

Lecture 4

CSE 331

Sep 4, 2019

Please do keep on asking Qs!

The only bad question is the one that is not asked!

Not just technical Qs but also on how the class is run

We're not mind readers



If you need it, ask for help



Read the syllabus CAREFULLY!

No graded material will be handed back till you pass the syllabus quiz!

Syllabus Quiz

Options

[View handin history](#)

[View writingup](#)

[Download handout](#)

 Due: December 12th 2019, 4:00 pm

 Last day to handin: December 12th 2019, 6:00 pm

Academic Integrity

Question 1: Sharing my answers to this syllabus-quiz with other 311 students.

- It is OK if I do it to help out a friend
- It does not matter since there is no grade attached with it
- It is an academic integrity violation and I should not do it.

Separate Proof idea/proof details

Note

Notice how the question header is divided into proof idea and proof details. THIS IS IMPORTANT! IF YOU DO NOT PRESENT A PROOF IDEA, YOU WILL NOT GET ANY CREDIT EVEN IF YOUR PROOF DETAILS ARE CORRECT.

Proof idea

As the hint suggests there are two ways of solving this problem, if you present both the solutions (but of course you only need to present one)

We begin with the approach of reducing the given problem to a problem you have seen earlier. \Rightarrow Build the following complete binary tree: every internal node in the tree represents a "parent" RapidGrower while its two children are the two RapidGrowers it divides itself into. After x seconds this tree will have height x and the number of RapidGrowers in the container after x seconds is the number of leaf nodes (leaf nodes: these complete binary tree has, which we know is 2^x). Hence, the claim is correct.

The proof by induction might be somewhat simpler for this problem if you are not comfortable with induction. In this case let $R(x)$ be the number of RapidGrowers after x seconds. Then we use induction to prove that $R(x) = 2^x$ while using the fact that $2 \cdot 2^x = 2^{x+1}$.

Proof Details

We first present the induction based proof. Consider the complete binary tree with height x and call it $T(x)$. Further, note that one can construct $T(x+1)$ from $T(x)$ by attaching two children nodes to all the leaves in $T(x)$. Notice that the newly added children are the leaves of $T(x+1)$. Now attach the root of $T(x)$ as the original RapidGrower in the container. Further, for any internal node in $T(x)$ ($x \geq 0$), assign its two children to the two RapidGrowers it divides itself into. Then note that there is a one to one correspondence between the RapidGrowers after x seconds and the leaves of $T(x)$. \Rightarrow Then we use the well known fact (you just learned about here with the great proof when we can find the fact: $T(x)$ has 2^x leaves, which means that the number of RapidGrowers in the container after x seconds is 2^x , which means that the claim is correct.

TA office hours finalized

note  stop following 112 views

TA office hours finalized

The TA office hours are finalized:

<http://www-student.cse.buffalo.edu/~atv/cse331/fall19/policies/tylabus.html>

You can also see the office hours in the course calendar (which in case you did not notice) is on the 331 landing page (<http://www-student.cse.buffalo.edu/~atv/cse331/fall19/index.html>)

The OH will start from Tuesday.

The 1st/1 appointments have not been setup yet. We'll post again once those are ready for bookings.

[office_hours](#)

 [good note](#) | 0 Updated 8 hours ago by An-Rudra

Office hours for proofs

note ☆

stop following 101 views

This week Tue-Th office hours are for proofs!

We have designated all the office hours for Tue, Wed and Th of this week (see [Q76](#) for the details of times) primarily for seeking help with proofs (this something we are trying out for the first time in 331). Also the recitations this week will also focus on proofs. So please do use this opportunity to go ask questions about proofs and related matters.

If you would like to get help but do not have any specific proof related Q, we recommend that you read the [mathematical background](#) pages and if you get stuck somewhere, then you have your Q :-). If you do not get stuck anywhere, you're probably OK with proofs at this point of time.

Apologies for the late notice (esp. for the Tue office hours) but hopefully y'all can use this opportunity.

#pin

[office_hours](#)

- An instructor (Chiranjeev Hemant Bansal) thinks this is a good note -

edit - good note | 3

Updated 11 hours ago by Ash Puro

1-on-1 appointments

Appointments

Instructions and important information for booking and canceling one-on-one meetings for CSE 331: Fall 2019.

