Mathematical Foundations of Computer Science

CS 499, Shanghai Jiaotong University, Dominik Scheder

- Homework assignment published on Tuesday, 2018-03-13
- Submit questions and first solutions by Sunday, 2018-03-18, 12:00 by email to dominik.scheder@gmail.com and the TAs.
- You will receive feedback by Wednesday, 2018-03-21
- Revise your solution and submit your final solution by Sunday, 2018-03-25 by email to dominik.scheder@gmail.com and the TAs.

3 Basic Counting

A function $[m] \to [n]$ is *monotone* if $f(1) \leq f(2) \leq \cdots \leq f(m)$. It is *strictly monotone* if $f(1) < f(2) < \cdots < f(m)$.

Exercise 3.1. Find and justify a closed formula for the number of strictly monotone functions from $[m]$ to $[n]$.

Solution. $\binom{n}{m}$. ■

Exercise 3.2. Find and justify a closed formula for the number of monotone functions from $[m]$ to $[n]$.

Solution. n^m. ■

Remark. By "closed" I mean something using expressions like $\times$, $+$, $\binom{n}{k}$, $n!$, but not $\sum$ or $\prod$. For example, $\binom{n}{k^2}$ is a closed formula but $\sum_{k=0}^{n}\binom{n}{k}$ is not.

Exercise 3.3. Prove that $\sum_{k=0}^{n}\binom{n}{k}^2 = \binom{2n}{n}$ for every $n \geq 0$ by finding a combinatorial interpretation.

Proof. We divide the set of $2n$ elements into 2 sets of n elements. For each $0 \leq k \leq n$, we pick out k elements from one set, exclude k elements in the other set and combine them, we will get n elements, which is equivalent to selecting n elements from $2n$ elements.

Therefore, $\sum_{k=0}^{n}\binom{n}{k}^2 = \binom{2n}{n}$ for every $n \geq 0$.

Exercise 3.4. [From the textbook] Find a closed formula for $\sum_{k=m}^{n}\binom{k}{m}\binom{n}{k}$ and prove it combinatorially, i.e., by giving an interpretation.

Solution. $\sum_{k=m}^{n}\binom{k}{m}\binom{n}{k} = \binom{n}{m} \cdot 2^{n-m}$.

Proof. The formula means for each $m \leq k \leq n$, we select k elements from n elements, then select m elements from these k elements, which is equivalent to first choose n elements from m elements, then decide whether to pick out the remaining elements that is $\binom{n}{m} \cdot 2^{n-m}$. ∎

Exercise 3.5. Let B_n be the number of partitions of the set $[n]$ (this is the same as the number of equivalence relations on $[n]$). This is called the Bell number, thus we denote it B_n. Prove that the following recursive formula for B_n is correct:

$$B_0 = 1$$
$$B_{n+1} = \sum_{k=0}^{n}\binom{n}{k} B_k \ .$$

Proof. $B_0 = 1$ is correct, undoubtedly.

Denote the elements in the set $\{x_1, x_2, \ldots, x_n, \ldots\}$.

Assume that for $B_0, B_1, \ldots, B_n$, the formula is right. Then we add a element x_{n+1} to the set. How many options do we have?

For each $0 \leq k \leq n$, we select $n-k$ elements from the former n elements and combine them with the element n_{k+1} which contains $\binom{n}{n-k} = \binom{n}{k}$ ways, then partition the remaining k elements. Therefore we have $B_{n+1} = \sum_{k=0}^{n}\binom{n}{k} B_k$.

Exercise 3.6. Let P_n be the number of ways to write the natural number n as a sum $a_1 + a_2 + \cdots + a_k$ such that $1 \leq a_1 \leq a_2 \leq \cdots \leq a_k$. For example, 3 can be written as 3, $2+1$, and $1+1+1$, so $P_3 = 3$. Find a recursive formula for P_n.

Remark. The formula might not be as simple as the above for B_n. Be creative! Start by writing a simple recursive program that computes P_n.

Solution. Let $G_i(n)$ be the number of ways to wirte the natural number n as a sum of i numbers following the rules $1 \leq a_1 \leq a_2 \leq \cdots \leq a_i$. Speacially we define $G_i(k) = 0$ where $i > k$.
Then we have $P_n = G_1(n) + G_2(n) + \cdots + G_n(n)$.
Consider wirting n as a sum of i numbers $a_1, a_2, \ldots a_i$ with $1 \leq a_1 \leq a_2 \leq \cdots \leq a_i$, and $a_1, a_2 \ldots a_k$ are 1s, $a_{k+1} \ldots a_i$ are larger than 1:
Substract these numbers by 1, then we get $0, 0 \ldots 0, a^*_{k+1}, \ldots a^*_i$.
For any k with $1 \leq k \leq i$, the sum of $a^*_{k+1}, \ldots a^*_i$ is $(n-i)$. Then the number of ways to wirte i numbers which includes k 1s is equal to $G_{i-k}(n-i)$.
In this way, $G_i(n) = \sum_{k=0}^{i-1} G_{i-k}(n-i)$ and $P_n = \sum_{i=1}^{n-1} \sum_{k=0}^{i-1} G_{i-k}(n-i) + G_n(n) = \sum_{i=1}^{n-1} \sum_{k=0}^{i-1} G_{i-k}(n-i) + 1$
Therefore $P_n - P_{n-1} = \sum_{k=1}^{n-1} G_k(n-k) = [n/2]$, so $P_n = P_{n-1} + [n/2]$. ■