CS 70 Discrete Mathematics and Probability Theory
Summer 2016 Dinh, Psomas, and Ye HW 2

Due Tuesday July 5 at 1:59PM

1. (8 points: 3/5) **Hit or miss**

   For each of the claims and proofs below, state whether the claim is true or not and whether the proof is correct or not. For each incorrect proof, point out what is wrong with the proof. Simply saying that the claim or the induction hypothesis is false is *not* a valid explanation of what is wrong with the proof.

   (a) **Claim:** For all nonnegative integers n, $2n = 0$.

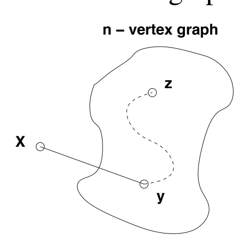
   Proof. We will prove by strong induction on n.
   Base Case: $2 \times 0 = 0$. It is true for $n = 0$.
   Inductive Hypothesis: Assume that $2k = 0$ for all $0 \le k \le n$.
   Inductive Step: We must show that $2(n+1) = 0$. Write $n+1 = a+b$ where $0 < a, b \le n$. From the inductive hypothesis, we know $2a = 0$ and $2b = 0$, therefore,

   $$2(n+1) = 2(a+b) = 2a+2b = 0+0 = 0.$$

   So the statement is true. $\square$

   (b) **Claim:** If every vertex in an undirected graph has degree at least 1, then the graph is connected.

   Proof. We use induction on the number of vertices $n \ge 1$.
   Base Case: There is only one graph with a single vertex and it has degree 0. Therefore, the base case is vacuously true, since the if-part is false.
   Inductive Hypothesis: Assume the claim is true for some $n \ge 1$.
   Inductive Step: We prove the claim is also true for $n+1$. Consider an undirected graph on n vertices in which every vertex has degree at least 1. By the inductive hypothesis, this graph is connected. Now add one more vertex x to obtain a graph on $(n+1)$ vertices, as shown below.



   All that remains is to check that there is a path from x to every other vertex z. Since x has degree at least 1, there is an edge from x to some other vertex; call it y. Thus, we can obtain a path from x to z by adjoining the edge $\{x, y\}$ to the path from y to z. This proves the claim for $n+1$. $\square$

2. (14 points: 3/3/4/4) **Trees**

   Recall that a **tree** is a connected acyclic graph (graph without cycles). In the note, we presented a few other definitions of a tree, and in this problem, we will prove two fundamental properties of a tree, and derive two definitions of a tree we learn from lecture note based on these properties. Let's start with the properties:

   (a) Prove that any pair of vertices in a tree are connected by exactly one (simple) path.

   (b) Prove that adding any edge to a tree creates a simple cycle.

   Now you will show that if a graph satisfies either of these two properties then it must be a tree:

   (c) Prove that if every pair of vertices in a graph are connected by exactly one simple path, then the graph must be a tree.

   (d) Prove that if the graph has no simple cycles and has the property that the addition of any single edge (not already in the graph) will create a simple cycle, then the graph is a tree.

3. (12 points: 6/6) **Networks and Tours**

   Please prove or disprove the following claims.

   (a) Suppose we have n websites such that for every pair of websites A and B, either A has a link to B or B has a link to A. Prove or disprove that there exists a website that is reachable from every other website by clicking at most 2 links. (*Hint: Induction*)

   (b) We have shown in the lecture (or you have read Lecture Note 4) that a connected undirected graph has an Eulerian tour if and only if every vertex has even degree.

   Prove or disprove that if a connected graph G on n vertices has exactly $2d$ vertices of odd degree, then there are d walks that *together* cover all the edges of G (i.e., each edge of G occurs in exactly one of the d walks; and each of the walks should not contain any particular edge more than once).

4. (12 points: 1.5/2/3.5/5) **Hypercube routing**

   Recall that an n-dimensional hypercube contains 2^n vertices, each labeled with a distinct n bit string, and two vertices are adjacent if and only if 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 for 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 the following example where $n = 4$: Suppose that the source vertex is (1001) and the destination vertex is (0100). Give the sequence of processors that the packet is forwarded to using the bit-fixing algorithm.

(b) The *Hamming distance* $H(x,y)$ between two n-bit strings x and y is the number of bit positions where they differ. Show that for an arbitrary source vertex and arbitrary destination vertex, the number of edges that the packet must traverse under this algorithm is the Hamming distance between the n-bit strings labeling source and destination vertices.

(c) Consider the following example where $n = 3$: Suppose that x is (110) and y is (011). What is the length of the shortest path between x and y? What is the set of all vertices and the set of all edges that lie on shortest paths between x and y? Do you see a pattern? You do not need to prove your answer here – you'll provide a general proof in part (d).

(d) Answer the last question for an arbitrary pair of vertices x and y in the hypercube. Can you describe the set of vertices and the set of edges that lie on shortest paths between x and y? Prove that your answers are correct. (*Hint:* consider the bits where x and y differ.)

5. (7 points: 1/2/2/2) **Bipartite graphs**

An undirected graph is called *bipartite* if its vertices can be partitioned into two disjoint sets L, R such that each edge connects a vertex in L with a vertex in R (i.e., there is no edge connecting two vertices in L or two vertices in R).

(a) Prove that a bipartite graph has no cycles of odd length.

(b) Prove that $\sum_{v \in L} \text{degree}(v) = \sum_{v \in R} \text{degree}(v)$.

(c) Let s denote the average degree of vertices in L and t the average degree of vertices in R, i.e., $s = \frac{1}{|L|} \sum_{v \in L} \text{degree}(v)$ and $t = \frac{1}{|R|} \sum_{v \in R} \text{degree}(v)$. Prove that $s/t = |R|/|L|$.