One
appointment/wk
for now

Instructions for booking appointments

Follow these instructions to book one-on-one appointments with a TA (for a slot of 30 minutes).

1. Go to the [course calendar](#) and search for a desired meeting time slot. You can only pick the office hours that are marked as 

Slots are starting off small

To start off with, we will have 30 slots per week. If this turns out to be popular, we will increase the number of one-on-one meeting slots later in the semester.

2. Then proceed to the [course appointment calendar](#) and select the time slot matching the desired time from the course calendar (from step 1). **Make sure to book the appointment using your buffalo.edu account.**
3. Click "Save" to book the appointment and remember to meet your TA on time! You'll also be able to view the appointment on your own google calendar now.



The screenshot shows a form titled "Book an appointment" with the following fields:

- My TA's name:** A text input field with "STANLEY" entered.
- My TA's location:** A text input field with "201" entered.
- My email:** A text input field with "stan" entered.
- My phone:** A text input field.
- Message:** A larger text area for additional information.

Makeup recitations

~~TODAY, 12-12:50pm in Davis 113A~~

TOMORROW, 11-11:50am in Davis 113A

Sign-up for mini projects

Deadline: Monday, Sep 23, 11:00am

CSE 331 [Syllabus](#) [Team/TA](#) [Piazza](#) [Schedule](#) [Homeworks](#) [Activities](#) **Mini Project** [Support Pages](#) [Channel](#)

CSE 331 Video Mir choices

Fall 2019

Please check the table below before submitting your mini project team composition to make sure your case study is not being used by another group. Case studies are assigned on a first come first serve basis.

Chosen Case Studies for Videos

Mini Project Details

Signup form

Questions/Comments?



Peer notetaker request

note 0

Cancel

Cancel

Peer notetaker request

Hi all,

Please see the message below from accessibility resources please do help out if you can. In addition to the contact information below, I believe you can also email access@buffalo.edu

If you do end up being a peer note taker, please let me know so that I can stop sending reminders in the future :)

Thank!

Megan

It studied in your CBE 351 class is eligible for the services of a Peer Notetaker. Notetakers provide an essential service that helps ensure equal access to education for students who receive accommodations. Students often find volunteering to be a Peer Notetaker enhances the classroom experience by encouraging more thorough, quality notes. Notetakers who qualify may receive a letter of recommendation or, if they qualify, an honorarium at the end of the semester.

If you are interested in becoming a Peer Notetaker for this course, please stop by our office as soon as possible. We are able to accept Notetakers on a first come, first serve basis.

Thank you in advance,

Megan Hughes
Access Support Coordinator
Accessibility Resources
65 Cooper Hall
University at Buffalo
Buffalo, NY 14260
(716) 643-2888
(716) 643-3118

[Cancel](#) [Cancel](#)

Incorrect Proof Details: Q1(b) on

HWO

Argument does not use ANYTHING about the problem statement!

Follows from part (a)

of perfect matchings with n men and n women.

Base case: $P(1) = 1! = 1$

This assumes number of perfect matchings only depends on n

Inductive hypothesis: Assume that $P(n-1) = (n-1)!$

Inductive step: Note that $P(n) = n * P(n-1) = n * (n-1)! = n!$

What are the issues with the above “proof”?

Incorrect Proof Details: Q1(b) on HWO

Needs justification

Claim 1: Number of perfect matchings is = number of permutations of $1\dots n$

Claim 2: Number of permutations of $1\dots n$ is $n!$

Needs justification

Claims 1 + 2 prove the result

Follow from 191 (?)

What are the issues with the above proof?

Proof by contradiction for Q1(a)

Assume for contradiction there is an example where number of perfect matchings depends on the identities of the men and women.

Let $n = 1$ and consider two cases

(1) $M = \{BP\}$ and $W = \{JA\}$

(2) $M = \{BBT\}$ and $W = \{AJ\}$

You can only assume things about the example directly implied by it being a counter-example

In both cases the number of perfect matchings is $1 = 1!$

Hence contradiction.

There is NO contradiction

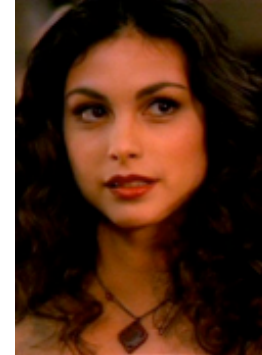
What are the issues with the above proof?

