5.1.8

13

Suppose that you must choose a password at your work that is five to seven characters long. How many possible passwords are there if...

a) each password can be any combination of alphanumeric characters?

b) each password must contain at least one digit? (The remaining characters are still alphanumeric.)

a)

Since each character can be any alphanumeric character, we have 62 possibilities. (26 uppercase letters, 26 lowercase letters, and 10 digits.) Since we can choose the characters over and over and order is important in a password, we will use permutations with repetition or n^k. Again, n is 62 and k can range from 5 to 7. We can write this in a couple of different but equivalent ways:

$$62^5 + 62^6 + 62^7$$

is one such way. Or we could formulate this more succinctly as:

$$\sum_{k=5}^{7} 62^k$$

This one has the advantages of being compact and elegant and perhaps even better mathematics.

b)

A naive approach to this problem is to count all the problems with a single digit for certain and then let the rest of the digits be any alphanumeric character as stated in the parenthetical. This would be any of:

$$\begin{aligned} 10 \cdot 62^4 + 10 \cdot 62^5 + 10 \cdot 62^6 &= 10(62^4 + 62^5 + 62^6) \\ &= 10 \sum_{k=4}^{6} 62^k \end{aligned}$$

Again, the elegance of the summation is amazing.

But this is a naive approach and may not lead to the correct solution. We can also formulate this as mutually exclusive situations where we choose a single digit and no other, two digits and no more, three digits and no more, etc. for each of the possible password lengths. Then, for each password length k, we would have:

$$\sum_{i=1}^{k} 10^i 52^{k-i}$$

since there are i digits and that leaves just 52 characters for the remaining slots in the password.

But there are three possible k values and so we must add all of these together:

$$\sum_{k=5}^{7}\sum_{i=1}^{k} 10^i 52^{k-i}$$

However, the two approaches give different values so they can't be the same. Is either right? Both need a little help, it turns out. While we can factor the 10 out and group them all in the second version from commutativity, we still need to account for the original order they were in somehow because we are in a permutation zone. The easy way to do this is to choose that order by placing the i digits amongst the k slots like so:

$$\sum_{k=5}^{7}\sum_{i=1}^{k} 10^i 52^{k-i}\binom{k}{i}$$

This gives a counting that agrees with another approach I found in my long-ago notes:

$$\sum_{k=5}^{7}\left(62^k - 52^k\right)$$

This one works by counting all the possible passwords of each length like we did in part a) and then simply subtracting the passwords that are all letters. This isn't as extensible as the mutual exclusion version, but it does work for the at least one case.

Further, these two can be transformed into one another if we make a clever application of the binomial theorem. Note the inner sum's resemblance to this theorem. It looks almost just like $(10+52)^k$ expanded except it starts from 1 instead of 0 for the sum. If we then both add and subtract that term so we can have the sum come out as 62^k and just subtract the first term which would be $\binom{k}{0}10^0 52^{k-0} = 1\cdot 1\cdot 52^k$ or just 52^k. Neat, eh?

Oh, I think that went too fast... Let's walk it through step-by-step:

$$\begin{aligned}
\sum_{k=5}^{7}\sum_{i=1}^{k} 10^i 52^{k-i}\binom{k}{i} &= \sum_{k=5}^{7}\left(\sum_{i=1}^{k} 10^i 52^{k-i}\binom{k}{i} + \binom{k}{0}10^0 52^{k-0} - \binom{k}{0}10^0 52^{k-0}\right) \\
&= \sum_{k=5}^{7}\left(\sum_{i=0}^{k} 10^i 52^{k-i}\binom{k}{i} - \binom{k}{0}10^0 52^{k-0}\right) \\
&= \sum_{k=5}^{7}\left((10+52)^k - 52^k\right) \\
&= \sum_{k=5}^{7}(62^k - 52^k)
\end{aligned}$$

There we go! (Note the addition/subtraction of the 0th term had to be grouped inside the outer sum.)

What about our naive approach? Can it be fixed similarly? I tried to include a binomial factor to account for where the one chosen digit was located ($\binom{k+1}{1} = (k+1)$) — remember that the sum there was not over all the characters but just the ones that weren't required to be a digit, but it didn't help. The counts are still off by a good deal. Perhaps it was just too naive to be helped?