14 Cardinality Rules

14.1 Counting One Thing by Counting Another

How do you count the number of people in a crowded room? You could count heads, since for each person there is exactly one head. Alternatively, you could count ears and divide by two. Of course, you might have to adjust the calculation if someone lost an ear in a pirate raid or someone was born with three ears. The point here is that you can often *count one thing by counting another*, though some fudging may be required. This is a central theme of counting, from the easiest problems to the hardest. In fact, we've already seen this technique used in Theorem 4.5.5, where the number of subsets of an n-element set was proved to be the same as the number of length-n bit-strings, by describing a bijection between the subsets and the bit-strings.

The most direct way to count one thing by counting another is to find a bijection between them, since if there is a bijection between two sets, then the sets have the same size. This important fact is commonly known as the *Bijection Rule*. We've already seen it as the Mapping Rules bijective case (4.7).

14.1.1 The Bijection Rule

The Bijection Rule acts as a magnifier of counting ability; if you figure out the size of one set, then you can immediately determine the sizes of many other sets via bijections. For example, let's look at the two sets mentioned at the beginning of Part III:

$$A = \text{all ways to select a dozen donuts when five varieties are available}$$
$$B = \text{all 16-bit sequences with exactly 4 ones}$$

An example of an element of set A is:

$$\underbrace{0\,0}_{\text{chocolate}} \quad \underbrace{}_{\text{lemon-filled}} \quad \underbrace{0\,0\,0\,0\,0\,0}_{\text{sugar}} \quad \underbrace{0\,0}_{\text{glazed}} \quad \underbrace{0\,0}_{\text{plain}}$$

Here, we've depicted each donut with a 0 and left a gap between the different varieties. Thus, the selection above contains two chocolate donuts, no lemon-filled, six sugar, two glazed, and two plain. Now let's put a 1 into each of the four gaps:

$$\underbrace{0\,0}_{\text{chocolate}} \quad 1 \quad \underbrace{}_{\text{lemon-filled}} \quad 1 \quad \underbrace{0\,0\,0\,0\,0\,0}_{\text{sugar}} \quad 1 \quad \underbrace{0\,0}_{\text{glazed}} \quad 1 \quad \underbrace{0\,0}_{\text{plain}}$$

and close up the gaps:

$$0011000000100100\,.$$

We've just formed a 16-bit number with exactly 4 ones—an element of B!

This example suggests a bijection from set A to set B: map a dozen donuts consisting of:

$$c \text{ chocolate}, l \text{ lemon-filled}, s \text{ sugar}, g \text{ glazed, and } p \text{ plain}$$

to the sequence:

$$\underbrace{0\dots0}_{c} \quad 1 \quad \underbrace{0\dots0}_{l} \quad 1 \quad \underbrace{0\dots0}_{s} \quad 1 \quad \underbrace{0\dots0}_{g} \quad 1 \quad \underbrace{0\dots0}_{p}$$

The resulting sequence always has 16 bits and exactly 4 ones, and thus is an element of B. Moreover, the mapping is a bijection: every such bit sequence comes from exactly one order of a dozen donuts. Therefore, $|A| = |B|$ by the Bijection Rule. More generally,

Lemma 14.1.1. *The number of ways to select n donuts when k flavors are available is the same as the number of binary sequences with exactly n zeroes and $k-1$ ones.*

This example demonstrates the power of the bijection rule. We managed to prove that two very different sets are actually the same size—even though we don't know exactly how big either one is. But as soon as we figure out the size of one set, we'll immediately know the size of the other.

This particular bijection might seem frighteningly ingenious if you've not seen it before. But you'll use essentially this same argument over and over, and soon you'll consider it routine.

14.2 Counting Sequences

The Bijection Rule lets us count one thing by counting another. This suggests a general strategy: get really good at counting just a few things, then use bijections to count everything else! This is the strategy we'll follow. In particular, we'll get really good at counting *sequences*. When we want to determine the size of some other set T, we'll find a bijection from T to a set of sequences S. Then we'll use our super-ninja sequence-counting skills to determine $|S|$, which immediately gives us $|T|$. We'll need to hone this idea somewhat as we go along, but that's pretty much it!

14.2.1 The Product Rule

The *Product Rule* gives the size of a product of sets. Recall that if $P_1, P_2, \ldots, P_n$ are sets, then

$$P_1 \times P_2 \times \cdots \times P_n$$

is the set of all sequences whose first term is drawn from P_1, second term is drawn from P_2 and so forth.