(d) In 1992, the University of Chicago interviewed a random sample of 2500 people in the U.S. about the number of opposite-gender sex partners they had had. They reported that on average men have 74% more opposite-gender partners than women. At around the same time, the U.S. Census Bureau reported that the female population of the U.S. was about 140 million and the male population was about 134 million. With reference to part c, explain why the University of Chicago and the U.S. Census Bureau can't both be right.

6. (12 points: 2/2/3/5) **TA Assignment**

You have been asked to assign TAs for the summer sessions. Each class has its own method for ranking candidates, and each candidate has their own preferences. An assignment is **unstable** if a class and a candidate prefer each other to their current assignments. Otherwise, it is **stable**.

Candidate information:

Candidate	CS61C Grade	CS70 Grade	CS61A Grade	Teaching Experience	Overall GPA	Preferences
A	A+	A	A	Yes	3.80	CS61C > CS70 > CS61A
B	A	A	A	No	3.61	CS61C > CS61A > CS70
C	A	A+	A-	Yes	3.60	CS61C > CS70 > CS61A

Ranking method:

- CS61C: Rank by CS61C grade. Break ties using teaching experience, then overall GPA.
- CS70: Rank by teaching experience. Break ties using CS70 grade, then overall GPA.
- CS61A: Rank by CS61A grade. Break ties using overall GPA, then teaching experience.

(a) Find a stable assignment.

(b) Can you find another, or is there only one stable assignment (if there is only one, why)?

(c) CS61C is overenrolled and needs two TAs. There is another candidate.

Candidate	CS61C Grade	CS70 Grade	CS61A Grade	Teaching Experience	Overall GPA	Preference
D	A+	A	A+	No	3.90	CS70 > CS61A > CS61C

Find a stable assignment.

(d) Prove your assignment in Part (c) is stable.

7. (8 points: 3/5) **Long Courtship**

(a) Run the traditional propose-and-reject algorithm on the following example:

Man	Preference List
1	$A > B > C > D$
2	$B > C > A > D$
3	$C > A > B > D$
4	$A > B > C > D$

Woman	Preference List
A	$2 > 3 > 4 > 1$
B	$3 > 4 > 1 > 2$
C	$4 > 1 > 2 > 3$
D	$1 > 2 > 3 > 4$

(b) We know from the notes that the propose-and-reject algorithm must terminate after at most n^2 proposals. Prove a sharper bound showing that the algorithm must terminate after at most $n(n-1)+1$ proposals. Is this instance a worst-case instance for $n = 4$? How many days does the algorithm take on this instance?

8. (16 points: 8/8) **Better Off Alone**

In the stable marriage problem, suppose that some men and women have standards and would not just settle for anyone. In other words, in addition to the preference orderings they have, they prefer being alone to being with some of the lower-ranked individuals (in their own preference list). A pairing could ultimately have to be partial, i.e., some individuals would remain single.

The notion of stability here should be adjusted a little bit. A pairing is stable if

- there is no paired individual who prefers being single over being with his/her current partner,
- there is no paired man and paired woman that would both prefer to be with each other over their current partners, and
- there is no single man and single woman that would both prefer to be with each other over being single.
- there is no paired man and single woman (or single man and paired woman) that would both prefer to be with each other over the current choice (the current partner or being alone).

(a) Prove that a stable pairing still exists in the case where we allow single individuals. You can approach this by introducing imaginary mates that people "marry" if they are single. How should you adjust the preference lists of people, including those of the newly introduced imaginary ones for this to work?

(b) As you saw in the lecture, we may have different stable pairings. But interestingly, if a person remains single in one stable pairing, s/he must remain single in any other stable pairing as well (there really is no hope for some people!). Prove this fact by contradiction.

9. (14 points: 7/7) **Karl and Emma fight!**

(a) Karl and Emma are having a disagreement regarding the traditional propose-and-reject algorithm. They both agree that it favors men over women. But they disagree about what, if anything, can be done without changing the ritual form of men proposing, women rejecting, and people getting married when there are no more rejections.

Karl mansplains: "It's hopeless. Men are obviously going to propose in the order of their preferences. It's male optimal so why would they do anything else? As far as the women are concerned, given that they face a specific choice of proposals at any given time, they are obviously going to select the suitor they like the most. So unless we smash the system entirely, it is going to keep all women down."

Emma says: "People are more perceptive and forward-looking that you think. Women talk to each other and know each other's preferences regarding men. They can also figure out the preferences of the men they might be interested in. A smart and confident woman should be able to do better for herself in the long run by not trying to cling to the best man she can get at the moment. By rejecting more strategically, she can simultaneously help out both herself and her friends."

Is Emma ever right? If it is impossible, prove it. If it is possible, construct and analyze an example (a complete set of people and their preference lists) in which a particular woman acting on her own (by not following the ordering of her preference list when deciding whether to accept or reject among multiple proposals) can get a better match for herself without hurting any other woman. Show how she can do so. The resulting pairing should also be stable.

(b) Karl and Emma have another disagreement! Karl claims that if a central authority was running the propose-and-reject algorithm then cheating the system might improve the cheater's chances of getting the more desirable candidate. The cheater need not care about what happens to the others.

Karl says: "Lets say there exists a true preference list. A prefers 1 to 2 but both are low on her preference list. By switching the reported preference order among 1 and 2, she can end up with 3 whom she prefers over 1 and 2 which wasn't possible if she did not lie. Isn't that cool?"

Emma responds: "That's impossible! In the traditional propose-and-reject algorithm switching the preference order 1 and 2 cannot improve A's chance to end up with 3."

Either prove that Emma is right or give an example of set of preference list for which a switch would improve A's husband (that is, she gets matched with 3), and hence proving Karl is right.