CMPS 102 — Fall 2018 – Homework 1

Alyssa Melton

I have read and agree to the collaboration policy.

Collaborators: none

Solution to Problem 2

Assume that there are m courses that each need x_m TA's. There are n graduate students that are applying for a TA position (there are more applicants than positions available.) Each course's professor has a preference list of applicants, and each applicant has a preference list of courses to TA for. We want to avoid instability as defined in the homework PDF.

Consider the following algorithm, similar to that of Gale-Shapely for n*n pairing, for pairing TA's to courses:

m = number of courses
x_m = number of spots available for each course (x is different for each m)
n = number of applicants applying for a TAing position

while there exists courses m with spots x_m still remaining
    m offers a position to their most desired applicant that has not already been offered a position by m
        for each applicant n that was offered a spot
            if the applicant n is free
                n accepts position m
            if applicant n is not free and if applicant n prefers this position to already accepted position
                applicant declines other offer and accepts this offer
                old position is now free and this position is now taken

Claim 1. *All matchings are stable matchings.*

Proof. Suppose a course, say, Algorithm Design, and Terrence is an unstable matching, such that the Algorithm Design professor prefers Terrance over a few other applicants that are currently matched with him, call them Jerry and Joe, and Terrance prefers to TA for Algorithm Design versus Data Structures, which he is currently matched (or is not put into any any class at all). Consider two cases.

Case 1: Algorithm Design never offers Terrence a position. Then, by definition of the algorithm, Algorithm Design prefers Jerry and Joe (and all others who are assigned to TA for Algorithm Design) over Terrence. Otherwise, Terrence would have been offered a position. Then, Algorithm Design and Terrence would be a stable matching, which is a contradiction to our proposal.

Case 2: Algorithm Design offers Terrence a position. Then, at some point Terrence must have declined Algorithm Analysis's offer for another class, say, Computer Architecture. Then, by definition of the algorithm, Terrence must prefer Computer Architecture over Algorithm Anlysis, in which case Algorithm Design and Terrance would be a stable matching. Another contradiction.

Therefore, we see that all pairings must be stable. □

Claim 2. *This algorithm's time and space complexity is* $O(mn)$.

Proof. No course offers a position to an applicant more than once. If the all m courses propose to all n applicants, there would be $m * n$ offers. Since no course offers a position to an applicant more than once, there cant be any additional offers. □