Questions/Comments?



On matchings

Mal



Inara

Wash



Zoe

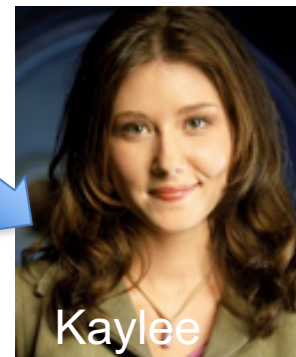
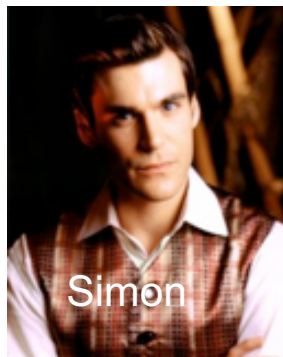
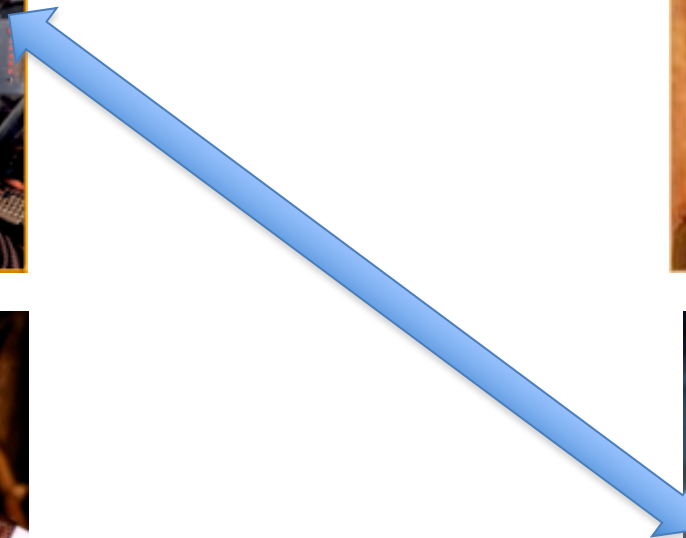
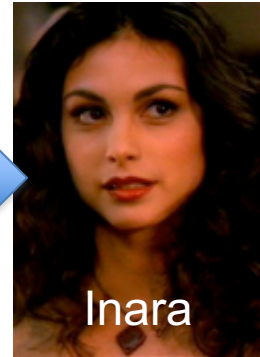
Simon



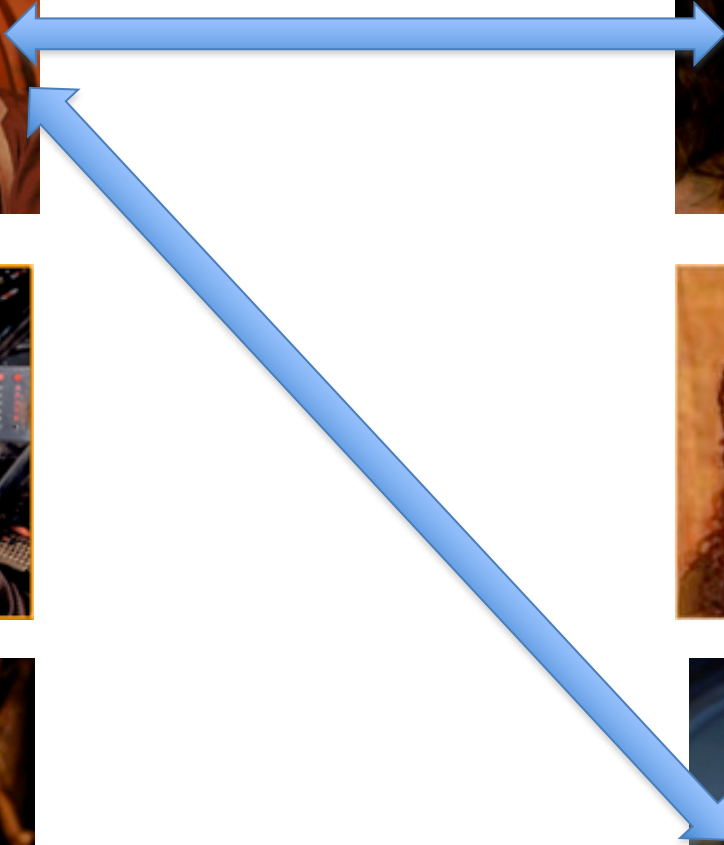
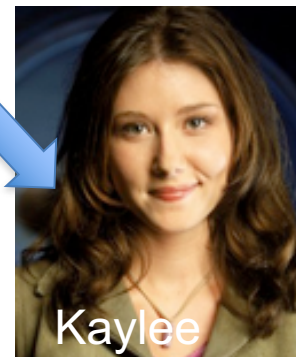
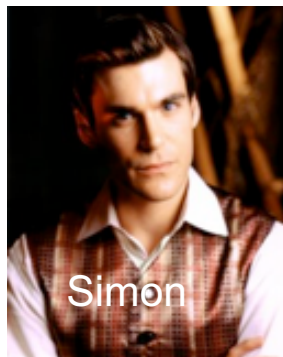
Kaylee



