### Stable Marriage Problem

- Small town with \( n \) boys and \( n \) girls.
- Each girl has a ranked preference list of boys.
- Each boy has a ranked preference list of girls.

How should they be matched?

### Count the ways..

- Maximize total satisfaction.
- Maximize number of first choices.
- Maximize worse off.
- Minimize difference between preference ranks.

### The best laid plans..

Consider the couples:
- Jennifer and Brad
- Angelina and Billy-Bob

Brad prefers Angelina to Jennifer.
Angelina prefers Brad to Billy-Bob.

Uh..oh.

### So..

Produce a pairing where there is no running off!

**Definition:** A *pairing* is disjoint set of \( n \) boy-girl pairs.

Example: A pairing \( S = \{\{\text{Brad, Jen}\}, \{\text{BillyBob, Angelina}\}\} \).

**Definition:** A *rogue couple* \( b, g^* \) for a pairing \( S \):
- \( b \) and \( g^* \) prefer each other to their partners in \( S \)

Example: Brad and Angelina are a rogue couple in \( S \).

### A stable pairing??

Given a set of preferences.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>A</td>
<td>C</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>B</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>

Is there a stable pairing?

How does one find it?

Consider a single gender version: stable roommates.

Example: Brad and Angelina are a rogue couple in \( S \).

### The Traditional Marriage Algorithm.

Each Day:
1. Each boy proposes to his favorite woman on his list.
2. Each girl rejects all but her favorite proposer (whom she puts on a string.)
3. Rejected boy crosses rejecting girl off his list.

Stop when each woman gets exactly one proposal.

Does this terminate?

...produce a pairing?

....a stable pairing?

Do boys or girls do “better”?
Example.

<table>
<thead>
<tr>
<th>Boys</th>
<th>Girls</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>X</td>
</tr>
<tr>
<td>B</td>
<td>X</td>
</tr>
<tr>
<td>C</td>
<td>X</td>
</tr>
</tbody>
</table>

Day 1 Day 2 Day 3 Day 4 Day 5

| 1  | A, B | X | A  |
| 2  | C    | B | A, B |
| 3  | B    | A  |

Termination.

Every non-terminated day a boy crossed an item off the list.
Total size of lists? \( n \) boys, \( n \) length list. \( n^2 \)
Terminates in at most \( n^2 + 1 \) steps!

Improvement Lemma:
On any day, if a boy \( b \) on a string, any future boy, \( b' \), on string is at least as good as \( b \).

Proof:
\( P(k) \)- “every day before \( k \) girl had better boy:”
\( P(0) \)– always true as there is no day before.
Assume \( P(k) \). Let \( b \) be boy on string on day \( k \).
On day \( k + 1 \), boy \( b \) comes back.
Girl can choose \( b \) just as well, or do better.
\( \implies P(k + 1) \).

Pairing when done.

Lemma: Every boy is matched at end.
Proof:
If not, a boy \( b \) must have been rejected \( n \) times.
Every girl has been proposed to by \( b \), and Improvement lemma
\( \implies \) each girl has a boy on a string.
and each boy on at most one string.
\( n \) girls and \( n \) boys. Same number of each.
\( \implies b \) must be on some girl’s string!
Contradiction.

Pairing is Stable.

Lemma: There is no rogue couple for the pairing formed by traditional marriage algorithm.

Proof:
Assume there is a rogue couple; \( (b, g^*) \)

\[ b \quad g \quad b \text{ likes } g^* \text{ more than } g. \]
\[ b' \quad g' \quad g' \text{ likes } b \text{ more than } b'. \]

Boy \( b \) proposes to \( g' \) before proposing to \( g \).
So \( g^* \) rejected \( b \) (since he moved on)
By improvement lemma, \( g^* \) likes \( b^* \) better than \( b \).
Contradiction.

Good for boys? girls?

Definition: A pairing is \( x \)-optimal if \( x \)'s partner is its best partner in any stable pairing.
Definition: A pairing is \( x \)-pessimal if \( x \)'s partner is its worst partner in any stable pairing.
Definition: A pairing is boy optimal if it is optimal for boys \( x \).

...and so on for boy pessimal, girl optimal, girl pessimal.
TMA is...

Good for boys??

**Theorem:** TMA produces a boy-optimal pairing.

There are boys who do not get their optimal girl.

Let \( t \) be the first day a boy \( b \) gets rejected by his optimal girl \( g \) from a stable pairing \( S \).

\( b' \) - knocks off \( b \) on day \( t \) \( \Rightarrow g \) prefers \( b' \) to \( b \)

By choice of \( t \), \( b' \) prefers \( g \) to optimal girl.

\( \Rightarrow b' \) prefers \( g \) to his partner \( g' \) in \( S \).

Rogue couple for \( S \).

\( o S \) is not a stable pairing. Contradiction.

Used Well-Ordering principle...again.

---

**How about for girls?**

**Theorem:** TMA produces girl-pessimal pairing.

\( T \) – pairing produced by TMA.

\( S \) – worse stable pairing for girl \( g \).

In \( T \), \((g, b)\) is pair.

In \( S \), \((g, b')\) is pair.

\( g \) likes \( b' \) less than she likes \( b \).

\( T \) is boy optimal, so \( b \) likes \( g \) more than his partner in \( S \).

Rogue couple for \( S \).

\( S \) is not stable.

Contradiction.

---

**Residency Matching..**

The method was used to match residents to hospitals.

In dating software.

For matching jobs to servers....

.....