Homework 1

● Graded

Student

Kulvir Singh

Total Points

33 / 40 pts

Question 1

Q1 **10** / 10 pts

✔ **+ 3 pts** Mentioned that a stable matching need not exist

✔ **+ 7 pts** Correct explanation

+ 0 pts None of the above

Question 2

Q2 **3** / 10 pts

✔ **+ 3 pts** Identified statement as true

+ 7 pts Gave correct explanation

+ 0 pts None of the above

Please note that this question has nothing to do with Gale-Shapley algorithm

Question 3

Q3 **10** / 10 pts

✔ **+ 3 pts** Identified the statement as false

✔ **+ 7 pts** Gave correct explanation

+ 0 pts None of the above

Question 4

Q4 **10** / 10 pts

✔ **+ 5 pts** Mentioning Luna is on the end of the list.

✔ **+ 5 pts** Mentioning that every other student has a unique, most preferred partner

+ 0 pts None of the above.

Question assigned to the following page: 1

HW1)

given : 4 roommates (A, B, C, D)

stable condition : no 2 not matched student would prefer each other over current roommates.

Prove/Disprove : A stable matching always exists

To prove/disprove this, let us take the following scenario.

Student	Preference of roomate (decreasing order) most pref. ⟶ least pref.		
A	B	C	D
B	C	A	D
C	A	B	D
D	A	B	C

given the above preference chart, we can start selecting roommates for the students.

(i) A selects B as a roommate who accepts as there are no other proposals made to B

$A \longrightarrow B$ (Accepted)

(ii) similarly, as (i) B & C can be formed

$B \longrightarrow C$ (Accepted)

(iii) similarly, as (i) & (ii) C & A can be formed

$C \longrightarrow A$ (accepted)

Question assigned to the following page: 1

Now, we have the following case for D

(iv) D $\xrightarrow{\text{proposes to}}$ A but A rejects as A is already agreed to C's proposal (iii) who is higher preference than D.

Now, D $\xrightarrow{\text{proposes to}}$ B but B rejects as B is already agreed to A's proposal (i) who is higher pref. than D

Lastly D $\xrightarrow{\text{proposes to}}$ C but C rejects as C is already agreed to B's proposal (ii) who is higher pref than D

D remains unmatched.

To conclude, we can confirm that a stable matching may not always exist

Hence **DISPROVED** the claim given in the question.

Question assigned to the following page: 2

(2) Given : Stable Matching Problem.

M ⟶ Preference (W, W_i, W_j - - - -)
W ⟶ Preference (M, M_i, M_j - - -)

To prove, in every stable matching, M is matched with W.

Let us assume the case where the execution of the Gale shaply algorithm is going on and we already have found pairs for a few free men.
Now for the next iteration, [M] is the free man selected.

M ⟶ will propose to the woman with the highest preference in his list, since this is the first time where he is proposing

As per the condition,

(i) [M ⟶ proposes to ⟶ W.]

Now, there can be 2 outcomes,

(ii) W is already engaged

(iii) W accepts M's proposal.

Question assigned to the following page: 2

For case (ii),
W has the option to break off her previous engagement and accept M if and only if M is at a higher preference than her previous suitor.

As per the question, (M) is W's highest priority so no matter who her previous man is, W will always accept (M)'s proposal.

∴ (M, W) will be a stable pair and always be matched.

For case (iii)
since W is free, (M, W) will become a stable pair. Now if any man say M' proposes W, and W breaks up with M, this means that M' >> M in the preference of W. This is not possible as given in the question, M is the highest priority of W.

∴ (M, W) will always be matched.

Hence proved,
In every stable matching, (M ⟶ W)
M is matched with W.

Question assigned to the following page: 3

(3) Prove by contradiction.
For the given preference set, the execution of the Gale - Shapley algorithm has been justified.

For the preference list :—
$(n = 3)$ (M_1, M_2, M_3)
(W_1, W_2, W_3)

Preference order	M_1	M_2	M_3	W_1	W_2	W_3
1st	W_1	W_2	W_3	M_3	M_1	M_1
2nd	W_2	W_1	W_2	M_2	M_2	M_2
3rd	W_3	W_3	W_1	M_1	M_2	M_3

The Gale - Shapley Algorithm will execute as follows :

(i) M_1 proposes first to W_1,
W_1 will accept as she has been free (not engaged)
(M_1, W_1)

(ii) M_2 proposes first to W_2
W_2 will accept as she is free (not engaged)
(M_2, W_2)

(iii) M_3 proposes first to W_3
W_3 will accept as she is free (not engaged)
(M_3, W_3)

Question assigned to the following page: 3

Now there are no free men left so algo terminates and we are left with the following matches

(M_1, W_1)
(M_2, W_2)
(M_3, W_3)

} all of which are the least prefered choices for the women.

Irrespective of the order in which the men will propose, we will always end up with this result for the assumed preference list.

∴ DISPROVED the claim that its not possible for any preference list to match women to least prefered men with men proposing first.

Question assigned to the following page: 4

(4) Let us create a preference table from the given question.

H	He	N	R	G	L
R	G	N	He	H	L
He	R	N	G	H	L
L	H(x)	R(x)	G(x)	He(x)	N(x)
N	H	R	He	G	L
G	N	H	He	R	L

By applying the concept of free choice first of the stable matching problem, we allow H to select its partner

(i) H ⟶ He (He Accepts as she is free and does not have a better option

(ii) R ⟶ G (G Accepts same reason as (i))

(iii) He ⟶ R (R Accepts same reason as (i))

(iv) L ⟶ H (H Accepts same reason as (i))

(v) (N ⟶ H)

Harry at this point has the choice of 2 proposal, one from N and L. Since, N > L. in H's preference

N ⟶ H accepted & L becomes free.

Question assigned to the following page: 4

(vi) $L \longrightarrow R$
who is the next preference for L but since R already has a better proposal, R rejects

(vii) $L \longrightarrow G_1$
again G_1 rejects, same reason as (vi)

(viii) $L \longrightarrow He$
again He rejects, same reason as (vi)

(ix) $L \longrightarrow N$ (accepted, as reason (i))

(x) $G_1 \longrightarrow N$
since N prefers G_1 over L, N accepts G_1's proposal leaving L free.

At point (x) L does not have any partners left to be matched with and hence remains free. This will be always be the case for the given preference list and L will remain free, making it unstable

Hence PROVED, no stable matching for the preferences given in the question