CS 70 Discrete Mathematics and Probability Theory
Fall 2021 **HW 6**

Due: Saturday 10/09, 4:00 PM
Grace period until Saturday 10/09, 5:59 PM

Sundry

Before you start writing your final homework submission, state briefly how you worked on it. Who else did you work with? List names and email addresses. (In case of homework party, you can just describe the group.)

1 Error-Correcting Codes

(a) Recall from class the error-correcting code for erasure errors, which protects against up to k lost packets by sending a total of $n+k$ packets (where n is the number of packets in the original message). Often the number of packets lost is not some fixed number k, but rather a *fraction* of the number of packets sent. Suppose we wish to protect against a fraction α of lost packets (where $0 < \alpha < 1$). At least how many packets do we need to send (as a function of n and α)?

(b) Repeat part (a) for the case of general errors.

2 Berlekamp-Welch Algorithm

In this question we will send the message $(m_0, m_1, m_2) = (1, 1, 4)$ of length $n = 3$. We will use an error-correcting code for $k = 1$ general error, doing arithmetic over GF(5).

(a) Construct a polynomial $P(x) \pmod 5$ of degree at most 2, so that

$$P(0) = 1, \qquad P(1) = 1, \qquad P(2) = 4.$$

What is the message $(c_0, c_1, c_2, c_3, c_4)$ that is sent?

(b) Suppose the message is corrupted by changing c_0 to 0. Set up the system of linear equations in the Berlekamp-Welch algorithm to find $Q(x)$ and $E(x)$.

(c) Assume that after solving the equations in part (b) we get $Q(x) = 4x^3 + x^2 + x$ and $E(x) = x$. Show how to recover the original message from Q and E.

3 Error-Detecting Codes

Suppose Alice wants to transmit a message of n symbols, so that Bob is able to *detect* rather than *correct* any errors that have occurred on the way. That is, Alice wants to find an encoding so that Bob, upon receiving the code, is able to either

(I) tell that there are no errors and decode the message, or

(II) realize that the transmitted code contains at least one error, and throw away the message.

Assuming that we are guaranteed a maximum of k errors, how should Alice extend her message (i.e. by how many symbols should she extend the message, and how should she choose these symbols)? You may assume that we work in $\mathrm{GF}(p)$ for very large prime p. Show that your scheme works, and that adding any lesser number of symbols is not good enough.

4 Counting, Counting, and More Counting

The only way to learn counting is to practice, practice, practice, so here is your chance to do so. Although there are many subparts, each subpart is fairly short, so this problem should not take any longer than a normal CS70 homework problem. You do not need to show work, and **Leave your answers as an expression** (rather than trying to evaluate it to get a specific number).

(a) How many ways are there to arrange n 1s and k 0s into a sequence?

(b) How many 7-digit ternary (0,1,2) bitstrings are there such that no two adjacent digits are equal?

(c) A bridge hand is obtained by selecting 13 cards from a standard 52-card deck. The order of the cards in a bridge hand is irrelevant.
i. How many different 13-card bridge hands are there? ii. How many different 13-card bridge hands are there that contain no aces? iii. How many different 13-card bridge hands are there that contain all four aces? iv. How many different 13-card bridge hands are there that contain exactly 6 spades?

(d) Two identical decks of 52 cards are mixed together, yielding a stack of 104 cards. How many different ways are there to order this stack of 104 cards?

(e) How many 99-bit strings are there that contain more ones than zeros?

(f) An anagram of ALABAMA is any re-ordering of the letters of ALABAMA, i.e., any string made up of the letters A, L, A, B, A, M, and A, in any order. The anagram does not have to be an English word.
i. How many different anagrams of ALABAMA are there? ii. How many different anagrams of MONTANA are there?

(g) How many different anagrams of ABCDEF are there if: (1) C is the left neighbor of E; (2) C is on the left of E (and not necessarily E's neighbor)

(h) We have 9 balls, numbered 1 through 9, and 27 bins. How many different ways are there to distribute these 9 balls among the 27 bins? Assume the bins are distinguishable (e.g., numbered 1 through 27).

(i) How many different ways are there to throw 9 identical balls into 27 bins? Assume the bins are distinguishable (e.g., numbered 1 through 27).

(j) We throw 9 identical balls into 7 bins. How many different ways are there to distribute these 9 balls among the 7 bins such that no bin is empty? Assume the bins are distinguishable (e.g., numbered 1 through 7).

(k) There are exactly 20 students currently enrolled in a class. How many different ways are there to pair up the 20 students, so that each student is paired with one other student? Solve this in at least 2 different ways. **Your final answer must consist of two different expressions.**

(l) How many solutions does $x_0 + x_1 + \cdots + x_k = n$ have, if each x must be a non-negative integer?

