CS/Math 240: Introduction to Discrete Mathematics 4/28/2011

Lecture 25 : Counting

Instructor: Dieter van Melkebeek Scribe: Dalibor Zelený

DRAFT

Today we start the last topic of the course, counting. For example, in program analysis we want to count the number of times a loop repeats. Often, we are interested in the cardinality of some set. For example in discrete probability theory, which we won't get to in this course, we want to count the size of the set of "good" outcomes and then related to the size of the set of all possible outcomes.

We begin with some basic counting techniques which we illustrate on multiple examples. After that, we generalize some of the basic techniques and give examples of their applications.

25.1 Basic Counting Techniques

We describe some basic counting rules and apply them in nontrivial ways.

25.1.1 Bijection Rule

We saw in Lecture 16 that if a function $f : T \to S$ is a total bijection, then $|S| = |T|$. Thus, in order to find the cardinality of S, we can, instead, find a bijection from T to S and find the cardinality of T.

We often use the set of sequences over some alphabet that satisfy some property as the set T.

25.1.2 Sum Rule

Suppose S is the *disjoint union* of sets $S_1, S_2, \ldots$. Being a disjoint union means that the sets $S_1, S_2, \ldots$ form a partition of S. When the union is disjoint, we use the symbol $\dot\cup$ instead of $\cup$, and we write $S = \dot\bigcup_i S_i$. With this notation at hand, we see that

$$|S| = \sum_i |S_i| . \tag{25.1}$$

25.1.3 Product Rule

Now suppose S is the Cartesian product of sets $S_1, S_2, \ldots$. Then every element of S is an ordered sequence of elements $s_1, s_2, \ldots$ where $s_i \in S_i$, We use the notation $S = \prod S_i$ to denote this. Since the choices for each position in the sequence are independent, we have

$$|S| = \prod_i |S_i| , \tag{25.2}$$

where the product symbol on the left-hand side means Cartesian product, and the product symbol on the right-hand side is multiplication.

25.1.4 Examples of Basic Counting Techniques

Through examples, we now show some applications of the three techniques we described.

Example 25.1: You are in a store that sells five different kinds of bagels: plain, poppy seed, sesame seed, onion, and with all three toppings. You want to buy a dozen bagels and can combine the different kinds in any way you want. You want to know how many different combinations of bagels you can pick.

Let S be the set of all possible combinations of 12 bagels, and let T be the set of binary strings of length 16 that contain exactly four 1s. We demonstrate a bijection from S to T in a moment. First let's see it on an example. Suppose you buy 2 plain bagels, 0 poppy seed bagels, 6 sesame seed bagels, 3 onion bagels, and 1 bagel with all toppings. These correspond to strings of zeros of lengths 2, 0, 6, 3, and 1, respectively. Now put ones between these strings of zeros and concatenate everything to get the string 0011000000100010. The number of zeros between two consecutive ones (or between the beginning of the string and the first one, or between the last one and the end of the string) indicates the number of bagels of one kind.

In general, suppose we buy x_1 plain bagels, x_2 poppy seed bagels, x_3 sesame seed bagels, x_4 onion bagels, and x_5 bagels with everything. Let's denote this choice using the tuple $(x_1, x_2, \ldots, x_5)$. Note that $\sum_{i=1}^{5} x_i = 12$. We map this choice of bagels to the string $0^{x_1}10^{x_2}10^{x_3}10^{x_4}10^{x_5}$, i.e, the function f is defined by $f(x_1, \ldots, x_5) = 0^{x_1}10^{x_2}10^{x_3}10^{x_4}10^{x_5}$. This map is a total function because we can map any choice of bagels to a string of length 16 using the strategy above. To see that this map is surjective, consider any string of the form $0^{x_1}10^{x_2}10^{x_3}10^{x_4}10^{x_5}$ of length 16 with four ones. This string corresponds to the choice of x_1 plain bagels, x_2 poppy seed bagels, and so on. Finally, the map is injective. To see this, suppose $f(x_1, \ldots, x_5) = f(y_1, \ldots, y_5)$, so $0^{x_1}10^{x_2}10^{x_3}10^{x_4}10^{x_5} = 0^{y_1}10^{y_2}10^{y_3}10^{y_4}10^{y_5}$. But then $x_i = y_i$ for all i.

