Online Tools and Sources of Information

D2L:

Pre-Lecture Notes

Lecture Notes

HW Assignments Posted

HW Submitted

Syllabus

Links to Piazza and Office Hours Schedule

Any other documnet shared during the semester

Piazza:

Discussion Board

Zoom:

Some Office Hours

Roles and Responsibilities

Instructor lectures and office hrs

TAs

Course Producers HW, proj, Exams, discussions

CS Dept Advisors any reg. issues

DEN Support any tech support Q's

Textbook

Algorithm Design by John Kleinberg and Eva Tardos

Supplemental Textbook

Introduction to Algorithms, 3rd Edition., by Cormen et al.

Your Responsibilities

Attend Lectures and Discussion Sessions

Complete the reading assignments from textbook

Complete and submit HW assignments

Do as many other practice problems as possible

Complete and submit assigned project

Exams

Your Grade

Exam 1	*28%*
Exam 2	*28%*
Exam 3	*28%*
HW assignments	*10%*
Project	*6%*

Grading Scale

90 - 100	*A*	*60 - 64.99*	*C+*
86 - 89.99	*A-*	*55 - 59.99*	*C*
80 - 85.99	*B+*	*50 - 54.99*	*C-*
70 - 79.99	*B*	*45 - 49.99*	*D*
65 - 69.99	*B-*	*Below 45*	*F*

Adjustments to the Grading Scale

Scale will be adjusted if the overall class average falls below 75

At least the top 20% of the class will receive an A

At least the next 10% of the class will receive an A-

Prerequisites

Discrete Math (Mathematical Induction)

Sorting Methods

Basic Data Structures:

Arrays, Linked Lists, Stack, Queue

Graphs Basics:

Tree, Path, Cycle, Directed/Undirected, DAG, Adjacency List/Matrix

Graphs Search Algorithms:

BFS, DFS

High level Syllabus

Today!

- Introduction
- Review of some prereqs + asymptotic notation
- Major algorithmic techniques
  - Greedy
  - Divide & Conquer
  - Dynamic Programming
- Network Flow
- Computational Complexity Theory
- Approximation Algorithms
- Linear Programming

Corrections

1- An algorithm is a set of instructions in machine language.

Kharazmi ~780-840

Algorithm

2- ...Algorithmic science advanced on Wall Street...

3- ... ~~Invite 6 million algorithms for a listen...~~

When studying a problem, we go through the following steps:

1- Come up with a concise problem statement

2- Present a solution

3- Prove correctness

4- Perform complexity analysis

Stable Matching

Stable Matching Example

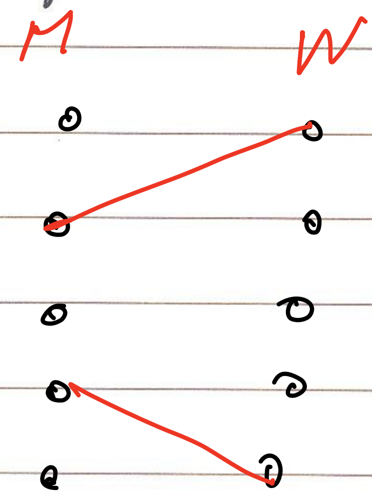
Problem: We are interested in matching $\underline{n}$ men with $\underline{n}$ women so that they could stay happily married ever after.

Step 1: Come up with a concise problem statement.

We have a set of $\underline{n}$ men, $M = \{m_1, \ldots m_n\}$

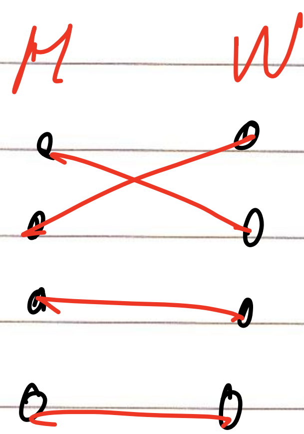
We have a set of $\underline{n}$ women, $W = \{w_1, \ldots w_n\}$

Def. A <u>Matching</u> S is a set of ordered pairs.



A matching

Def. A <u>perfect matching</u> S' is a matching with the property that each member of M and each member of W appear in exactly one pair in S'.



