1 Stable Matching

Consider the set of candidates $C = \{1, 2, 3\}$ and the set of jobs $J = \{A, B, C\}$ with the following preferences.

C	J		
1	A	B	C
2	B	A	C
3	A	B	C

J	C		
A	2	1	3
B	1	2	3
C	1	2	3

Run the applicant propose-and-reject algorithm on this example. How many days does it take and what is the resulting pairing? (Show your work)

Solution:

The algorithm takes 3 days to produce a matching. The resulting pairing is {(A, 1), (B, 2), (C, 3)}

Jobs	Day 1	Day 2	Day 3
A	1,3	1	1
B	2	2,3	2
C			3

2 Good, Better, Best

In a particular instance of the stable marriage problem with n applicants and n jobs, it turns out that there are exactly three distinct stable matchings, S_1, S_2, and S_3. Also, each applicant m has a different partner in the three matchings. Therefore each applicant has a clear preference ordering of the three matchings (according to the ranking of his partners in his preference list). Now, suppose for applicant m_1, this order is $S_1 > S_2 > S_3$.

Prove that every applicant has the same preference ordering $S_1 > S_2 > S_3$.

Hint: Recall that a applicant-optimal matching always exists and can be generated using applicant proposes matching algorithm. By reversing the roles of stable matching algorithm, what other matching can we generate?

Solution:

In class, you were given the traditional propose-and-reject algorithm, which was guaranteed to produce a applicant-optimal matching. By switching applicant's and jobs's roles, you would be guaranteed to produce a job-optimal matching, which, by a lemma from class, would also be applicant-pessimal. By the very fact that these algorithms exist and have been proven to work in this way, you're guaranteed that an applicant-optimal and a applicant-pessimal matching always exist.

Since there are only three matchings in this particular stable matching instance, we thus know that one of them must be applicant-optimal and one must be applicant-pessimal. Since m_1 prefers S_1 above the other stable matchings, only that one can be applicant-optimal by definition of applicant-optimality. Similarly, since m_1 prefers S_3 the least, it must be the applicant-pessimal. Therefore, again from definitions of optimality/pessimality, since each applicant has different matches in the three stable matchings, they *must* strictly prefer S_1 to both of the others, and they *must* like S_3 strictly less than both of the others. Thus, each applicant's preference order of stable matchings must be S_1, S_2, S_3.