Thus, f is a bijection, and it follows that the number of ways we can pick 12 bagels is the same as the number of binary strings of length 16 with exactly four ones by the bijection rule. As we shall see at the end of lecture, this number is 1820. ⊠

Example 25.2: Now let's count the number of subsets of a domain D of size n. Let's call this set S. We give a bijection from the set T of all binary strings of length n to the set S.

Fix an enumeration $d_1, d_2, \ldots, d_n$ of D, and construct the map $f : T \to S$ by

$$f(x_1 x_2 \ldots x_n) = \{d_i \mid x_i = 1\}.$$

This is a total bijection because we map every binary string to some subset of D. It is surjective because a subset A is the image of the string $x_1 x_2 \ldots x_i$ where $x_i = 1$ if $d_i \in S$ and $x_i = 0$ otherwise. The function is injective too. Thus, $|S| = |T|$.

Because $T = \{0,1\}^n = \{0,1\} \times \{0,1\} \times \cdots \times \{0,1\}$, i.e., T is the n-fold Cartesian product of the set $\{0,1\}$ with itself,. it is possible to find $|T|$ using the product rule. Since we can pick each position in a binary string of length n independently of all the other positions, and there are two choices for each position, we get $|T| = 2^n$, and this is also the number of subsets of D. ⊠

Example 25.3: Say some website requires users to have passwords that have between 6 and 8 characters. Furthermore, the first character should be a letter, either uppercase or lowercase, and the remaining characters can be uppercase or lowercase letters or one of the digits 0 through 9.

Let F be the set of all possible characters for the first character in the password, and let N be the set of all possible characters for the symbols that follow. We have $|F| = 52$ and $|N| = 62$.

Let P_i be the set of all possible passwords of length i, and note that the set of all possible passwords is the disjoint union $P_6 \dot\cup P_7 \dot\cup P_8$. Thus, we can use the union rule to count the number of possible passwords.

Before we apply the union rule, we need to determine $|P_i|$. For that, we use the product rule. The choices of the individual characters are independent, and we have $P_i = F \times N^{i-1}$ where N^{i-1} stands for the $(i-1)$-fold Cartesian product of N with itself. Thus, we have $|P_i| = |F||N|^{i-1}$.

Finally, the set of all possible passwords has cardinality

$$|P_6| + |P_7| + |P_8| = 52 \cdot 62^5 + 52 \cdot 62^6 + 52 \cdot 62^7 = 186125210680448,$$

which is roughly $1.9 \cdot 10^{14}$. ⊠

25.2 Generalized Counting Techniques

Now that we have seen some basic applications of counting, it's time to look at generalizations of the techniques we have used so far.

25.2.1 Generalized Product Rule

In the applications of the product rule we have seen, we assumed that the choices for each component of a Cartesian product were independent of each other. We now relax this independence condition.

Let S be the set of sequences of length k. Suppose there are n_1 choices for the first term of the sequence. After the first term is fixed, there are n_2 choices for the second term. After the first two terms are fixed, there are n_3 choices for the third term. This continues until the end of the sequence, where n_k gives the number of choices for the k-th term given that the first $k-1$ terms have been fixed. In this case,

$$|S| = \prod_{i=1}^{i} n_i. \tag{25.3}$$

Example 25.4: Given a chess board (a board with 8 rows and 8 columns), we want to find the number of ways to place three different pieces, say a pawn, a bishop, and a queen, so that no two pieces are in the same row, and no two pieces are in the same column.

Let S be the number of ways to place the pieces according to our rules, and let T be the set of sequences $(r_p, c_p, r_b, c_b, r_q, c_q) \in \{1, \ldots, 8\}^6$ such that r_p, r_b and r_q are all different, and also c_p, c_b and c_q are all different. The meanings of the six coordinates are the row and column of the pawn, the row and column of the bishop, and the row and column of the queen. This gives us a bijection between S and T.

We have 8 options for the values of r_p and c_p, so we have $n_1 = n_2 = 8$ in the sense of (25.3). Now we have the constraint $r_b \neq r_p$, so the number of options for r_b once r_p and c_p were picked is 7. Similarly, there are 7 options for the value of c_b, so $n_3 = n_4 = 7$. Finally, $r_q \notin \{r_p, r_b\}$ and $c_q \notin \{c_p, c_b\}$, so $n_5 = n_6 = 6$. Finally, using the generalized product rule yields $|S| = |T| = 8 \cdot 8 \cdot 7 \cdot 7 \cdot 6 \cdot 6 = 112896$. ⊠

The next example has a more mathematical flavor to it. We need the following definition.

