2-6-20

Programming Assignment 1

Part 1: Report

(a) location optimal pseudocode:

```
Initially all locations m ∈ M and employees n ∈ N are free
while ∃ m that is has a free slot
    do choose such a location m
        let n be the highest ranked employee in m's preference list
        whom has not been hired by m
        if n is free
            then (m,n) become paired
        else n is currently working at m'
            if n prefers m to m'
                (m,n) become paired
                m' is unpaired with n
return the set S of employed pairs
```

(b) complexity is $O(mn)$. In the worst case, the outer loop will iterate m times, and the inner loop will iterate $n-1$ times for each m (if every location is rejected by every employee until finally matched). This gives a total number of iterations of $m \cdot (n-1)$, and this gives a $O(mn)$.

(c)

```
Initially all employees n ∈ N and locations m ∈ M are free.
while ∃ n that is free and hasn't interviewed with every location m ∈ M
    do choose such an employee n
        let m be the highest ranked location in n's preference list
        if m has open slots
            then (n,m) become paired.
        else m is currently employing a set {n1, ..., nslots}
            for all employees n1', n2', ..., nslots'
                if m prefers n to ni'
                    (n,m) become paired
                    (m) is unpaired with ni'
return the set S of employed pairs
```

(d) complexity is $O(mns)$, where s is the number of jobs available at m. The difficulty arises because the job locations can accept multiple employees, which adds a dimension to its list of matchings. This list must be traversed to avoid instabilities.