RAHUL ETHIRAJ
ethiraj@usc.edu

CSCI 570 - Spring 2018 - HW1

Due January 20th

1. Graded Problems :

1. Run G-S twice, once w/ men proposing and once w/ ~~women~~ women proposing if they are the same, we have a consensus optimal stable matching.

   Runtime : $O(n^2)$

   ∴ True

2. a, b, c, d

   Let's assume the below two disjoint pairs is a stable matching.

   a (d, c) b (c, d) c (b, a) d (a, b)

   ⇒ (a, b) (c, d)

   But, there exists an instability where the order changes, as below.

   ⇒ (a, d) (b, c) ⇒ Contradiction

   ∴ Stable matching does not always exist.

3. Step 1: Problem :

   Need to match n students with m hospitals such that they have $p_i > 0$ positions available at hospital h_i and $n \geq m$.

Input:

we have a set of m hospitals $H = \{h_1, h_2 \dots h_m\}$

we have a set of n students $N = \{n_1, n_2 \dots n_n\}$

$$p_i > 0$$

$$n \leq m$$

$$\sum_{i=1}^{m} p_i \leq n$$

Output:

Assignment of students to hospitals w/ no instabilities.

Step 2: Solution similar to G-S algorithm.

Step 3: Proof of correctness:

1. Student is either "committed" to a hospital or "free". A hospital either has available positions or it is "full".

(2) While some h_i has $p_i > 0$

h_i offers position to next student s_j on its preference list

if s_j is free then,

s_j accepts the offer

else (s_j is already committed to a hospital h_k)

if s_j prefers h_k to h_i

s_j remains committed to h_k

else s_j becomes committed to h_i

p_i at $h_i \Rightarrow p_i$--

p_k at $h_k \Rightarrow p_k$++

(3) Alg. terminates after mn iterations.

(4) Solution is a perfect matching

(5) Solution is a stable matching

Step 4 : Prove correctness :

Assume $\sum_{i=1}^{m} p_i < n$, the algorithm terminates with an assignment in which all available positions are filled, because any hospital that did not fill all its positions must have offered one to every student ; but then all these students would be committed to some hospital, which contradicts our assumption that $\sum_{i=1}^{m} p_i < n$.

Step 5 : Proof by contradiction :

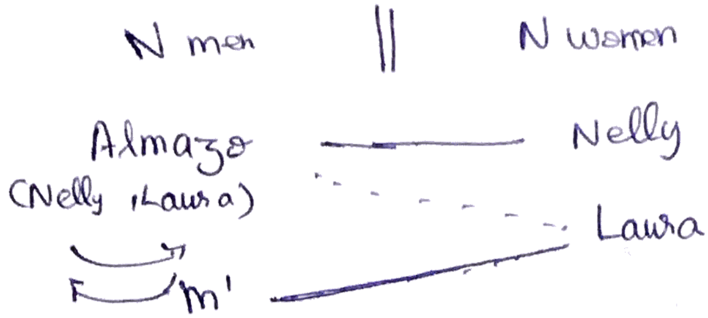
i) $s \to h$
$s' \to$ free
h prefers s' to s.
h would have offered a position to s' before it offered one to s; from then on, s' would have a position at some hospital, and hence would not be free at the end - a contradiction.

ii) $s \to h$
$s' \to h'$
h prefers s' to s
s' prefers h to h'
h would have offered a position to s' before it offered one to s; and moreover s' must have rejected h' in favour of h which he /she preferred.

Step 6 : Perform complexity analysis :

The algorithm terminates in $O(mn)$ steps because each hospital offers a position to a student atmost once, and in each iteration, some hospital offers a position to some student.

4.



Step 1:

Problem : Find a new matching w/o running the G-S algorithm fully and taking advantage of the old results.

Input :

N men $M = \{m_1, m_2 \cdots m_N\}$
N women $W = \{w_1, w_2 \cdots w_N\}$

Output :

Assignment of people after the preference change of Almanzo.

Step 2 :

Partially G-S algorithm.

Step 3 :

Let S^0 be the set of pairs with old matchings.
S' be the set ~~as~~ after the new changes.

Algorithm :

- Break the engagement b/w (Almazo, Nelly) and (m', Laura). Place them into the free ~~pool~~ pool of men M' and women W'.

- Find all such men, where the current engaged women in their preference list, is ranked lower than "Laura".

- Break engagements of all such men and women and place them again in the free pool. m' and w'.

- Run G-S algorithm for m' and w'.

Step 4: Prove correctness
Step 5: Proof by contradiction
Step 6: Perform complexity analysis: $O(n^2)$

} Same as G-S algo.

2. Practice Problems:

1. Read chapter 1

2. Lets assume in a such a way that m and w are not ranked first.

m	w	
(w', w)	(m', m)	(m, w') (m', w) ⇒ Unstable by contradiction
m'	w'	
(w, w')	(m, m')	⇒ (m, w) (m', w')

∴ True

3. True → similar to 2.

4.

	A		B	
Schedule:	a_1	a_2	b_1	b_2
Ratings:	20	40	10	30

$(a_1, b_1) \Rightarrow A$
20 10

$(a_1, b_2) \Rightarrow B$
20, 30

Anyone can change the schedule ⇒ Unstable