Definition 25.1. *A* permutation *of a domain D is a sequence π consisting of elements of D such that each element of D appears exactly once in it.*

Example 25.5: We count the number of permutations of a domain D with $|D| = n$. Let S be the set of all permutations of D. Since each element of D appears exactly once in the permutation, the sequence that represents a permutation has length n.

Once we pick the first term in the sequence, we have $n-1$ options for the second term. After that, two terms of the sequence have been chosen. Since those two terms are different and each element of D can only appear once in the permutation, there are $n-2$ options for the third term. This continues until the very last term. In that case, $n-1$ terms have been determined and are all distinct, so there is exactly one choice for the last term. Thus, by the generalized product rule, we have $|S| = n(n-1)(n-2)\cdots 2\cdot 1$. ⊠

The quantity from Example 25.5 is widely used, so we give it a name.

Definition 25.2. *The* factorial *of n, denoted $n!$, and read "n factorial", is the product of the first n positive integers. That is,*

$$n! = \prod_{i=1}^{n} i.$$

For convenience, we also define $0! = 1$.

If you consider Example 25.5, defining $0! = 1$ makes sense. There is exactly one way to write down the list of the elements of an empty set. Just write down the empty sequence.

Let's derive some bounds for the value of $n!$. First, $n!$ is the product of n integers, each of which is at most n, so $n! \leq n^n = 2^{n\log n}$ (where the logarithm is base 2). Second, the terms $n, n-1, \ldots, n/2+1, \ldots, n-\lfloor n/2\rfloor+1$ are all at least $n/2$, and there are at least $n/2$ of them, so $n! \geq (n/2)^{n/2} = 2^{(n/2)(\log n-1)}$.

The estimates from the previous paragraph are not exactly tight. There is a much better approximation of $n!$ called *Stirling's approximation* which says that

$$n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n. \tag{25.4}$$

Recall that $\sim$ means asymptotic equivalence (see Lecture 14). We do not prove (25.4) in this course.

25.2.2 Generalized Bijection Rule

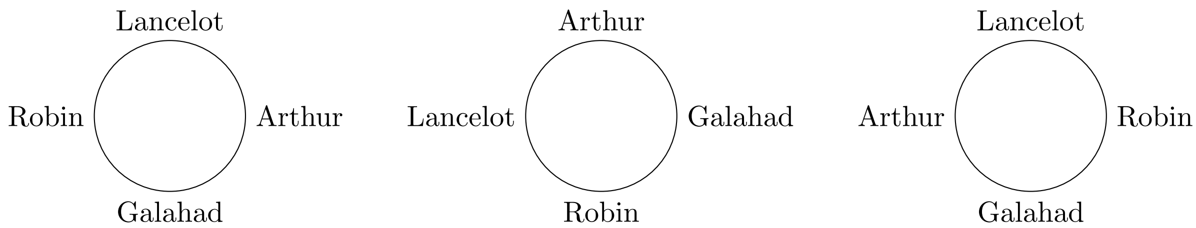
Now let's generalize the bijection rule. Suppose $f : T \to S$ is a total function that onto and k-to-1. (A function f is k-to-1 if for each $s \in S$, there are exactly k elements $t \in T$ such that $f(t) = s$). Then $|S| = |T|/k$.

Example 25.6: Consider a chess board. We want to place two pawns on this chess board so that they are in different rows and different columns. How many ways can we do this? One would be tempted to use the approach we took in Example 25.4 without change. Unfortunately, this does not work. Unlike different chess pieces, two pawns are indistinguishable. In other words, placing the first pawn on square a and the second pawn on square b yields the same configuration as placing the first pawn on square b and the second pawn on square a. But we only need a slight modification of the approach from Example 25.4 to get the right answer.

Let S be the set of all possible configurations we can obtain by placing two pawns on the chess board so that they are in different rows and different columns. Consider the set of strings $T = \{(r_1, c_1, r_2, c_2) \in \{1, \ldots, 8\}^4 \mid r_1 \neq r_2 \wedge c_1 \neq c_2\}$. This is an onto function that is 2-to-1 because (a, b, c, d) and (c, d, a, b) define the same configuration, and no other 4-tuple describes the same configuration as those two. Thus, $|S| = |T|/2$ by the generalized bijection rule.

