COMP 446 / 546 ALGORITHM DESIGN AND ANALYSIS

LECTURE 1 INTRODUCTION

ALPTEKİN KÜPÇÜ

Based on slides of Serdar Taşıran and Piyush Kumar

ALGORITHMS

Algorithm: Named after the 9th century mathematician el-Harezmi (al-Khwarizmi)

An algorithm: A tool for solving a well-specified computational problem.

Problem statement:

- Inputs, outputs
- The desired input/output relationship

Algorithm describes a specific computational procedure for producing the required output

- **Definiteness**: Each step should be defined precisely
- **Correctness**: Should produce correct output for each input
- **Finiteness**: Should produce desired output after a finite number of steps
- **Effectiveness**: Each step should be performed exactly in a finite amount of time
- **Generality**: Should be applicable to any case (not just particular inputs)

We are interested in designing algorithms and analyzing their performance in this course.

EFFICIENCY

Hardware and memory are fast and cheap. Computing getting cheaper and cheaper. Intelligent manpower is expensive

Assumptions:

- Single processor, random-access machine
  - Algorithm steps are executed sequentially
  - Each steps take 1 unit of time
    - Makes our results machine-independent
    - Otherwise impossible to compare!

We assume you already know:

- Big-Oh notation and asymptotic complexity analysis
- Basic data structures and algorithms
- Everything in the Appendix of your textbook
- How to analyze recurrence relations

ANALYZING ALGORITHMS

Is the algorithm correct?

- Does it terminate on all inputs?
- Does it produce the required output?

What amount of resources does the algorithm use up?

- Memory
- Communication bandwidth
- Number of logic gates (if implemented in hardware), speed (depth) of logic circuit
- **Running time**

EFFICIENT ALGORITHMS

A stupid approach uses up computing power faster than you might think. Examples:

- Sorting a million numbers (emails, files, etc.)

$O(n^2)$ algorithm	$2\ n^2$ instructions	10^9 inst/second	2000 sec.
$O(n \log n)$ algorithm	$50\ n \log n$ instructions	10^9 inst/second	1 sec.

- Interactive graphics: Algorithms must terminate in 1/30 of a sec.

OUR FIRST ALGORITHM: STABLE MARRIAGE

THE PROBLEM

- **There are n men and n women**
- **Each man has a preference list, so does each woman.**
- **These lists have no ties.**
- **Devise a system by which each of the n men and n women can end up getting married.**

- **Different metrics possible:**
  - Maximize the number of people who get their first match?
  - Maximize the average satisfaction?
  - Maximize the minimum satisfaction?
- **Can anything go wrong?**

SAMPLE PREFERENCE LISTS

Man	1st	2nd	3rd
X	A	B	C
Y	B	A	C
Z	A	B	C

Woman	1st	2nd	3rd
A	Y	X	Z
B	X	Y	Z
C	X	Y	Z

What is wrong?

Unstable pairs: (X,C) and (B,Y)
X and B prefer each other to their current pairs.

STABLE MATCHING

Man	1st	2nd	3rd
X	A	B	C
Y	B	A	C
Z	A	B	C

Woman	1st	2nd	3rd
A	Y	X	Z
B	X	Y	Z
C	X	Y	Z

No pair creating *instability*.

ANOTHER STABLE MATCHING

Man	1st	2nd	3rd
X	A	B	C
Y	B	A	C
Z	A	B	C

Woman	1st	2nd	3rd
A	Y	X	Z
B	X	Y	Z
C	X	Y	Z

STABILITY IS PRIMARY

- **Any reasonable list of criteria must contain the stability criterion.**
- **A pairing is doomed if it contains a rogue couple.**

- **Solution Idea: Allow the pairs to keep breaking up and reforming until they become stable**
  - Can we argue that the couples will not continue breaking up and reforming forever?

