Homework 1

1. Consider the G-S algorithm for n men and n women. What is the maximum number of times a man may be rejected as a function of n? Give an example where this happens. (5pts)

   The maximum number of times a man may be rejected is $n - 1$, since no man will be rejected by all n women.

   Rubric (5pt):

   - 2 pt: Correctly state n - 1
   - 3 pt: Provides a correct explanation.

2. Determine whether the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample. (5pts)

   For all $n \geq 2$, there exists a set of preferences for n men and n women such that in the stable matching returned by the G-S algorithm when men are proposing, every man is matched with their most preferred woman.

   True: This happens so long as the most preferred woman for each man is different.

   Rubric (5pts):

   - 2 pts: Correctly identify the statement as true.
   - 3 pts: Provide a correct explanation or example.

3. Solve Kleinberg and Tardos, Chapter 1, Exercise 3. (5pt)

   We will give an example (with n = 2) of a set of TV shows/associated ratings for which no stable pair of schedules exists. Let a1, a2 be set the shows of A and let b1; b2 be the set of shows of B. Let the ratings of a1; a2; b1; b2 be 1, 3, 2 and 4 respectively. In every schedule that A and B can choose, either a1 shares a slot with b1 or a1 shares a slot with b2. If a1 shares a slot with b1, then A has an incentive to make a1 share a slot with b2 thereby increasing its number of winning slots from 0 to 1. If a1 shares a slot with b2, then B has an incentive to make b2 share a slot with a2 thereby increasing its number of winning slots from 1 to 2. Thus every schedule that A and B can choose is unstable.

   Rubric (5pt):

   - 1 pt: Correct claim that there are configurations of shows without stable matching
   - 4 pt: Provides a correct counterexample

4. Solve Kleinberg and Tardos, Chapter 1, Exercise 4. (15pts)

   We will use a variation of the G-S algorithm, then show that the solution returned by this algorithm is a stable matching. In the following algorithm, we use hospitals in the place of men; and students in the place of women, with respect to the earlier version of the G-S algorithm given in Chapter 1. This algorithm terminates in $O(mn)$ steps because each hospital offers a position to a student at most once, and in each iteration some hospital offers a position to some student.

   The algorithm terminates by producing a matching M for any given preference list. Suppose there are $p > 0$ positions available at hospital h. The algorithm terminates with all of the positions filled, since, any hospital that did not fill all of its positions must have offered them to every student. Every student who rejected must be committed to some other hospital. Thus, if h still has available positions, it would mean total number of available positions is strictly greater than n, the number of students. This contradicts the assumption given, proving that all the positions get filled.

```
1: while there exists a hospital h that has available positions do
2:    Offer position to the next highest ranked student s in the preference list
      of h that wasn't offered by h before
3:    if s has not already accepted a position at another hospital h' then
4:       s accepts the position at h.
5:    else
6:       Let s be matched with h'.
7:       if s prefers h' to h then
8:          then s rejects the offer of h
9:       else
10:         s accepts the position at h freeing up a position at h'
11:      end if
12:   end if
13: end while
```

The assignment is stable. Suppose that the matching M produced by our adapted G-S algorithm contains one or more instabilities. If the instability was of the first type (a student $s^{'}$ was preferred over a student s by a hospital h but was not admitted), then h must have made an offer to s before $s^{'}$ who wasn't offered, which is a contradiction because h prefers $s^{'}$ to s. Thus, the instability was not of the first type.If the instability was of the second type (there are student s and $s^{'}$ currently at hospitals h and $h^{'}$ respectively, and there's a swap mutually beneficial to h and $s^{'}$), then h must not have admitted $s^{'}$ when it considered it before s, which implies that $s^{'}$ prefers $h^{'}$ to h, a contradiction. Thus, the instability was not of the second type. Thus, the matching was stable. Thus at least one stable matching always exists (and it is produced by the adapted G-S algorithm as above).

