Lecture outline

- Last week recap
- Final exam date vote
- **Stable matching**
  - Definition
  - Algorithm
  - Applications
- *My* research: stable matching + shortest paths

Last week recap

- Max-Flow problem
  - Algorithms:
    - Ford-Fulkerson $O(f^*(n+m))$
    - Edmond-Karp $O(nm^2)$
  - Applications:
    - Maximum Bipartite Matching $O(nm)$
    - Baseball elimination
- Min-cut problem
  - Krager's algorithm $O(n^4 \log n)$

  Find the minimum cut for ANY cut in the graph, not just $s - t$ cuts as in Max-Flow.

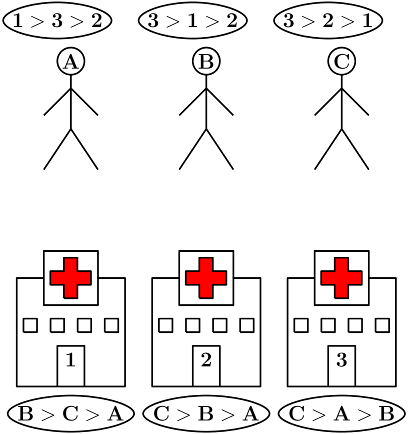
Final exam date vote

- Last lecture: Thu June **8th**: 2pm-3:20pm
- Official date: Thu June **15th**: 1:30pm-3:30pm
- Go to kahoot.it on your phone.

Stable matching problem

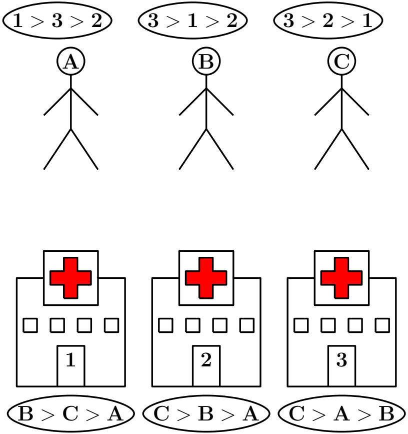
Setting:

- n med students need to find a hospital for their residency.
- n hospitals accept one student each.
- Each student ranks the hospitals by preference.
  - E.g.: student A prefers hospital 1, then 3, then 2.
- Each hospital ranks the students by preference.

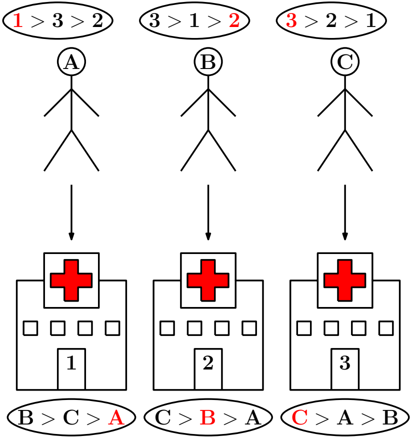


Goal

- Assign a student to each hospital.
- Based on all the preferences.
- Not everyone can have their first choice.
- So... what do we do?



Example of matching

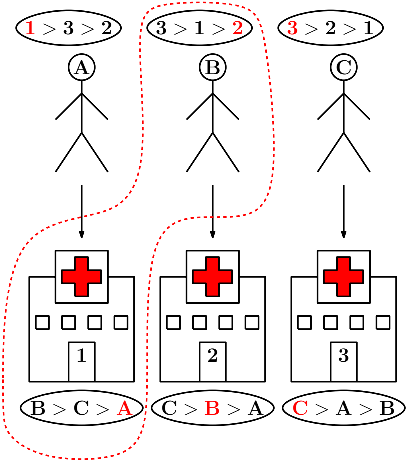


A good matching...?

Unstable pair

- **Unstable pair**: A student and a hospital that prefer each other to their current matches.
  - E.g.: student B and hospital 1

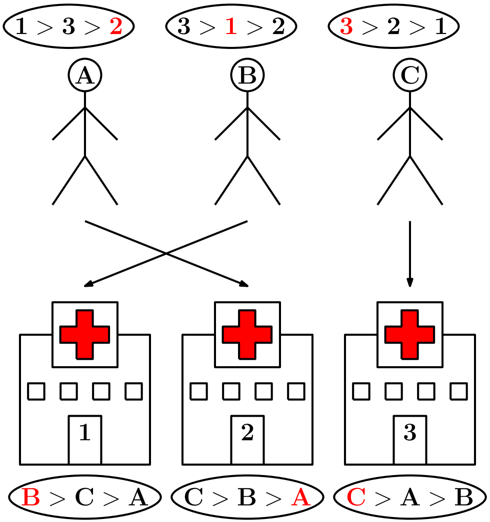
"Why go with them when we can be with each other?"



