CSCI 104L Lecture 11: Introduction to Counting

Hashtables

Question 1. A company has assigned a unique 3-digit ID to each of its 1000 employees. We want to design a data structure so that you can input an employee ID and quickly bring up their employee record. How should we implement this?

Question 2. Does something similar work if we want to do this from USC student ID to student records? Why or why not?

Question 3. What if we want to store the English dictionary as a set of strings, so we can quickly look up if something is a word or not. Can we do something like this for non-integer keys?

A **hash function** takes a valid input (in the case of the last question, a word in the English language) and outputs the entry in the array to store it. To be a good hash function it must:

- be efficient to calculate
- distribute the inputs well
- be consistent

Question 4. Is $h(k) = 0$ a good hash function? Why or why not?

Question 5. Is $h(k) = k\%m$, where m is the size of the table, a good hash function? Why or why not?

Question 6. Is $h(k) =$ a random integer between 0 and $m-1$, where m is the size of the table, a good hash function? Why or why not?

The goal is to design a hash function where the probability of collision is $\leq \frac{1}{m}$. Any "good" hash function that satisfies this is called a "Universal Hash Function."

Counting

Question 7. A company has three new employees: Aaron, Rebecca, and Nicholas. There are 12 offices. How many different ways are there to assign offices to these employees?

The Product Rule: If a procedure can be broken down into a sequence of two tasks, and there are n_1 ways to do the first task and n_2 ways to do the second task, then there are $n_1 n_2$ ways to do the procedure.

Question 8. How many ways are there to arrange 10 people in a line?

Question 9. How many bit strings of length 8 either start with a 1 or end with two 0's?

The Subtraction Rule: If a task can be done in either n_1 ways or n_2 ways, and there is an overlap between these two methods of n_3 common ways, then the number of ways to do the task is $n_1 + n_2 - n_3$.

Question 10. The company 'Maddy Makes Games' has 350 job applicants. 220 of these applicants are computer science majors, 147 are business majors, and 51 are double majors in computer science and business. How many applicants majored in neither computer science nor business?

Question 11. How many different ways are there to seat 4 people around a circular table, where two seatings are considered identical if each person has the same left neighbor in both seatings, and the same right neighbor in both seatings?

The Division Rule: There are $\frac{n}{d}$ ways to do a task using a procedure which can be done in n different ways, and for any specific way w, it is identical to d of the n total ways.

Question 12. How many ways are there to arrange 10 people in a line, where two lineups are considered identical if they are mirror images of each other?

Question 13. In a version of BASIC, variables can be one or two alphanumeric characters (lower case or capitalized letters are not distinguished). The first character must be a letter. There are 5 two character strings which are reserved and cannot be used. How many different variable names are there?

Question 14. A computer system requires a password between 6 and 8 alphanumeric characters (lower case or capitalization is not distinguished). At least one character must be a digit. How many different passwords are there?

Question 15. How many bit strings of length 4 do not have two consecutive 1's?

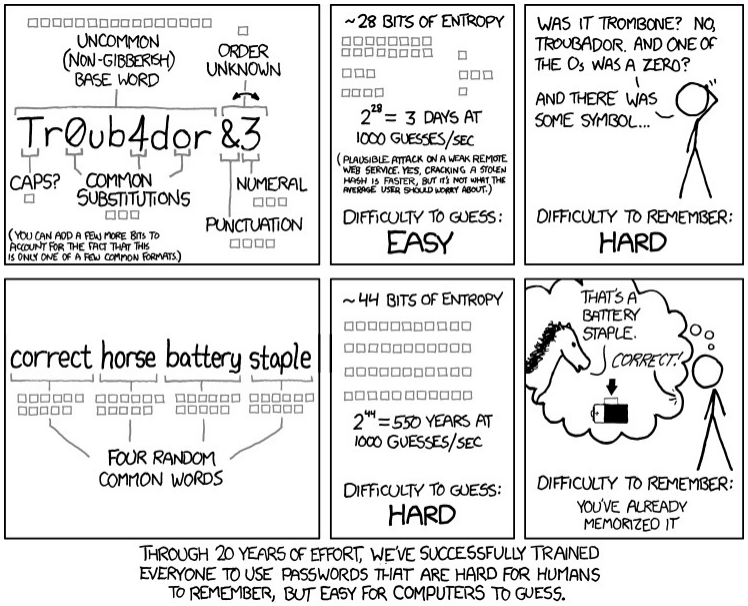

Figure 1: XKCD #936: Password Strength. To anyone who understands information theory and security and is in an infuriating argument with someone who does not (possibly involving mixed case), I sincerely apologize.