A perfect matching

Add notion of preferences

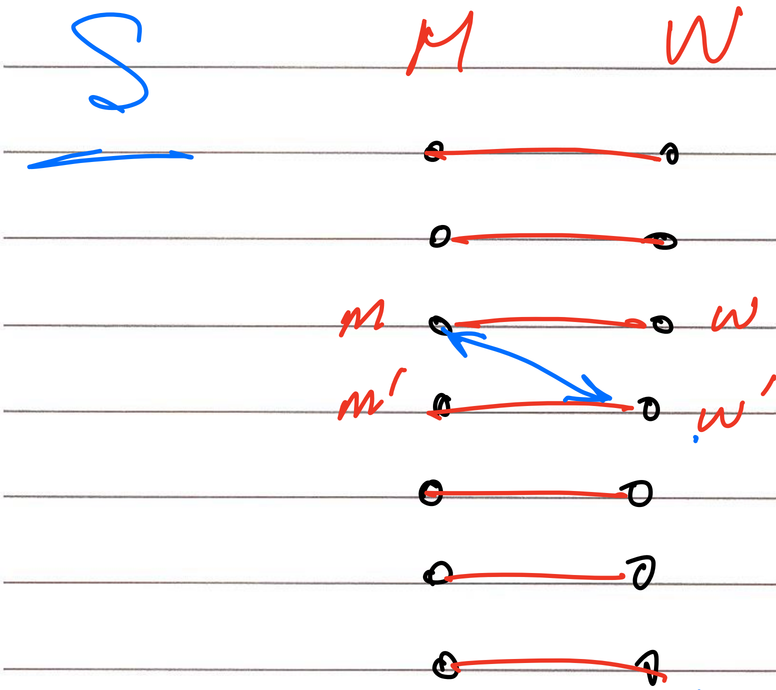
Each man $m \in M$ ranks all women

- $\underline{m}$ prefers $\underline{w}$ to $\underline{w'}$ if $\underline{m}$ ranks $\underline{w}$ higher than w'.

- Ordered ranking of $\underline{m}$ is his preference list

$$P_{m_i} = \{ w_{i_1}, w_{i_2}, \ldots w_{i_n} \}$$

Same for women, i.e. each woman $w \in W$ ranks all men...



Such a pair (m, w') is Called an instability WRT S

Def. Matching S is stable if

1- It is perfect

2- There are no instabilities WRT S

Step 1: Input: Preference lists for a set of n men & n women.

Output: Set of n marriages w/ no instabilities

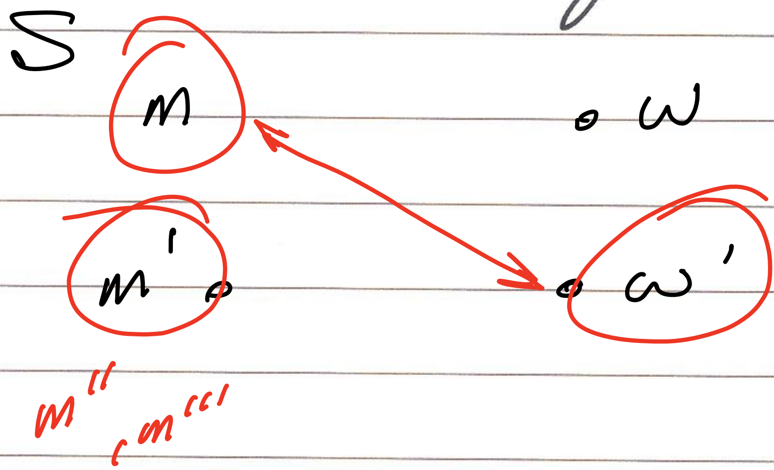
Step 2: Gale-Shapley Alg.

Step 3 Proof of Correctness

1. From the woman's perspective, she starts single, and once she gets engaged and potentially gets into better engagements.
2. From the man's perspective, he starts single, gets engaged, and can rejected repeatedly only to settle for a lower ranking woman.
3. Alg. terminates in at most n^2 iterations.
4. Sol. will be a perfect matching
5. Sol. will be a stable matching

Proof by Contradiction

Assume an instability exists in our solution involving two pairs (m, w), (m', w')



Q: Did <u>m</u> propose to w' at some point in the execution?

If no, then <u>w</u> must be higher than <u>w'</u> on his list $\Rightarrow$ contradiction!

If yes, he must have been rejected in favor of m'' and due to (1) either $m'' = m'$ or m' is better than m''

$\Rightarrow$ contradiction!