5. Provide a modified GS that produces set S of hospital-student pairings.

```
Initially all slots ∈ h ∈ Hospitals and s ∈ Students are free
while there is a hospital h in Hospitals such that h has free slots
    choose such a hospital h
    while h has free slots and hasn't proposed to every student s for which (h,s) ∉ F
        let s be the highest ranked student in h's preference list to which h has not yet proposed
        If s is free then
            (h,s) become paired
        Else s is currently paired to hospital h'
            If s prefers h' to h then
                h remains free
            Else s prefers h to h'
                (h, s) become paired
                h' becomes free such that h' has a free slot opened
            Endif
        End if
    End while
End while
return the set S of hospital-student pairs
```

Claim: The algorithm terminates in polynomial time. Specifically $O(n \cdot m)$ for n hospitals and m students.

Proof: Each iteration consists of a hospital proposing (the only time) to students it has never proposed to before. So, let $P(t)$ denote the set of pairs (h,s) such that h has proposed to s by the end of t, we see that the size of $P(t+(i \geq 1))$, where i is the number of students proposed to by h on a given iteration, is strictly greater than the size of $P(t)$. Because slots are not unique, i.e. there is no ordering among the slots of a given h, there are only $n \cdot m$, hospital-student pairs in total, so the value of $P(\cdot)$ can increase at most $n \cdot m$ times over the course of the algorithm. It follows that there are at most $n \cdot m$ iterations. □

Claim: The algorithm produces no pairings of students s and s' and hospital h such that s is assigned to h, and s' is assigned to no hospital, but h prefers s' to s.

Proof: Suppose for a contradiction that the algorithm produces a pairing of students s and s' and hospital h such that s is assigned to h and s' is assigned to no hospital but h prefers s' to s. Because h proposes to students in order of declining preference, we know that h will propose to s' before s. Now, because students never become unmatched after being paired, they only trade up, we know that from this point on s will be paired to some hospital at the termination of the algorithm. Continuing, it may occur that s' will unmatch h at which point h will continue proposing down it's list. When the algorithm terminates h is paired with s and s' is paired with another hospital – a contradiction. Thus, if h prefers s' to s the algorithm produces no pairings such that h is paired to s and s' is paired to no hospital. □

Claim: The algorithm produces no pairings of students s and s' and hospitals h and h' such that s is paired to h, s' is paired to h' but, h prefers s' to s and s' prefers h to h'.

Proof: Suppose for a contradiction that in matching S that h prefers s' to s and s' prefers h to h' but h is paired to s and h' is paired to s'. Note that by definition h's resultant pairings including s represent it's last proposals. So, either h didn't propose to s' in which case s is higher on h's preference list than s', contradicting that h prefers s' to s, or h was rejected by s' in favor of some other h'' that either is or isn't h' but that s' prefers to h contradicting the supposition that s' prefers h to h'. Because both lead to contradictions, it follows that S is a stable matching. □