Homework 6 **Graph Algorithms**

Due: Friday, May 26, 11:45pm

Please answer the following questions, each of which is worth 10 points.

1. (CS 163 students only) Use the Gale-Shapley algorithm (with men proposing to women) to find a stable matching for the following set of four men and four women, given their preference lists (from highest to lowest preference).

Man	**Preference list**
Adam	Eve, Greta, Hannah, Faith
Ben	Eve, Faith, Greta, Hannah
Caleb	Greta, Eve, Faith, Hannah
David	Faith, Eve, Hannah, Greta

Woman	**Preference list**
Eve	Ben, Caleb, David, Adam
Faith	Adam, Ben, Caleb, David
Greta	Caleb, David, Adam, Ben
Hannah	Ben, David, Adam, Caleb

2. Let G be a complete bipartite graph such that $|X| = |Y| = n$ and for each pair of vertices $x \in X$ and $y \in Y$, there is an edge joining x and y. Show that G has $n!$ distinct maximum (perfect) matchings.

3. Consider a particular solution for the Stable Marriage Problem with n men and n women.

   (a) Suppose there is at least one man did not get his top choice and further suppose that this man has changed his mind: he now finds his wife to be more compatible than expected. In other words, suppose this man changes his preference list by moving his wife higher on the list. Does this change the stability of the marriages? Either prove the marriages remain stable or give an example to show that they don't.

   (b) What if instead, there is some man who moves his wife lower on his preference list. Does this change the stability of the marriages? Either prove the marriages remain stable or give an example to show that they don't.

4. For each of the following claims, state whether the claim is true or false, and give a short proof to justify your answer (i.e., either a brief argument why it is true or a counter-example showing that it is false). Note: By a stable marriage instance, we mean an input for stable marriage, i.e., the sets of men and women and their preference lists.

(a) In a stable marriage instance, if man M and woman W each put each other at the top of their respective preference lists, then the Gale-Shapely algorithm is guaranteed to match M with W.

(b) In a stable marriage instance with at least two men and two women, if man M and woman W each put each other at the bottom of their respective preference lists, then the Gale-Shapely algorithm will never match M with W.

5. In the stable marriage algorithm described in class, the men propose to the women in preference order, and each woman accepts the best proposal she has seen so far. If there are n men and n women, how many women (as a function of n) can be forced to settle for their last choice? Describe a scenario that would cause this many women to be matched with last-choice men: what preference orderings cause this behavior, and what does the algorithm do when given this input?

6. (CS 265 students only) In the *course enrollment* problem, we are given n students and m discussion sections. Each discussion section, u, has some number, q_u, of seats, and we assume that the total number of students is larger than the total number of seats (i.e., $\sum_{u=1}^{m} q_u < n$). Each student ranks the m discussion sections in order of preference, and the instructor for each discussion ranks the n students. Our goal is to find an assignment of students to seats (one student per seat) that is stable in the following sense:

   - There is no student-section pair, (s, u), such that s prefers u to her allocated discussion section and the instructor for u prefers s to one of the students assigned to u.
   - There is no discussion section u for which the instructor prefers some unassigned student s to one of the students assigned to u.

   Note that this problem is almost the same as the Stable Marriage Problem, with two differences: (i) there are more students than seats; and (ii) each discussion section generally has more than one seat. Explain how to modify the propose-and-reject Gale-Shapely algorithm so that it finds a stable assignment of students to seats. Show that your algorithm is correct and analyze its running time.