Homework 7

Out: 18 Oct. Due: 25 Oct.

Instructions: *Start each problem on a new sheet. Write your name, SID number, section number and "CS70" on every sheet. Clip together (do not staple!) the sheets for different problems (but if you use more than one sheet for one problem, staple those). Put your solutions in the homework box on Soda level 2 by 5pm on Thursday. You are encouraged to form small groups (two to four people) to work through the homework, but you* **must** *write up all your solutions on your own.*

1. How to Steal the TA's Car *(Carried over from HW6)*

One of the CS70 TAs lives in a builiding with an electronic garage door that is opened by a remote control. One day, you overhear his room-mate talking about how, after losing his remote control and having the landlord demand $60 to replace it, he discovered that the remote controls can be bought online for $12, and all you need to do after buying one is to open it up and set ten tiny on-off switches to match the "code" of the door. These settings, of course, can be obtained by looking at the switches in another remove for the same door.

After a particularly nasty homework, you decide to take revenge by getting one of these remotes and breaking into the garage. Of course, unlike the room-mate, you don't have access to another remote from which to copy the code. It takes 5 seconds to flip a single switch in either direction (they're tiny, and CS students aren't known for fine motor control), and 1 second to test whether the current switch settings work by just pressing the button. What is the shortest amount of time that you need to *definitely* open the garage door? Explain precisely how you would go about achieving this minimum time. Keep in mind that, per Murphy's law, even if you try all but one of the combinations, it is possible that the right combination is the one you haven't tried.

2. Routing on the hypercube

Recall that the n-dimensional hypercube contains 2^n vertices, each labeled with a distinct n-bit string; two vertices are adjacent iff their bit strings differ in exactly one position.

(a) The hypercube is a popular architecture for parallel computation. Let each vertex of the hypercube represent a processor and each edge represent a communication link. Suppose we want to send a packet from vertex x to vertex y. Consider the following "bit-fixing" algorithm:

> In each step, the current processor compares its address to the destination address of the packet. Let's say that the two addresses match up to the first k positions. The processor then forwards the packet and the destination address on to its neighboring processor whose address matches the destination address in at least the first $k+1$ positions. This process continues until the packet arrives at its destination.

Consider an example in which $n = 4$ and the packet is to be sent from vertex $x = 1001$ to vertex $y = 0100$. Write down the sequence of processors that the packet is forwarded to using the bit-fixing algorithm.

(b) For any two n-bit strings x and y, define the *Hamming distance* $H(x, y)$ between x and y to be the number of bit positions in which they differ. In terms of $H(x, y)$, what is the number of edges traversed by a packet moving from x to y under the bit-fixing algorithm?

(c) What is the length of a shortest possible path from x to y in the hypercube? How many such paths are there?

(d) For a fixed pair of vertices x and y in the n-dimensional hypercube, consider the *subgraph* $G_{xy} = (V_{xy}, E_{xy})$, where V_{xy} and E_{xy} consist of all vertices and edges, respectively, that lie on shortest paths between x and y. In terms of $H(x, y)$, what are the sizes of V_{xy} and E_{xy}? Can you describe the graph G_{xy}?

3. **Counting cards**

Consider a standard deck of 52 cards, consisting of four suits each containing cards labeled $A, 2, \ldots, 10, J, Q, K$. You receive a hand of four cards. Answer each of the following questions, being careful to *show your working* in each case.

(a) How many possible 4-card hands are there? [Note: the order of the cards is not important.]

(b) How many 4-card hands consist only of "face cards" (i.e., A, J, Q, K)?

(c) How many 4-card hands contain at least one King?

(d) How many 4-card hands contain two cards of one suit and two of a different suit?

(e) How many 4-card hands contain a contiguous sequence of four values (e.g., $4, 5, 6, 7$ or $A, 2, 3, 4$), regardless of the suits? [Note: Ace may count high or low, so we also include the sequence J, Q, K, A.]

4. **More counting**

Consider again a standard deck of 52 cards. This time we will do some counting related to the entire deck of cards, rather than to a hand of cards as in Q4. Answer the following questions, again being careful to show your working.

(a) How many orderings (i.e., permutations) of the deck are there?

(b) How many permutations have an Ace as the first card?

(c) How many permutations have an Ace in the first five cards?

(d) How many permutations have the King and Queen of Hearts next to each other (in either order)?

(e) How many permutations have all the Diamonds in the first half of the deck?

5. **Algebraic vs. combinatorial proofs**

Consider the following identity:

$$\binom{2n}{2} = 2\binom{n}{2} + n^2.$$

(a) Prove the identity by algebraic manipulation (using the formula for the binomial coefficients).

(b) Prove the identity using a combinatorial argument.