Announcements:

1. Prof. Kleinburg's OH moved to 2-3 today, Gates 317. (Generally 3-4 Weds.)

2. Waitlist questions? courses@cis.cornell.edu

   Read CS course enrollment web page.
   Open ticket, if needed, using link at bottom.
   We believe some space will open up in 4820, not enough for everyone on waitlist.

3. Homework 1 is coming Fri morning.

   By Fri morning you should contact us (Ed@email) if you're not on Canvas, Ed/Gradescope for 4820.
2 applicants, 2 hospitals, each hiring 1.

An alternative system based on rankings.

Alice: MSK > MGH  MSK: A > B
Bob: MSK > MGH  MGH: A > B

Def. In a set of applicants (A) and firms (F) a matching is a set of ordered pairs, M, such that

1. each pair in M has exactly one applicant, one firm
2. each party in \( A \cup F \) belongs to at most one pair in M.

belong to one pair: “matched”
belong to no pair: “free”
A perfect matching is one where every party is matched.

Assume now that each applicant has a total ordering of F and each firm has a total ordering of A. ("preferences")

If M is a perfect matching a blocking pair with respect to M is an (applicant, firm) pair \((a, f)\) such that:
1. \(a\) is not matched to \(f\) in \(M\)
2. \(a\) prefers \(f\) to its partner
3. \(f\) prefers \(a\) to its partner.

A stable perfect matching is one without blocking pairs.

Given the participants and their prefs does there exist a stable perfect matching (is it unique?) and how to find one?
A: Yes, a stable perfect matching always exists. (Gale - Shapley, 1950's)

The Proposed Algorithm

Initialize M = Ø

while I a free firm f that hasn't yet proposed to every applicant:
  f finds its most preferred applicant that it hasn't yet proposed to, a.

  if a is free:
    insert (a,f) into M

  if a is matched to some f' ≠ f:
    if a prefers to f to f':
      remove (a,f') from M
      insert (a,f)
    else:
      do nothing

output M