House Allocation:
- Players \( [n] \), each has a house
- Each player has private total ordering (preferences) over all houses
- Can we (re)allocate houses to make players happier?
- "Bid" is an ordering

Top Trading Cycle:
- Let \( A = [n] \)
- While \( A \neq \emptyset \):
  - For each \( i \in A \), let \( f(i) \) be favorite house belonging to agent in \( A \)
  - Directed graph \( G = (A, \{ (i, f(j)) \mid i \in A \} ) \)
Every vertex has outdegree 1
\[ \Rightarrow \geq 1 \] directed cycle,
every vertex in \( \leq 1 \) directed cycle

- For each directed cycle, reallocate: every vertex gives house to predecessor
- Remove all agents who were reallocated from A

Easy properties: will terminate with feasible allocation, in polynomial time

Incentive compatibility:

Truthful bidding \( \Rightarrow \) no worse off (get house at least as good as starting)

Allocated along cycle, could have loops
Truthful bidding is dominant:

Fix agent $i \in [n]$, all other bids.

If $i$ bids truth: $N_i = \{\text{agents allocated in iteration } i\}$

Let $i^* \text{ s.t. } i \in N_{i^*}$

Claim 1: $i$ gets favorite house not in $N_i \cup \ldots \cup N_{i^* - 1}$

Claim 2: If $K$ has path to $i$ in some iteration, then that path in all graphs until $i$ allocated (iteration $i^*$)

$\Rightarrow$ if $i$ lies, creates a cycle with nodes $P$ that have path to $i$

$\Rightarrow$ $P \land (N_i \cup N_{i^* \ldots \cup N_{i^* - 1}) = \emptyset$

$\Rightarrow$ lying doesn't help!
Quality: No utilities or money: how can we even define quality of an allocation?

Def: Blocking coalition of an allocation: set of agents who can pull out from allocation and reallocate among themselves so no one worse off, ≥ 2 better off

Def: A core allocation is an allocation with no blocking coalitions

Thm: TTC returns a core allocation

Pr: Let S ⊆ A be some coalition, and suppose they reallocate among themselves.

∃ graph of disjoint directed cycles:

(ux) means u gives house to x

[Diagram: A graph with directed edges and a triangle, indicating a cycle.]

If cycle all in same Ni:
All in cycle would get favorite act in Ni \cup - UNi+1 from TTC
If cycle has nodes from different $N_i$: $j > i$

\[ u \in N_i \Rightarrow u \in N_j \]

$\implies$ u worse off!

**Thm:** TTC returns only core allocation

**PF:** Induction.

**TTC:** every agent in $N_i$ gets favorite home

$\implies$ Every core allocation agrees with TTC on $N_i$

or else $N_i$ would be blocking coalition

Suppose some core allocation agrees with TTC on $N_i \cup \ldots \cup N_{i-1}$

$\implies$ must agree with TTC on $N_i$, or else $N_i$
a blocking coalition
**Stable Matching**:
- Two sets of agents $U, V$ $|U| = |V| = n$
- Each agent has total ordering over other set

**Def**: A **stable matching** is a perfect bipartite matching (bijection $\pi: U \rightarrow V$) s.t. if $u, v$ not matched then either:
- $u$ prefers $\pi(u)$ to $v$, or
- $v$ prefers $\pi^{-1}(v)$ to $u$

---

Q: - Do stable matchings always exist?
- If yes, can we compute them?
- Incentive-compatible?
Proposed Deferred Acceptance Algorithm:

while there is an unmatched \( u \in U \):
- \( u \) sends a proposal to favorite \( u \in V \) that has not yet rejected \( u \)
- If \( v \) unmatched, match \( u \) and \( v \)
- If \( v \) matched to \( u \in U \), \( v \) chooses to match to favorite of \( u \in U \), rejects the other

**Lemma:** At most \( n^2 \) iterations

**PF:**

Every iteration is a new proposal
At most \( n^2 \) different proposals
\( \Rightarrow \leq n^2 \) iterations

**Lemma:** Terminates with a perfect matching

**PF:** Always matching, why perfect?
Sps not matched
\( \Rightarrow \) rejected by all \( u \in V \)
Every cell gets at least one proposal

Every cell matched at end

Every cell matched \( \geq \)

**Lemma:** The perfect matching is stable

**Proof:** Suppose \( u \in U, v \in V \) not matched.

If \( u \) never proposed to \( v \):

\( u \) matched to someone it prefers over \( v \)

If \( u \) did propose to \( v \):

\( v \) matched to someone it prefers over \( u \)

Is this a "good" stable matching?

**Definition:** For \( u \in U \), let \( h(u) \) favorite node in \( V \) that \( u \) is matched to in any stable matching
Thm: In matching from algorithm, every u \in U is matched to h(u)

PF: Let R = \{ (u,v) : u rejects v in algorithm \}

WTS: If (u,v) \in R \implies no stable matching matches u and v

Induction (algorithm invariant):
At beginning, R = \emptyset

Consider iteration where u rejected by v in favor of u' (add (u,v) to R)
\exists v prefers u' to u
By def of algorithm (u',v') \in R \implies that u' prefers to u
\implies by induction, no stable matching where u'
   does better than u.
So ses \exists stable matching with (u,v)
\implies u' is doing worse than v and u is doing worse than u',
\implies not stable!