left( \frac{1}{1-\sigma_+ z} + \frac{1}{1-\sigma_- z} \right) \]
so our answer is
\[ x_n = \frac{\sigma_+^n + \sigma_-^n}{2}. \]