CS 3330 Algorithms

1. Stable Matching

- *stable matching*

Solved Exercise 1

Consider a town with n men and n women seeking to get married to one another. Each man has a preference list that ranks all the women, and each woman has a preference list that ranks all the men.

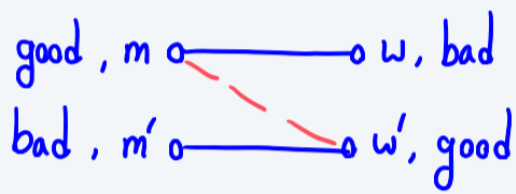
The set of all $2n$ people is divided into two categories: *good* people and *bad* people. Suppose that for some number k, $1 \leq k \leq n - 1$, there are k good men and k good women; thus there are $n - k$ bad men and $n - k$ bad women.

Everyone would rather marry any good person than any bad person. Formally, each preference list has the property that it ranks each good person of the opposite gender higher than each bad person of the opposite gender: its first k entries are the good people (of the opposite gender) in some order, and its next $n - k$ are the bad people (of the opposite gender) in some order.

Show that in every stable matching, every good man is married to a good woman.

proof by contradiction:

Suppose that it is not the case, i.e. there is a good man m who is married to a bad woman. This also means there is a bad man m' who is married to a good woman w'. We claim (m,w') is an unstable pair.



Stable matching

1. Decide whether you think the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample.

True or false? *In every instance of the Stable Matching Problem, there is a stable matching containing a pair (m, w) such that m is ranked first on the preference list of w and w is ranked first on the preference list of m.*

False, consider the following preferences:

$$h_1 : s_1 > s_2 \qquad s_1 : h_2 > h_1$$
$$h_2 : s_2 > s_1 \quad , \quad s_2 : h_1 > h_2$$

It is cas to see $\{(h_1, s_1), (h_2, s_2)\}$ is stable.

6. Peripatetic Shipping Lines, Inc., is a shipping company that owns n ships and provides service to n ports. Each of its ships has a *schedule* that says, for each day of the month, which of the ports it's currently visiting, or whether it's out at sea. (You can assume the "month" here has m days, for some $m > n$.) Each ship visits each port for exactly one day during the month. For safety reasons, PSL Inc. has the following strict requirement:

   (†) *No two ships can be in the same port on the same day.*

   The company wants to perform maintenance on all the ships this month, via the following scheme. They want to *truncate* each ship's schedule: for each ship S_i, there will be some day when it arrives in its scheduled port and simply remains there for the rest of the month (for maintenance). This means that S_i will not visit the remaining ports on its schedule (if any) that month, but this is okay. So the *truncation* of S_i's schedule will simply consist of its original schedule up to a certain specified day on which it is in a port P; the remainder of the truncated schedule simply has it remain in port P.

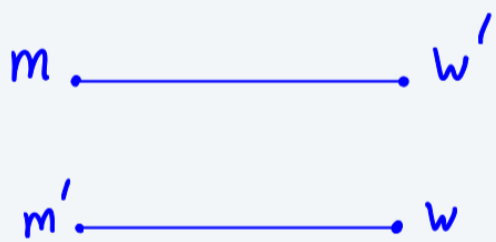
   Now the company's question to you is the following: Given the schedule for each ship, find a truncation of each so that condition (†) continues to hold: no two ships are ever in the same port on the same day.

   Show that such a set of truncations can always be found, and give an algorithm to find them.

2. Decide whether you think the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample.

   True or false? *Consider an instance of the Stable Matching Problem in which there exists a man m and a woman w such that m is ranked first on the preference list of w and w is ranked first on the preference list of m. Then in every stable matching S for this instance, the pair (m, w) belongs to S.*

True: for the sake of contradiction suppose that there exists a stable matching where



Notice that for m, $w > w'$ and for w, $m > m'$ $\Rightarrow$ it is not stable $\lightning$

is already there. Notice that:

① S_2 visits P_1 before P_2 $\Rightarrow$ visit time of ports can be thought as ship's preferences

② P_1 visits S_1 before S_2 $\Rightarrow$ reverse visit time of ships can be thought as port's preferences.

Greedy algorithms

It is easy to see that given these preferences, a stable matching corresponds to a good truncation. Try to prove it yourself.

Stable matching

(b) [15 point] Provide an example in which a student benefits from falsifying their preference list in the Gale-Shapely Algorithm. (Hint: Use the example with three med-school students and three hospitals that we discussed during the lecture.)

(b) Consider the example we studied in the class.

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

hospitals' preference lists

	1st	2nd	3rd
Xavier	Boston	Atlanta	Chicago
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

students' preference lists

5p

Recall that Gale-Shapely algorithm returns $\{(A,X),(B,Y),(C,Z)\}$ as the hospital optimal stable matching.

Stable matching

5p

Next, consider the following preference list in which Xavier falsifies his preference list.

	1st	2nd	3rd
Atlanta	Xavier	Yolanda	Zeus
Boston	Yolanda	Xavier	Zeus
Chicago	Xavier	Yolanda	Zeus

hospitals' preference lists

	1st	2nd	3rd
Xavier	Boston	Chicago	Atlanta
Yolanda	Atlanta	Boston	Chicago
Zeus	Atlanta	Boston	Chicago

students' preference lists

5p

Running the Gale_Shapely algorithm returns {(A,Y),(B,X),(C,Z)} which is means that Xavier benefits from falsifying his preference list.