Rule 14.2.1 (Product Rule)**.** *If $P_1, P_2, \ldots P_n$ are finite sets, then:*

$$|P_1 \times P_2 \times \cdots \times P_n| = |P_1| \cdot |P_2| \cdots |P_n|$$

For example, suppose a *daily diet* consists of a breakfast selected from set B, a lunch from set L, and a dinner from set D where:

$$\begin{aligned} B &= \{\text{pancakes, bacon and eggs, bagel, Doritos}\} \\ L &= \{\text{burger and fries, garden salad, Doritos}\} \\ D &= \{\text{macaroni, pizza, frozen burrito, pasta, Doritos}\} \end{aligned}$$

Then $B \times L \times D$ is the set of all possible daily diets. Here are some sample elements:

$$\begin{gathered} (\text{pancakes, burger and fries, pizza}) \\ (\text{bacon and eggs, garden salad, pasta}) \\ (\text{Doritos, Doritos, frozen burrito}) \end{gathered}$$

The Product Rule tells us how many different daily diets are possible:

$$\begin{aligned} |B \times L \times D| &= |B| \cdot |L| \cdot |D| \\ &= 4 \cdot 3 \cdot 5 \\ &= 60. \end{aligned}$$

14.2.2 Subsets of an *n*-element Set

The fact that there are 2^n subsets of an n-element set was proved in Theorem 4.5.5 by setting up a bijection between the subsets and the length-n bit-strings. So the original problem about subsets was tranformed into a question about sequences—*exactly according to plan*! Now we can fill in the missing explanation of why there are 2^n length-n bit-strings: we can write the set of all n-bit sequences as a product of sets:

$$\{0,1\}^n ::= \underbrace{\{0,1\} \times \{0,1\} \times \cdots \times \{0,1\}}_{n \text{ terms}}.$$

Then Product Rule gives the answer:

$$|\{0,1\}^n| = |\{0,1\}|^n = 2^n.$$

14.2.3 The Sum Rule

Bart allocates his little sister Lisa a quota of 20 crabby days, 40 irritable days, and 60 generally surly days. On how many days can Lisa be out-of-sorts one way or another? Let set C be her crabby days, I be her irritable days, and S be the generally surly. In these terms, the answer to the question is $|C \cup I \cup S|$. Now assuming that she is permitted at most one bad quality each day, the size of this union of sets is given by the *Sum Rule*:

Rule 14.2.2 (Sum Rule)**.** *If $A_1, A_2, \ldots, A_n$ are* disjoint *sets, then:*

$$|A_1 \cup A_2 \cup \ldots \cup A_n| = |A_1| + |A_2| + \ldots + |A_n|$$

Thus, according to Bart's budget, Lisa can be out-of-sorts for:

$$\begin{aligned} |C \cup I \cup S| &= |C| + |I| + |S| \\ &= 20 + 40 + 60 \\ &= 120 \text{ days} \end{aligned}$$

Notice that the Sum Rule holds only for a union of *disjoint* sets. Finding the size of a union of overlapping sets is a more complicated problem that we'll take up in Section 14.9.

14.2.4 Counting Passwords

Few counting problems can be solved with a single rule. More often, a solution is a flurry of sums, products, bijections, and other methods.

For solving problems involving passwords, telephone numbers, and license plates, the sum and product rules are useful together. For example, on a certain computer system, a valid password is a sequence of between six and eight symbols. The first symbol must be a letter (which can be lowercase or uppercase), and the remaining symbols must be either letters or digits. How many different passwords are possible?

Let's define two sets, corresponding to valid symbols in the first and subsequent positions in the password.

$$\begin{aligned} F &= \{a, b, \ldots, z, A, B, \ldots, Z\} \\ S &= \{a, b, \ldots, z, A, B, \ldots, Z, 0, 1, \ldots, 9\} \end{aligned}$$

In these terms, the set of all possible passwords is:[1]

$$(F \times S^5) \cup (F \times S^6) \cup (F \times S^7)$$

[1]The notation S^5 means $S \times S \times S \times S \times S$.

Thus, the length-six passwords are in the set $F \times S^5$, the length-seven passwords are in $F \times S^6$, and the length-eight passwords are in $F \times S^7$. Since these sets are disjoint, we can apply the Sum Rule and count the total number of possible passwords as follows:

$$\begin{aligned}
&|(F \times S^5) \cup (F \times S^6) \cup (F \times S^7)| \\
&\quad = |F \times S^5| + |F \times S^6| + |F \times S^7| && \text{Sum Rule} \\
&\quad = |F| \cdot |S|^5 + |F| \cdot |S|^6 + |F| \cdot |S|^7 && \text{Product Rule} \\
&\quad = 52 \cdot 62^5 + 52 \cdot 62^6 + 52 \cdot 62^7 \\
&\quad \approx 1.8 \cdot 10^{14} \text{ different passwords.}
\end{aligned}$$