Stable matching

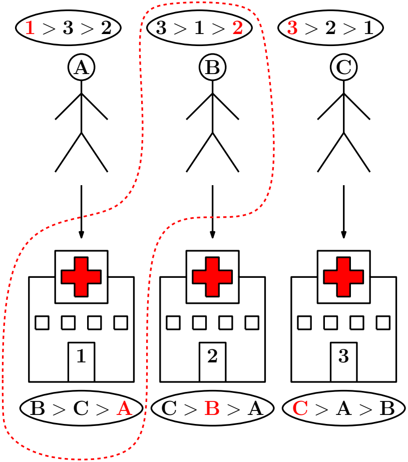
- **Stable matching:** matching with no unstable pairs.

...does it even exist?

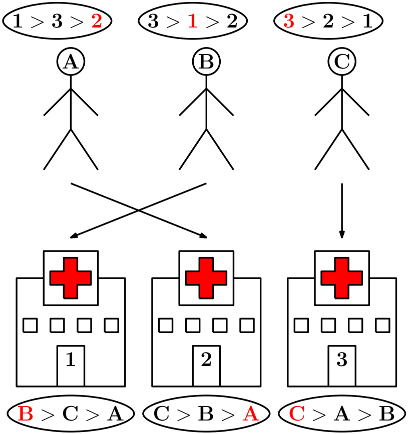


Stable $\neq$ more satisfaction

Non-stable

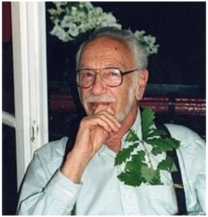


Stable



Background

- Also called "stable marriage problem"
- Originally: marry n women with n men.
- Unstable pair $\rightarrow$ extramarital affair
- Studied in 1962 by



David Gale

Lloyd Shapley

- Shapley won the Nobel Prize of economics for the theory of stable matchings.

Gale-Shapley algorithm

- every student starts unmatched
- repeat while there are unmatched students:
  - each unmatched student applies to its highest-ranked hospital to which he has not applied yet
  - the hospitals retain the highest-ranked student that applies to them
  - if a hospital already had a student from a previous round and gets a better match, the old student becomes unmatched again

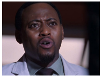
Simulation

- **UCI** Health Center
- **LA** Hospital
- **NY** Hospital
- **Moon** Hospital

- J.D.

- Turk

- Foreman



- 13

Simulation

- Guess: it ended in 3 rounds.

Just in case, I prepared a longer example.

- **UCI**: 13, J.D., Foreman, Turk
- **LA**: J.D., Turk, 13, Foreman
- **NY**: Turk, 13, Foreman, J.D.
- **Moon**: J.D., Turk, Foreman, 13

- **J.D.**: UCI, Moon, NY, LA
- **Turk**: UCI, LA, Moon, NY
- **Foreman**: LA, UCI, NY, Moon
- **13**: LA, NY, Moon, UCI

Gale-Shapley: everyone gets matched

At the end, there cannot be a student (e.g., Turk) and a hospital (e.g., NY) both unmatched.

- Turk must have applied to NY at some point (since students will eventually apply to every hospital, if necessary).
- From the point when Turk applies to NY, NY always has a student thereafter.

Gale-Shapley: stability proof

Consider any student–hospital pair, e.g., Turk–NY.
We must see that Turk–NY cannot be an unstable pair.
(Recall: Turk–NY is an unstable pair if both prefer each other to their matches)

3 cases:

1. Turk is matched to a hospital he prefers MORE than NY. $\rightarrow$ not unstable.
2. Turk is matched to NY. $\rightarrow$ not unstable.
3. Turk is matched to a hospital (e.g., UCI) he prefers LESS than NY.
   Then Turk, applied to NY before UCI, but NY rejected him.
   Therefore, NY prefers its student to Turk. $\rightarrow$ not unstable.

Properties

- Existence

Gale-Shapley always finds a stable matching. Therefore, a stable matching **always exists**.

- Optimality

Who is better off, students or hospitals?
Students get the best match they could possibly get.

- Runtime

No student ever applies to the same hospital twice.
$\rightarrow O(n^2)$

Some applications

- National Resident Matching Program
  - Each hospital has more than one position.
  - Must account for couples $\rightarrow$ becomes NP-hard. Oops
- Internet-based advertisement auctions.
- *Not* marrying people.

"All models are wrong... and some are useful."

My research

- Nodes are houses.
- Edges are roads.
- Big nodes are post offices.
- Goal: stably match each post office to the same number of houses.
- Houses rank post offices by distance.
- Post offices rank houses by distance.