Discrete Mathematics: Homework #2

Due on February 1, 2025 at 4:00pm

Professor Satish Rao

Zachary Brandt
zbrandt@berkeley.edu

Problem 1: Universal Preference

Suppose that preferences in a stable matching system are universal: all n jobs share the preferences $C_1 > C_2 > \cdots > C_n$ and all candidates share the preferences $J_1 > J_2 > \cdots > J_n$.

A) What pairing do we get from running the algorithm with jobs proposing? Prove that this happens for all n.

B) What pairing do we get from running the algorithm with candidates proposing? Justify your answer.

C) What does this tell us about the number of stable pairings? Justify your answer.

Part A

The pairings we get from running the stable matching algorithm with jobs proposing is one where, for an index i, candidate C_i is paired with job J_i, i.e., the pairings we get are $\{(C_1, J_1), (C_2, J_2), \ldots, (C_n, J_n)\}$.

To prove that this happens for all n, I'll use the principle of induction to prove the *base case* and the *inductive step*.

- *Base Case*: When $n = 1$, the claim holds since there is only one possible pairing (C_1, J_1).
- *Inductive Hypothesis*: Assume that the pairing we get is $\{(C_1, J_1), (C_2, J_2), \ldots, (C_k, J_k)\}$ for some value of $n = k$ where $k \in \mathbb{N}$.
- *Inductive Step*: For $n = k + 1$, we can show that the preferences and matching are the following:

Jobs	**Candidates**
J_1	$C_1 > C_2 > \cdots > C_k > C_{k+1}$
J_2	$C_1 > C_2 > \cdots > C_k > C_{k+1}$
$\vdots$	$\vdots$
J_k	$C_1 > C_2 > \cdots > C_k > C_{k+1}$
J_{k+1}	$C_1 > C_2 > \cdots > C_k > C_{k+1}$

Candidates	**Jobs**
C_1	$J_1 > J_2 > \cdots > J_k > J_{k+1}$
C_2	$J_1 > J_2 > \cdots > J_k > J_{k+1}$
$\vdots$	$\vdots$
C_k	$J_1 > J_2 > \cdots > J_k > J_{k+1}$
C_{k+1}	$J_1 > J_2 > \cdots > J_k > J_{k+1}$

The algorithm takes $k + 1$ days to produce a stable matching. The resulting pairing is as follows. The circles indicate the job that a candidate picked on a given day (and rejected the rest).

Candidate	**Day 1**	**Day 2**	$\ldots$	**Day k**	**Day $k+1$**
C_1	$1, 2, \ldots, k, k+1$	1	$\ldots$	1	1
C_2		$2, \ldots, k, k+1$	$\ldots$	2	2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
C_k				$k, k+1$	k
C_{k+1}					k+1

It takes 1 more day for the algorithm to terminate, and since candidate C_{k+1} is least preferred by all jobs, it will be proposed to last. The job proposing will be job J_{k+1}, since it is least preferred by all candidates. Therefore, we get a pairing $\{(C_1, J_1), (C_2, J_2), \ldots, (C_k, J_k), (C_{k+1}, J_{k+1})\}$. By induction, we have shown that we get a pairing $\{(C_1, J_1), (C_2, J_2), \ldots, (C_n, J_n)\}$ for all n.

Problem 1 continued on next page...

Part B

The pairings we get from running the stable matching algorithm with jobs proposing is one where, for an index i, candidate C_i is paired with job J_i, i.e., the pairings we get are $\{(C_1, J_1), (C_2, J_2), \ldots, (C_n, J_n)\}$.

Part C

Because both the job-optimal and candidate-optimal stable matching algorithms produce the same pairing, there is only 1 stable pairing.