Rubric (15pts):

- 8pts: Algorithm
  - 1pt: Loop condition (line 1)
  - 2 pts: hospitals offer the next highest-ranked student (line 2)
  - 2pts: case that s is free (lines 3-4)
  - 3pts: cases if s is at another h'
- 7pts: Proof
  - 1 pt: Algorithm terminates in finite steps (optional to mention in O(mn) steps)
  - 2pts: All positions get filled
  - 2pts: Explain why no instability of the first type
  - 2pts: Explain why no instability of the second type

5. Consider a stable marriage problem where the set of men is given by $M = m_1, m_2, ..., m_N$ and the set of women is $W = w_1, w_2, ..., w_N$. Consider their preference lists to have the following properties:

$$\forall w_i \in W : w_i \text{ prefers } m_i \text{ over } m_j \qquad \forall j > i$$

$$\forall m_i \in M : m_i \text{ prefers } w_i \text{ over } w_j \qquad \forall j > i$$

Prove that a unique stable matching exists for this problem. Note: the $\forall$ symbol means "for all". (10pts)

We will prove that matching S where m_i is matched to w_i for all $i = 1, 2, \cdots, N$ is the unique stable matching in this case. We will use the notation $S(a) = b$ to denote that in matching S, a is matched to b. It is evident that S is a matching because every man is paired with exactly one woman and vice versa. If any man m_j prefers

w_k to w_j where $k < j$, then such a higher ranked woman w_k prefers her current partner to m_j. Thus, there are no instabilities and S is a stable matching.

Now let's prove that this stable matching is unique. By way of contradiction, let's assume that another stable matching S', which is different from S, exists. Therefore, there must exist some i for which $S'(w_i) = m_k$, $k \neq i$. Let x be the minimum value of such an i. Similarly, there must exist some j for which $S'(m_j) = w_l$, $j \neq l$. Let y be the minimum value of such a j. Since $S'(w_i) = m_i$ for all $i < x$, and $S'(m_j) = w_j$ for all $j < y$, $x = y$. $S'(w_x) = m_k$ implies $x < k$. Similarly, $S'(m_y) = w_l$ implies that $y = x < l$. Given the preference lists, $m_y = m_x$ prefers w_x to w_l, and w_x prefers m_x to m_k. This is an instability and hence, S' cannot be a stable matching.

Rubric (10pts):

- Proof that stable matching exists (4pts)
- Explain that this stable matching is unique (6pts)

6. State True/False: An instance of the stable marriage problem has a unique stable matching if and only if the version of the Gale-Shapely algorithm where the male proposes and the version where the female proposes both yield the exact same matching. (10pts)

True.

Proving the given statement requires proving the claims in two directions ('if' and 'only if').

Claim: *If there is a unique stable matching then the version of the GaleShapely algorithm where the male proposes and the version where the female proposes both yield the exact same matching."*

The proof of the above claim is clear by virtue of the correctness of GaleShapely algorithm. That is, the version of the Gale-Shapely algorithm where the male proposes and the version where the female proposes are both correct and hence will both yield a stable matching. However since there is a unique stable matching, both versions of the algorithms should yield the same matching.

Next, we prove the converse: *If the version of the Gale-Shapely algorithm where the male proposes and the version where the female proposes both yield the exact same matching then there is a unique stable matching."*

The proof of the converse is perhaps more interesting. For the definition of best valid partner, worst valid partner etc., see page 10-11 in the textbook.

Let S denote the stable matching obtained from the version where men propose and let $S0$ be the stable matching obtained from the version where women propose.

From (page 11, statement 1.8 in the text), in S, every woman is paired with her worst valid partner. Applying (page 10, statement 1.7) by symmetry to the version of Gale-Shapely where women propose, it follows that in $S0$, every woman is paired with her best valid partner. Since S and $S0$ are the same matching, it follows that for every woman, the best valid partner and the worst valid partner are the same. This implies that every woman has a unique valid partner which implies that there is a unique stable matching.

Rubric (10pt):

- 4pts: Correct '=>' proof
  - (a) 1pt: Correctly state forward claim
  - (b) 3pt: Correctly justify forward claim
- 6pts: Correct '<=' proof
  - (a) 1pt: Correctly state backwards claim
  - (b) 5pts: Correctly justify backwards claim

Ungraded Problems

7. Solve Kleinberg and Tardos, Chapter 1, Exercise 2. (5pts)

True. Suppose S is a stable matching where m and w are not paired with each other. Suppose that instead m is matched with w' and w is matched with m'. Then the pairing (m, w) is an instability with respect to S, since m prefers w over w' and w prefers m over m', contradicting the stability of S. Thus, every stable matching must contain (m, w).

Rubric (5pt):

- 2 pt: Correctly identify the statement as true.
- 3 pt: Provide a correct explanation.

8. Determine whether the following statement is true or false. If it is true, give an example. If it is false, give a short explanation. (5pts)

For some $n \geq 2$, there exists a set of preferences for n men and n women such that in the stable matching returned by the G-S algorithm, every woman is matched with their most preferred man, even though that man does not prefer that woman the most.