MEN PROPOSE (WOMEN DISPOSE)

Gale-Shapley Algorithm (men propose)

Initialize each person to be free.

while (some man m is free and hasn't proposed to every woman)

  w = first woman on m's list to whom m has not yet proposed

  if (w is free)

    assign m and w as engaged

  else if (w prefers m to her fiancé m')

    assign m and w as engaged, and m' to be free

  else

    w rejects m

ANALYSIS

- Does the algorithm terminate?
- Running time?
- Space requirement?

Improvement Lemma: **If a woman has a committed suitor, then she will always have someone at least as good, from that point in time onwards (and on the termination of the algorithm).**

Corollary: **Each woman will marry her absolute favorite of the men who proposed to her.**

Demotion Lemma: **The sequence of women to whom a man m proposes gets worse and worse (in terms of his preference list)**

LEMMA 1

- **No Man can be rejected by all the Women.**
- **Proof: by contradiction**

Suppose Bob is rejected by all the women.
At that point:

Each women must have a suitor other than Bob
(By Improvement Lemma, once a woman has a suitor she will always have at least one)
The n women have n suitors, Bob not among them.
Thus, there must be at least n+1 men !

COROLLARY OF LEMMA 1

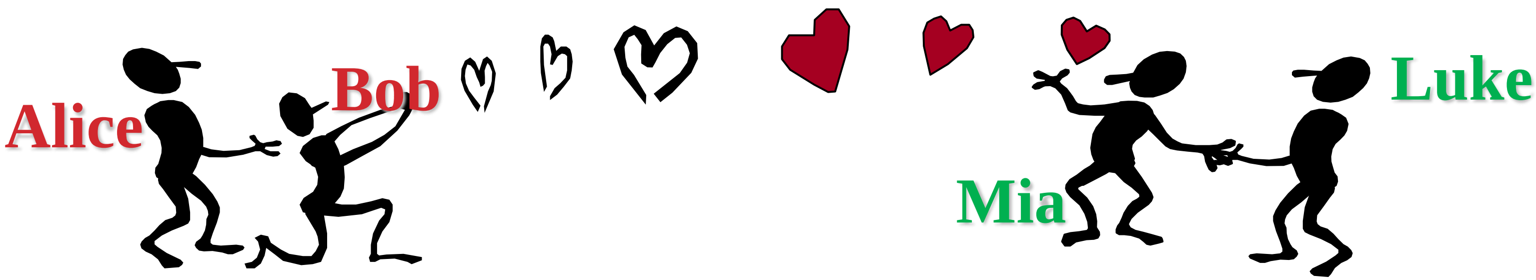
- **If a man m is free at some point in the execution of the algorithm, then there is a woman to whom he has not yet proposed.**

- **The algorithm returns a perfect matching at the end. (Since there is no free man or woman left.)**

A perfect matching is a matching which covers all vertices of the graph.