(m) How many solutions does $x_0 + x_1 = n$ have, if each x must be a *strictly positive* integer?

(n) How many solutions does $x_0 + x_1 + \cdots + x_k = n$ have, if each x must be a *strictly positive* integer?

5 Shipping Crates

A widget factory has four loading docks for storing crates of ready-to-ship widgets. Suppose the factory produces 8 indistinguishable crates of widgets and sends each crate to one of the four loading docks.

(a) How many ways are there to distribute the crates among the loading docks?

(b) Now, assume that any time a loading dock contains at least 5 crates, a truck picks up 5 crates from that dock and ships them away. (e.g., if 6 crates are sent to a loading dock, the truck removes 5, leaving 1 leftover crate still in the dock). We will now consider two configurations to be identical if, for every loading dock, the two configurations have the same number of leftover crates in that dock. How would your answer in the previous part compare to the number of outcomes given the new setup? Do not compute the actual value in this part, we will doing that in parts (c) - (e). We are looking for a qualitative answer (greater than part (a), equal to part (a), less than or equal to part (a), etc.) Justify your answer.

(c) We will now attempt to count the number of configurations of crates. First, we look at the case where crates are removed from the dock. How many ways are there to distribute the crates such that some crate gets removed from the dock?

(d) How many ways are there to distribute the crates such that no crates are removed from the dock; i.e. no dock receives at least 5 crates?

(e) Putting it together now, what are the total number of possible configurations for crates in the modified scenario? *Hint:* Observe that, regardless of which dock receives the 5 crates, we end up in the same situation. After all the shipping has been done, how many possible configurations of leftover crates in loading docks are there?

6 Grids and Trees!

Suppose we are given an $n \times n$ grid, for $n \geq 1$, where one starts at $(0,0)$ and goes to (n,n). On this grid, we are only allowed to move left, right, up, or down by increments of 1.

(a) How many shortest paths are there that go from $(0,0)$ to (n,n)?

(b) How many shortest paths are there that go from $(0,0)$ to $(n-1,n+1)$?

Now, consider shortest paths that meet the conditions where we can only visit points (x,y) where $y \leq x$. That is, the path cannot cross line $y = x$. We call these paths n-legal paths for a maze of side length n. Let F_n be the number of n-legal paths.

(c) Compute the number of shortest paths from $(0,0)$ to (n,n) that cross $y = x$. (Hint: Let (i,i) be the first time the shortest path crosses the line $y = x$. Then the remaining path starts from $(i,i+1)$ and continues to (n,n). If in the remainder of the path one exchanges y-direction moves with x-direction moves and vice versa, where does one end up?)

(d) Compute the number of shortest paths from $(0,0)$ to (n,n) that do not cross $y = x$. (You may find your answers from parts (a) and (c) useful.)

(e) A different idea is to derive a recursive formula for the number of paths. Fix some i with $0 \leq i \leq n-1$. We wish to count the number of n-legal paths where the last time the path touches the line $y = x$ is the point (i,i). Show that the number of such paths is $F_i \cdot F_{n-i-1}$. (Hint: If $i = 0$, what are your first and last moves, and where is the remainder of the path allowed to go?)

(f) Explain why $F_n = \sum_{i=0}^{n-1} F_i \cdot F_{n-i-1}$.

(g) Create and explain a recursive formula for the number of trees with n vertices $(n \geq 1)$, where each non-root node has degree at most 3, and the root node has degree at most 2. Two trees are different if and only if either left-subtree is different or right-subtree is different.

(Notice something about your formula and the grid problem. Neat!)

7 Fermat's Wristband

Let p be a prime number and let k be a positive integer. We have beads of k different colors, where any two beads of the same color are indistinguishable.

(a) We place p beads onto a string. How many different ways are there to construct such a sequence of p beads with up to k different colors?

(b) How many sequences of p beads on the string are there that use at least two colors?

(c) Now we tie the two ends of the string together, forming a wristband. Two wristbands are equivalent if the sequence of colors on one can be obtained by rotating the beads on the other. (For instance, if we have $k = 3$ colors, red (R), green (G), and blue (B), then the length $p = 5$ necklaces RGGBG, GGBGR, GBGRG, BGRGG, and GRGGB are all equivalent, because these are all rotated versions of each other.)

How many non-equivalent wristbands are there now? Again, the p beads must not all have the same color. (Your answer should be a simple function of k and p.)

[*Hint*: Think about the fact that rotating all the beads on the wristband to another position produces an identical wristband.]

(d) Use your answer to part (c) to prove Fermat's little theorem.