- HW 0, HW 1 released.
- Recitations begin the week of Feb 1.
- If you have trouble accessing Canvas, Gradescope, OTHQ, please make a private post on Piazza.
- Enroll yourself on Piazza.
- Violation of course policies will be dealt with harshly.

**Stable Matching.**

**Input:** Set of students $X$

Set of students $Y$

Each vertex has a preference list ranking all vertices from the other.
Set.

**Stable**

Output: A matching that pairs each vertex in X with a vertex in Y.

(1, 1, 2, 3)

Saei's alg.

Permute all vertices in Y, \( \pi \) : permutation.

\[
\begin{align*}
\pi(1) &= a \\
\pi(2) &= b \\
\pi(3) &= c \\
\vdots & \vdots
\end{align*}
\]

Instability in a matching.

\( (y', y) \) \text{ in } Y
When a matching does not have an instability it is called a stable matching.

Q. Does every instance have a stable matching?

Q. Can an instance have exactly one stable matching? Can an instance have > 1 stable matchings?

Q. Suppose an instance has a stable matching. Can we find one?

(y, y')

(\text{stable matching})

(x', x')

(\text{stable matching})
\[(y, y') \rightarrow (x', x) \quad (y, y') \rightarrow (x, x')\]

\[
\begin{align*}
&\text{a} \quad \text{b} \quad \text{c} \quad \text{d} \\
&\text{e} \quad \text{f} \quad \text{g} \quad \text{h} \\
\end{align*}
\]

**GS (Gale-Shapley)**

- Initially all vertices are free.

- While there exists a vertex \( x \in X \) s.t. \( x \) has not
yet propose to all vertices in Y, do

\[ y \leftarrow \text{highest ranked vertex in } Y \text{ whom } x \text{ has not yet proposed to}. \]

if \( y \) is free then

\( (x, y) \) become a pair

else if \((x', y)\) exists then

if \( y \) prefers \( x \) to \( x' \) then

\( (x, y) \) becomes a pair

\( x' \) becomes free

- return all pairs

Q: Will this algorithm always terminate?

- each vertex in \( X \) proposes at most once
to each vertex in $Y$.
- $\vdash \leq n^2$ proposals in all
- $\vdash$ alg. terminates.

Observation 1 Once a vertex $y \in Y$ receives its first proposal, $y$ always remains engaged
and as the alg. progresses, its partner can only get better.

Observation 2 At any point in the GS alg each vertex is paired with $\leq 1$ vertex.

Lemma: GS alg. outputs a perfect

matching. all vertices in $X \cup Y$ are matched.
Proof: Assume for contradiction that GS alg does not give a perfect matching. Let $x \in X$ be a vertex that is free at the end of the alg.

- $x$ must have a match to all vertices in $Y$.
- By Observation 1, all vertices in $Y$ are paired with vertices in $X$.
- $n-1$ vertices in $X$ are paired with $n$ vertices in $Y$.
- By PHP, some vertex in $X$ must be paired with 2 vertices in $Y$, contradicting Obs 2.
Thus: GS alg. outputs a stable matching.

Proof: Assume otherwise.

\((..., y', ..., y) \xrightarrow{\text{instability}} (..., x, ..., x')\)

1. In the GS alg., \(x\) must have proposed to \(y'\) before \(x\) proposed to \(y\).
2. \(y'\) must have rejected \(x\).
3. \(y'\) rejected \(x\) because if \(x''\) that it prefers over \(x\).
4. \(y'\) ends up with \(x'\). Thus \(x' = x''\) or \(x'\) is ranked higher than \(x''\) & hence \(x'\) is ranked "1" \(x''\).
Contradiction: that $y'$ prefers $x$ over $x'$.

$$\text{valid}(x) = \{ y \in Y \mid \exists \text{ a stable matching in which } (x, y) \text{ is a pair } y'\}$$

$$\text{But}(x) = \{ y \in Y \mid \text{ highest ranked valid partner of } x \}$$

- $y \in \text{valid}(x)$
- $\forall y' \in Y \text{ s.t. } y' \text{ is ranked higher than } y$ in $x$'s list, $y' \in \text{valid}(x)$

$$S^* = \{ (x, \text{But}(x)) \mid x \in X \}$$

**Theorem:** All executions of the GS alg output $S^*$.

**Proof:** Assume for contradiction that $y'$
execution $E$ of the GS alg, some vertices in $X$ do not get paired with their best valid partners. Among all these vertices, let $x \in X$ have the honor of being the first vertex to get rejected by valid $c$. Let $y = \text{Best}(x)$.

Why did $y$ reject $x$?

By defn., there must be a stable matching $S$ that contains $(x, y)$ as a pair.

$S = (\ldots, x', \ldots, x, y)$
valid(y) = \{ x \in X \mid (x, y) \text{ is a pair in some stable matching} \}

\text{wont}(y) = x \text{ if } y = \bullet
- x \in \text{valid}(y) 
- (\ldots x, \ldots)$
- $\forall x' \in X$, s.t. $x'$ is ranked lower than $x$ in $y$'s list, $x' \notin \text{valid}(y)$.

**Then:** All executions of the GS algo pair up $y$ with its worst$(y)$. 