Name: Hager A. Eldakroury

EID: hae386

A)

For the Gale-Shapely implementation for hospital-optimal, the algorithm is as follows:

```
While a hospital $h_i$ has free slots
    $h_i$ offers a position to the first resident $r_i$ in the hospital's preference list to whom the hospital hasn't offered a position before.
   If ($r_i$ is free)
      Match $r_i$ with $h_i$
      Decrement $h_i$ slots
  Else if ($r_i$ is committed to a hospital $h_j$)
        If($r_i$ prefers $h_i$ to $h_j$ )
             $r_i$  becomes committed to $h_i$
             Decrement $h_i$ slots
            Increment $h_j$ slots
```

B)

Number of hospitals=m

Number of Residents=n

Time Complexity=O(m*n)

Explanation:

- Each hospital can propose to at most n residents since the hospital can't propose to the same resident twice. (m (the outer loop) * n (the inner loop) total O(m*n))

- Operations inside the inner loop of residents:

  - Checking if the resident is free or committed O(1)
  - Comparing the hospitals' rankings O(1)
  - Matching the resident O(1)
  - Incrementing/decrementing the slots O(1)

Thus, the total running time= O(m*n).

C)

For the Gale-Shapely implementation for resident-optimal, the algorithm is as follows:

```
While a resident $r_i$ is free
    $r_i$ proposes to the first hospital $h_i$ in the resident’s preference list to which the resident hasn’t proposed to before.
  If ($h_i$ has a free spot)
        Match $r_i$ with $h_i$
        Decrement $h_i$ slots
       If($r_i$ ranks less than the lowest ranked resident in the hospital’s matched residents)
              Mark $r_i$ as than the lowest ranked resident in the hospital’s matched residents
  Else if ($h_i$ doesn’t have free spots)
        Set $r_j$ as the lowest ranked resident in the hospital matched residents
       If($r_i$ ranks higher than $r_j$)
           $r_i$  becomes committed to $h_i$
          $r_j$  becomes free
           Decrement $h_i$ slots
      Else
          $r_j$  stay committed to $h_i$
```

D)

Number of hospitals=m

Number of Residents=n

Maximum no of slots=s

Time Complexity=O(m*n*s)

Explanation:

- Each resident can propose to at most m hospital since the resident can't propose to the same hospital twice. (n (the outer loop) * m (the inner loop) s (the lowest ranked comparison in the inner loop) total O(m*n*s))

- Operations inside the inner loop of hospitals:

  - Checking if the hospital has free spots or committed O(1)
  - Comparing with the lowest ranked resident O(1)
  - Updating the lowest ranking resident O(s) (s=no of slots)
  - Committing a resident to a hospital O(1)
  - Incrementing/decrementing the slots O(1)

Optimization: since as opposite to the hospital-optimal algorithm, the resident has to compare the ranking with all the matched residents in the hospital (no of slots). In the hospital-optimal, each hospital has to only compare its ranking with another hospital.

As a trial to optimize the running time, I tried to save the lowest ranked resident in each hospital and only compare the resident's ranking to this variable. But the complexity remained the same as updating the lowest ranked variable has a complexity of O(s).

The only other optimal solution I can think of is to have a structure where all matched residents in each hospital are sorted by their ranking and insertion/deletion takes O(1).