Finally, we find $|T|$ the same way as in Example 25.4 and get $|S| = 8\cdot 8\cdot 7\cdot 7/2 = 1568$. ⊠

Example 25.7: At king Arthur's court, people are seated at a round table. Say there are n people seated at a table. Two seating arrangements are different if there is some person who has a different neighbor on his right (i.e., in the counterclockwise direction) in the two arrangements. For example, the arrangement in Figure 25.1b is the same as the arrangement in Figure 25.1a. On the other



(a) One seating arrangement.

(b) Same seating arrangement as the first one.

(c) This seating arrangement is different from the previous two. For example, the right neighbor of Galahad is Robin, whereas his right neighbor in the other two arrangements is Arthur.

Figure 25.1: Three ways of seating four people at a round table. The first two are equivalent seating arrangements, and the third one is different from the previous two.

hand, the assignment in Figure 25.1c is different from the other two seating assignments in Figure 25.1.

Let S be the set of all possible seating arrangements of the n people, and let T be the set of all permutations of those n people. Let $\pi \in T$, and fix one chair at the table. Define $f(\pi)$ to be the seating arrangement obtained by placing the first person in π on the fixed chair, the second person in π to the first person's right, and keep going around the table. For example, if we fix the leftmost seat (the one where Arthur is) in Figure 25.1a and take the permutation (Arthur, Lancelot, Robin, Galahad), we obtain the seating arrangement in Figure 25.1a.

We claim that the map is n-to-1. Let the people be p_1 through p_n, and assume without loss of generality that $\pi = (p_1, p_2, \ldots, p_n)$. Then person p_i has person p_{i+1} on his right for $i \in \{1, \ldots, n-1\}$, and person p_n has person p_1 on his right. Let's see how many other permutations yield this seating assignment. Once person p_1 is seated, the seats for the other people are determined by the rules for who the neighbor of p_i is. Since there are n locations where we can seat person p_1, there are n permutations of the people that produce the same seating arrangement as the permutation $(p_1, p_2, \ldots, p_n)$ (the other $n-1$ permutations are cyclic shifts of this permutation).

Also note that our function f is surjective because given an arrangement, we can just list an arbitrary person as the first element of the permutation and then go around the table in a counterclockwise direction and add people to subsequent terms in the permutation in the same order in which we see them when we go around the table.

Thus, f is an n-to-1 function from T to S. Since $T = n!$, the generalized bijection rule implies $|S| = |T|/n = n!/n = (n-1)!$. ⊠

Our last example finally solves the counting problem of Example 25.1.

Example 25.8: Consider a domain D with n elements. We would like to know how many s-element subsets D has.

For example, the 2-element subsets of the set $D = \{a, b, c, d\}$ are the six sets $\{a, b\}$, $\{a, c\}$, $\{a, d\}$, $\{b, c\}$, $\{b, d\}$, and $\{c, d\}$.

Let's reduce the problem to the problem of counting sequences of a certain type. Let S be the set of s-element subsets of D, and let T be the set of all permutations of the domain D. Define $f : T \to S$ by setting $f(\pi)$ to be the set of the first s terms in π. Since all terms in a permutation are different, $f(\pi)$ contains s elements.

This function is onto because to get π such that $f(\pi) = A$, we just list the s elements of A in an arbitrary order as the first s elements of π, and then list the remaining elements of the domain in the last $n-s$ positions in an arbitrary order. Permuting the first s elements of π doesn't change the function value, and neither does permuting the last $n-s$ elements. There are $s!$ ways to permute the first s elements, and $(n-s)!$ ways to permute the last $n-s$ elements. Furthermore, we can pick those two permutations independently, so the product rule applies, and we get that f is $s!(n-s)!$-to-1. Since there are $n!$ permutations of a set of n elements, we get

$$|S| = \frac{n!}{s!(n-s)!}. \tag{25.5}$$

$\boxtimes$

The result of Exercise 25.8 is also a very commonly used quantity, and has a name. We denote the right-hand side of (25.5) by $\binom{n}{s}$ and read it as "n choose s".

To wrap up today's lecture, let's return to the situation of Example 25.1. There, we wanted to count the number of strings of length 16 that contain exactly four ones. Note that the positions of ones in such a string in a one-to-one correspondence with the four-elements subsets of $\{1, \ldots, 16\}$. Thus, the number of such strings is $\binom{16}{4} = 1820$, and this is also the number of ways we can pick a dozen bagels that come in five different varieties.