Question 1)

Algorithm Idea: For this algorithm, we will continue to use a similar outline that the Gale-Shapely Algorithm used. We will have 3 arrays. m for hospital array, n for student array and r for rejected student array of size n. We have this because this allows the worse-case possible and no students are gain an interview with the hospital. M will represent hospital and N will represent students. All students and hospitals will be initially free when we start this. We will also have a separate array that will be used to hold the leftover students. It will then take one student who is free. It will then check if that student has applied to every hospital or not. If not, we will have it apply to their top spot. The hospital will then have two choices. If the hospital has not assigned an interview with another student, they will assign the current student that have now to an interview. If the hospital has a student already in an interview and the current student is someone they like more, they can cancel the old interview and assign a new interview with the current student. If they already have a student in an interview and like the other student more than the current student applying, then they will reject them. Now, back to the first if statement. If that student has applied to every hospital, they will then put them in the r array, meaning that these students have been rejected by every hospital. It will then print out the pairings for student and hospital.

Algorithm: Initially all m ∈M and all n ∈N are free and array r is empty of size n.

While there is a student n in array n that is free

Choose student n

If student n applied to all hospitals on his list then

n is moved to n’ array and is now considered rejected.

Else n applies to his top hospital that he has no applied to yet.

If m is free then

(n,m) has an interview

Else m currently has an interview set to n’

If m prefers n’ to n then

n is rejected and remains free

Else m prefers n to n’

(n,m) has an interview

n’ is now free

Endif

Endif

Endif

Endwhile

Return the set S of interviews

Proof Idea: Using proof of contradiction and the Gale-Shapely Algorithm, I will prove that there is a stable assignment.

Proof Detail: For sake of contradiction, we will assume that S will cause instability every time. For this problem to cause instability, you can either have h prefer s’ to s when s’ is not assigned to h, or you have h prefer s’ to s and h’ prefers s to s’ when each hospital has the respected other choice. Assuming that the Gale-Shapely Algorithm fits this problem perfectly, we know that each s and h has a preference list, where s gets a choice to apply to their top h and h has a choice to reject or accept their current c. Going to the first way of instability, since we can use the GS Algorithm fairly, we know that s’ will apply to h, even when h has s already. The GS algorithm will keeping looping through all the students that are currently free until there are no free students. Each time s is chosen, they are able to apply to that h and h can compare s to s’ and choose which one they want. This makes the first instability impossible to happen. Now, on to the second instability. Since s’ will always apply to their top h based on the GS algorithm, there is no way they can apply to a h that is lower ranked than the h that they want. This causes this instability to invalid. So based on contradiction and the Gale-Shapely Algorithm, this problem will always have a stable matching.