Announcements

- HW 1 and Vitamin 1 due tomorrow at 4PM
- In-person discussion will be in Latimer 102

stable matching instance = a set of n jobs and n candidates where each job and candidate has a preference list

Example:

Jobs	Candidates
1	A > B > C
2	B > C > A
3	C > A > B

Candidates	Jobs
A	2 > 3 > 1
B	3 > 1 > 2
C	1 > 2 > 3

matching = a set of (J, C) pairs where every job is matched to exactly one candidate and every candidate is matched to exactly one job

Example of a matching: $\{(1, B), (2, A), (3, C)\}$

rogue couple = a (J, C) pair where J prefers C over its currently matched candidate and C prefers J over her currently matched job

Example: In the matching above, (2, C) would be a rogue couple because 2 prefers C > A and C prefers 2 > 3.

stable matching = a matching with no rogue couples

Example: In the stable matching instance above, the stable matchings are

$M_1 = \{(1, A), (2, B), (3, C)\}$

$M_2 = \{(1, B), (2, C), (3, A)\}$

$M_3 = \{(1, C), (2, A), (3, B)\}$

job-optimal = every job is matched with their best possible candidate in any <u>stable</u> matching

Example: M_1 is job-optimal, while M_3 is candidate-optimal.

job-pessimal = every job is matched with their worst possible candidate in any <u>stable</u> matching

Example: M_1 is candidate-pessimal, while M_3 is job-pessimal.

Propose-and-Reject / Stable Matching Algorithm

Every morning: Each job proposes to its most-preferred candidate who has not yet rejected this job

Every afternoon: Each candidate puts her most preferred offer on a string and rejects all other jobs

Every night: Each rejected job crosses the candidate off its list.

Repeat until there are no rejections.

Note: The traditional propose-and-reject algorithm involves jobs proposing and candidates rejecting.

When jobs propose, the stable matching is job-optimal.

job-optimal $\iff$ candidate-pessimal

job-pessimal $\iff$ candidate-optimal

If there is only one stable matching, the job-optimal matching and the candidate-optimal matching are the same.

CS 70 Discrete Mathematics and Probability Theory
Spring 2022 Koushik Sen and Satish Rao

DIS 1B

1 Stable Matching

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

Jobs	Candidates
1	A > B > C
2	B > A > C
3	A > B > C

Candidates	Jobs
A	2 > 1 > 3
B	1 > 3 > 2
C	1 > 2 > 3

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

	Day 1	Day 2	Day 3	Day 4	Day 5
A	1, 3	1	1, 2	2	2
B	2	2, 3	3	1, 3	1
C					3

Final matching: $\{(1, B), (2, A), (3, C)\}$

2 Propose-and-Reject Proofs

Prove the following statements about the traditional propose-and-reject algorithm.

(a) In any execution of the algorithm, if a candidate receives a proposal on day i, then she receives some proposal on every day thereafter until termination.

For any day $k \geq i$, if a candidate gets a proposal on day k, then she will accept one and then that job will propose again on day $k+1$. By induction, she will receive a proposal on every day after i.

(b) In any execution of the algorithm, if a candidate receives no proposal on day i, then she receives no proposal on any previous day j, $1 \leq j < i$.

If a candidate receives a proposal on day j, then she will also receive a proposal on day i by part (a). Thus proven by contraposition.

(c) In any execution of the algorithm, there is at least one candidate who only receives a single proposal. (Hint: use the parts above!)

Let k be the last day. There must be at least one candidate who did not receive a proposal on day k-1, otherwise the algorithm would have ended earlier.
By part (b), this candidate who did not receive a proposal on day k-1 also did not receive any proposals earlier, so they only received a single proposal on day k.

3 Be a Judge

By stable matching instance, we mean a set of jobs and candidates and their preference lists. For each of the following statements, indicate whether the statement is True or False and justify your answer with a short 2-3 line explanation:

(a) There is a stable matching instance for n jobs and n candidates for $n > 1$, such that in a stable matching algorithm with jobs proposing, every job ends up with its least preferred candidate.

False. This would require every job being rejected n-1 times and every candidate rejecting n-1 jobs, but this is impossible by question 2.

(b) In a stable matching instance, if job J and candidate C each put each other at the top of their respective preference lists, then J must be paired with C in every stable pairing.

True. If J is not paired with C, then (J, C) would be a rogue couple, so the matching is not stable.

(c) In a stable matching instance with at least two jobs and two candidates, if job J and candidate C each put each other at the bottom of their respective preference lists, then J cannot be paired with C in any stable pairing.

False.

1	A > B
2	A > B

A	1 > 2
B	1 > 2

{(1, A), (2, B)} is a stable matching where 2 and B are paired despite being at the bottom of each other's preferences.

(d) For every $n > 1$, there is a stable matching instance for n jobs and n candidates which has an **unstable** pairing where **every** unmatched job-candidate pair is a rogue couple or pairing.

True. If we match every job with its least-preferred candidate and every candidate with her least-preferred job, then every unmatched pair is a rogue couple.