Homework-1:

(1) Read the chapter ✓

(2.) In every instance of 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.

Ans. $\boxed{\text{False}}$

Counter example:

→ Let M be a set of 3 men - m_1, m_2, m_3

→ Let W be a set of 3 women - w_1, w_2, w_3

→ Now let the preference of men be as follows:

$m_1 \longrightarrow w_1, w_2, w_3$

$m_2 \longrightarrow w_2, w_1, w_3$

$m_3 \longrightarrow w_3, w_1, w_2$

→ And let the preference of women be in the following order:

$w_1 \longrightarrow m_2, m_3, m_1$

$w_2 \longrightarrow m_3, m_2, m_1$

$w_3 \longrightarrow m_1, m_2, m_3$

→ Considering the stable matching from men's preference will be:

$\{m_1 w_1, m_2 w_2, m_3 w_3\}$

→ Considering the stable matching from women's preference will be:

$\{w_1 m_2, w_2 m_3, w_3 m_1\}$

→ Hence, it is clear that stable matching exists in conditions where m is not ranked first for w and vice versa for the pair (m, w) as well.

This proves that the given statement is false.

(3) Consider an (example) instance of the stable matching problem in which there exits 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.

Ans: $\boxed{\text{True}}$

For the condition where m and w both have each other as first preference mutually, there is no chance of looking/going to the choices below, neither will any other men/women will be able to break this pair.

For example:

① Men's Preference:

$m_1 \to w_1, w_2, w_3$

$m_2 \to w_1, w_2, w_3$

$m_3 \to w_1, w_3, w_2$

Women's Preference:

$w_1 \to m_1, m_2, m_3$

$w_2 \to m_1, m_3, m_2$

$w_3 \to m_2, m_1, m_3$

Stable matching:
$\{m_1 w_1, m_2 w_2, m_3 w_3\}$

Stable matching:
$\{m_1 w_1, m_3 w_2, m_2 w_3\}$

→ Hence, in any case the stable matching is maintained in m_1 and w_1.
This proves that the given statement is true.

(4) An instance of the stable marriage problem has a unique stable matching if and only if the version of the Gale-Shapley algorithm where the male proposes and the version where the female proposes both yield the exact same matching.

Ans: $\boxed{\text{False.}}$

Counter example:

→ let M be a set of 3 men and W - a set of 3 wom

→ Preferences:

$m_1 \to w_1, w_2, w_3$ $\quad$ $w_1 \to m_2, m_3, m_1$

$m_2 \to w_2, w_1, w_3$ $\quad$ $w_2 \to m_3, m_2, m_1$

$m_3 \to w_3, w_2, w_1$ $\quad$ $w_3 \to m_1, m_2, m_3$

→ Stable matching outcomes:

1. $\{m_1 w_1, m_2 w_2, m_3 w_3\}$
2. $\{m_2 w_1, m_3 w_2, m_1 w_3\}$

→ Hence, it is clear that stable matching exists even if the outcome of where the male proposes and the version where the female proposes, both do not yield the exact same matching.

This proves that the given statement is false.

(5.) A stable roommate problem with 4 students a, b, c, d is defined as follows. Each student ranks the other three in strict order of preference. A matching is defined as the separation of the students into two disjoint pairs. A matching is stable if no two separated students prefer each other to their current roommates. Does a stable matching always exist? If yes, give a proof. Otherwise give an example roommate preference where no stable matching exists.

Ans: Let the preferences of students a, b, c, d be in the following order:

a → b, c, d
b → d, c, a
c → d, a, b
d → b, a, c

→ Stable matching outcome will be: {ab, cd}
→ But, the first preference of b happens to be d, and that of d happens to be b, over their current partners.
This violates the rule for stability.
Hence, this proves that instability exists for this particular scenario.
So, it is not necessary to get stable matching in every scenario.

(7) → Let us consider hospitals − h and h'
→ Let us consider students − s, s', s'' ~~s~~
→ Let the preferences of the hospitals be as follows.
  h' → s', s, s''
  h → s', s, s''
→ Let the preferences of students be as follows:
  s → h, h'
  s' → h, h'
  s'' → h', h
→ On implementing stable matching algorithm on the above choices, we get the following stable outcome
  {hs', h's, h's''}
→ So there is no pair here which satisfies the condition of instability.
(Here, we have considered that h' has two vacancies and h has one vacancy)
→ Algorithm:

While there is a hospital who has vacancy
  and hasn't approached to every student
    Choose such a hospital h
    Let s be the highest ranked student
      in h's preference list to whom h
      has not yet approached.
    If s is free then
      (h, s) gets connected.
    Else s is currently working at h'
      If s prefers h' to h then
        h still has vacancy
      Else s prefers h to h' ~~th~~
        (h, s) gets connected

h' becomes free.

Endif

Endif

Endwhile

Return the set S of (connected) engaged pairs.

(6.) Let S be the schedule of Network A and T be the Schedule of Network B.

Let the Ratings of 4 time slots be as follows:

S	T	Winner.
2	4	T
4	3	S
6	5	S
8	7	S

S has 3 wins

T has 1 win.

Now let there be new schedule T'

S	T'	Winner.
2	3	T'
4	5	T'
6	7	T'
8	4	S

S has one win

T' has three wins.

Hence, instability has arisen as after the change in schedule, Network B wins over Network A.

(8.) When the preference list of Almanzo Wilder changes, implement the following:

→ ~~While there is no unpaired man and woman~~

If Laura is engaged then

If Laura's list has rank(Almanzo) > rank(her current partner)

Almanzo-Laura gets engaged

Nelly becomes free

Laura's ~~current~~ ex partner becomes free.

Laura's ~~current~~ ex partner approaches next woman in his list.

Else Laura's list has rank(current partner) > rank(Almanzo)

Almanzo remains engaged to Nelly

End If

Else

Almanzo - Laura gets engaged.

Nelly becomes free

Laura's ex-partner gets free.

Laura's exp-partner approaches next woman in his list.

End If.

~~End While~~ (Let m be Laura's ex-parner)

~~Return the set S of engaged pairs.~~

1 → While man m is free

Let w be the woman that is ranked next to Laura and m approaches.

If w = Nelly then

m gets engaged with Nellys

~~End If~~

Else m gets engaged with w ≠ Nelly

End If

→ While m' is free

Repeat 1