

Homework 1: Stable Matching

Due: Sept. 18th by 10am
Hand in via Moodle

This homework assignment is shorter than future assignments will be. Please use the extra time to do a really good job with the write-up. Be sure to see the Algorithm Design write-up guidelines on the course website.

Thesis Advisor Allocation Problem

You will be assigned an advisor for your senior thesis using a variant of the Gale-Shapley algorithm for Stable Matching. Let n be the number of senior CS majors and m be the number of advisors. Each advisor has a fixed number of slots for thesis advisees they can advise, and this number may be different for each advisor. Each student has a ranking of all advisors in order of preference and each advisor has a ranking of all students. We assume there are the same number of total advisee slots summed across all advisors as there are students.

We would like to find a way to assign students to advisors so that all students receive an advisor and so that the matching is *stable* using the same meaning of stability as the Stable Matching Problem, i.e., that there are no students who would like to switch to an advisor where that advisor would also prefer to advise that student.

1. **Algorithm Design Question:** Show that there is always a stable assignment of students to advisors and give an algorithm to find one.
2. **Who Benefits?:** There are a number of specific algorithmic choices that you may not have identified and explained in your above algorithmic design but which are important to you as a student receiving a thesis advisor via this algorithm. *You should consider the below choices as creating modifications to*

the algorithm you introduced above, but not as building on each other in a cumulative way; i.e., you may consider the below questions independently of each other within the context of your introduced algorithm.

- (a) *Students or advisors in the outer loop.* Your algorithm likely has an outer loop that goes through all students to check their rankings of advisors *or* goes through all advisors to check their rankings of students. These two options result in different stable matchings. Give an example where the option chosen matters and justify a specific choice of advisors or students in the outer loop.
- (b) *Ordering of the outer loop.* Within the group chosen above for the outer loop (all students or all advisors), there will also be a student/advisor who is first in the loop, someone will be second, etc. Either:
 - i. Give an example where this order matters, explain how would you like this ordering to be set up to best benefit *you*, and explain how the ordering should be set up to be fair to all students. Your answer should include an example and a paragraph description.
OR
 - ii. Prove that the ordering of students or advisors in the outer loop does not matter.
- (c) *Honesty of rankings:* As a student, is it possible for you to submit a ranking ordering that is dishonest and thus end up with an advisor you prefer more than the one you would have ended up with if you had been honest? Specifically consider the case where you might lie by swapping the ordering of two advisors in your ranking. Either prove that lying in this way cannot improve your assigned advisor or given an example of a set of student and advisor rankings for which lying would be a benefit. You may do this with advisors or with students in the outer loop; just state which version of the algorithm you are using - you do not need to give a full write-up for this small change.