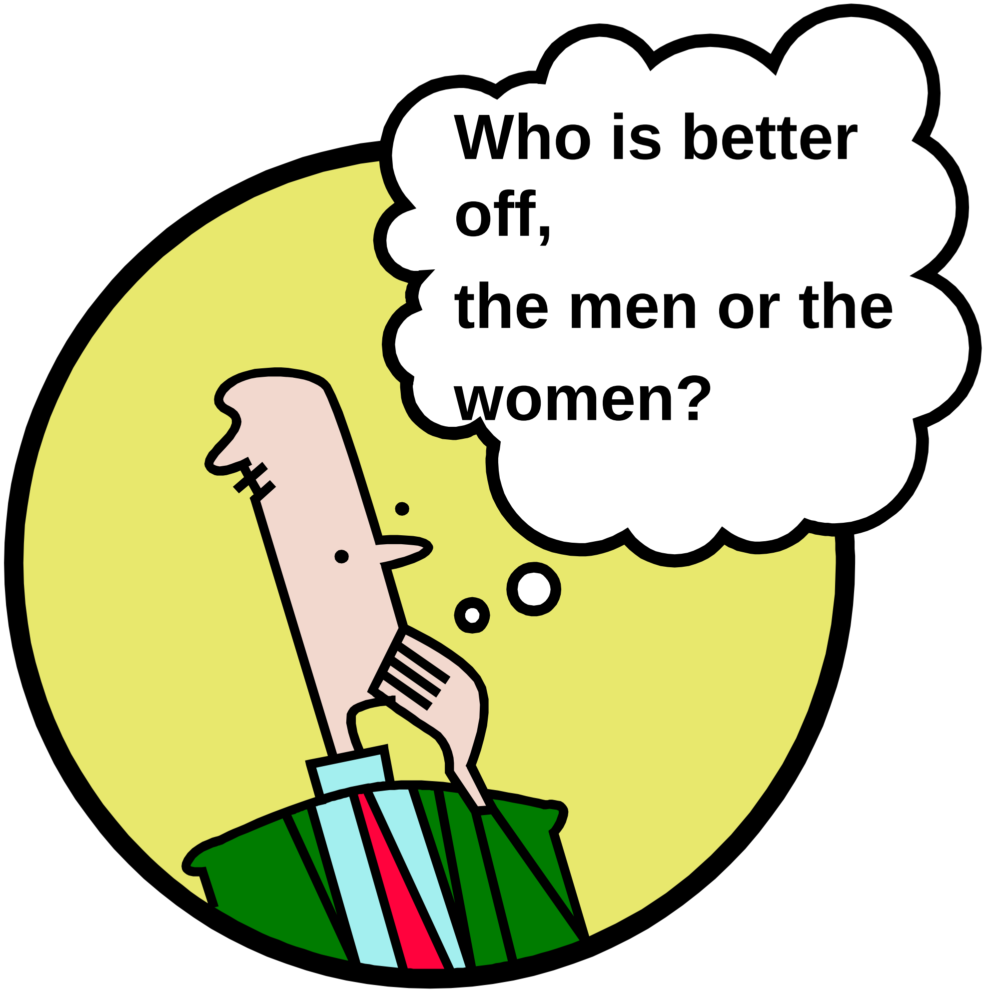
LEMMA 2

- Gale-Shapley algorithm returns a stable matching.
- Proof: by contradiction



- Assume there is an unstable pair: Bob and Mia
  - This means Bob likes Mia more than his partner, Alice.
  - Thus, Bob proposed to Mia before he proposed to Alice.
  - Mia must have rejected Bob for someone she preferred.
  - By the Improvement lemma, she must like her partner Luke more than Bob.
  - Pairs are not unstable. Contradiction!

QUESTION!