True: Consider the following set of preferences among $n = 3$ men and women:

Men	Preferences	Women	Preferences
1	C > A > B	A	1 > 2 > 3
2	A > B > C	B	2 > 3 > 1
3	A > C > B	C	3 > 1 > 2

Then an execution of the G-S algorithm may proceed as follows:

1. Man 1 proposes to woman C, who accepts.
2. Man 2 proposes to woman A who accepts.
3. Man 3 proposes to woman A, who rejects.
4. Man 3 proposes to woman C, who accepts, freeing man 1.
5. Man 1 proposes to woman A, who accepts, freeing man 2.
6. Man 2 proposes to woman B, who accepts.

Then the algorithm terminates with man 1, 2, and 3 matched with woman A, B, and C, respectively. Note that every woman is matched with their most preferred man, even though that man does not prefer that woman the most.

Rubric (5pt):

- 2 pt: Correctly identify the statement as true.
- 3 pt: Provide a correct example and explanation.

9. Four students, a, b, c, and d, are rooming in a dormitory. Each student ranks the others in strict order of preference. A *roommate matching* is defined as a partition of the students into two groups of two roommates each. A roommate matching is *stable* if no two students who are not roommates prefer each other over their roommate. Does a stable roommate matching always exist? If yes, give a proof. Otherwise, give an example of roommate preferences where no stable roommate matching exists. (10pts)

A stable matching need not exist.

Consider the following list of preferences. Note a, b, and c all prefer d the least.

- $a : b > c > d$
- $b : c > a > d$
- $c : a > b > d$

Now, there can only be 3 sets of disjoint roommate pairs.

- If the students are divided as (a, b) and (c, d), then (b, c) cause an instability, since c prefers b over d and b prefers c over a.
- If the students are divided as (a, c) and (b, d), then (a, b) cause an instability, since b prefers a over d and a prefers b over c.
- If the students are divided as (a, d) and (b, c), then (a, c) cause an instability, since a prefers c over d and c prefers a over b.

Thus every matching is unstable, and no stable matching exists with this list of preferences.

Rubric (10pts):

- 3 pts: Correctly identify that stable matching won't always exist.
- 7 pts: Provide a correct explanation and example.

10. Solve Kleinberg and Tardos, Chapter 1, Exercise 8. (10pts)

Assume we have three men m_1, m_2, and m_3 and three women w_1, w_2, and w_3 with preferences as given in the table below.
Column w_3 shows the true preferences of woman w_3, while in column w_3' she pretends she prefers man m_3 to m_1.

m_1	m_2	m_3	w_1	w_2	w_3	(w_3')
w_3	w_1	w_3	m_1	m_1	m_2	m_2
w_1	w_3	w_1	m_2	m_2	m_1	m_3
w_2	w_2	w_2	m_3	m_3	m_3	m_1

First, let us consider one possible execution of the G-S algorithm with the true preference list of w_3.

m_1	w_3			w_3
m_2		w_1		w_1
m_3			$[w_3][w_1]w_2$	w_2

First m_1 proposes to w_3, then m_2 proposes to w_1. Then m_3 proposes to w_2 and w_1 and gets rejected, finally proposes to w_2 and is accepted. This execution forms pairs (m_1, w_3), (m_2, w_1) and (m_3, w_2), thus pairing w_3 with m_1, who is her second choice. Now consider the execution of the G-S algorithm when w_3 pretends she prefers m_3 to m_1 (see column w_3').

Then the execution might look as follows:

m_1	w_3		$-$	w_1			w_1
m_2		w_1		$-$	w_3		w_3
m_3			w_3		$-$	$[w_1]w_2$	w_2

Man m_1 proposes to w_3, m_2 to w_1, then m_3 to w_3. She accepts the proposal, leaving m_1 alone. Then m_1 proposes to w_1 which causes w_1 to leave her current partner m_2, who consequently proposes to w_3 (and that is exactly what w_3 prefers). Finally, the algorithm pairs up m_3 (recently left by w_3) and w_2. As we see, w_3 ends up with the man m_2, who is her true favorite. Thus we conclude that by falsely switching the order of her preferences, a woman may be able to get a more desirable partner in the G-S algorithm.

Rubric (10pts):

- 2 pts: Correctly identify the statement as true.
- 8 pts: Provide a correct explanation and example.