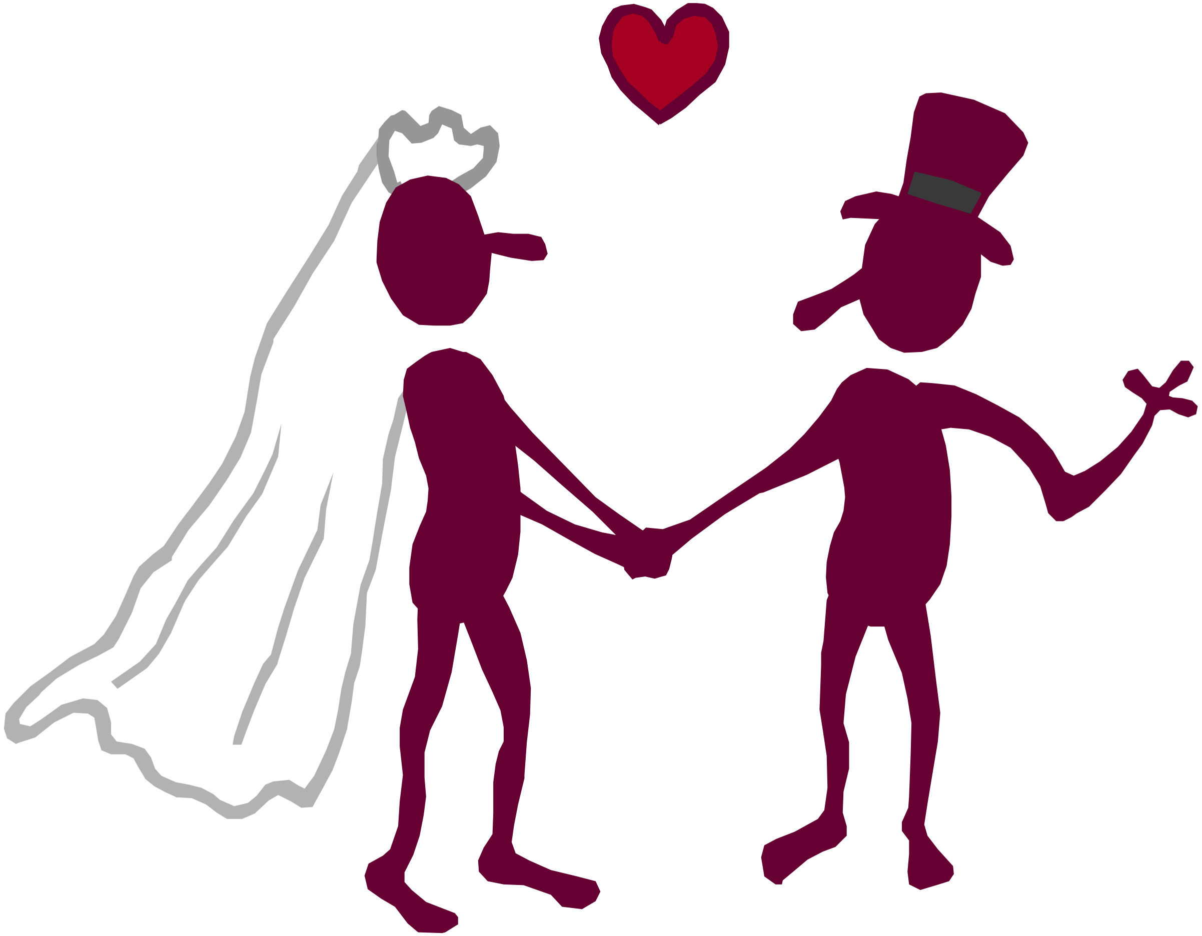
BEST (VALID) PARTNER FOR BOB?

- **Best woman for "Bob"? The woman at the top of Bob's list?**
- **A woman w is a valid partner of a man m if there is a stable matching that contains (m,w).**
- **A man's best valid partner is the highest ranked woman for whom there is some stable pairing in which they are matched**
- **She is the best woman he can conceivably be matched in a stable world.**
- **A Man's *worst valid partner* is the lowest ranked woman in his preference list that is a valid partner.**

DATING DILEMMA

- A pairing is (woman-) man-optimal if every (woman) man gets (her) his best valid partner. This is the best of all possible stable worlds for every (woman) man simultaneously.
- A pairing is (woman-) man-pessimal if every (woman) man gets (her) his worst valid partner. This is the worst of all possible stable worlds for every (woman) man simultaneously.
- The Gale-Shapley algorithm always produces a man-optimal and woman-pessimal pairing.

CONCLUSION: MARRY WELL!



ADVICE TO FEMALES

Learn to make the first move.

EXTENSIONS

Extensions

- Sets of unequal size
- Unacceptable partners
- Indifference
- Many-to-many assignments

Deceit, Coalitions and Strategy

- In general, lying cannot be totally discouraged in Stable Marriage Problem. (sounds familiar??)

Other similar problems

- Internship assignments: Given a set of companies and students, pair them.
- Given airlines and pilots, pair them.
- Computer Vision: Given two images, pair the points belonging to the same point in 3D to extract depth from the two images.
- Dorm room assignments.
- Hospital residency assignments.

REVIEW PROBLEM

Gale-Shapley Algorithm (men propose)

Initialize each person to be free.

while (some man m is free and hasn't proposed to every woman)

    w = first woman on m's list to whom m has not yet proposed

    if (w is free)

        assign m and w as engaged

    else if (w prefers m to her fiancé m')

        assign m and w as engaged, and m' to be free

    else

        w rejects m

Complexity and Space requirements?