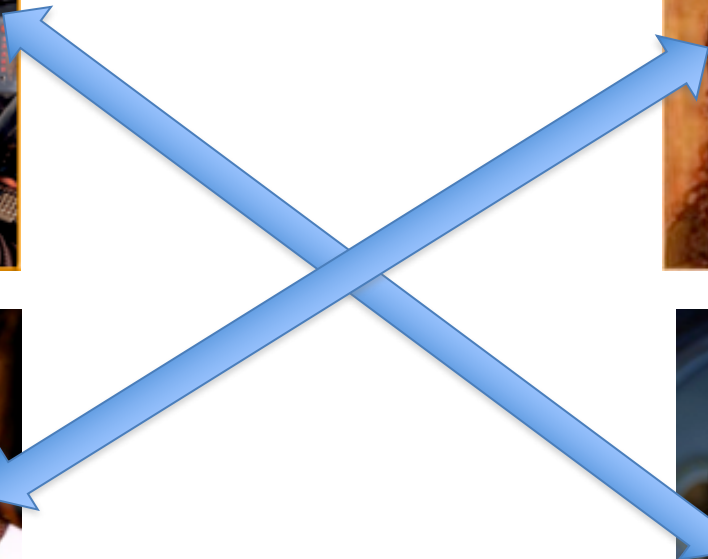
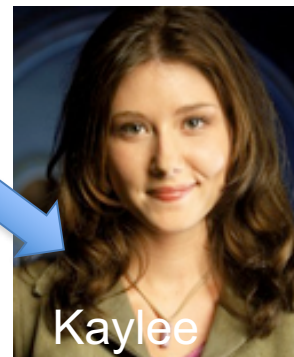
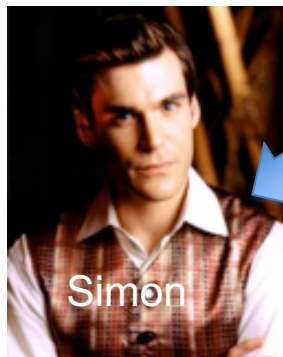
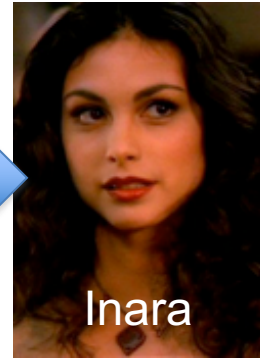
A valid matching



Not a matching

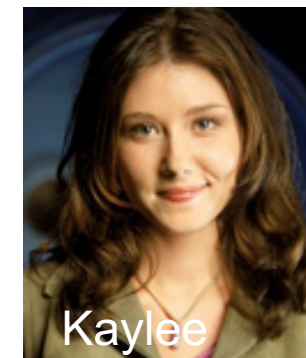
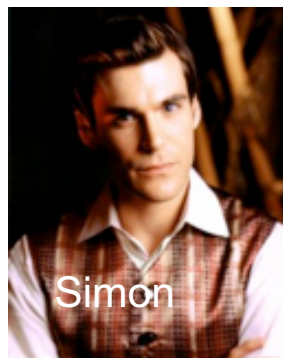
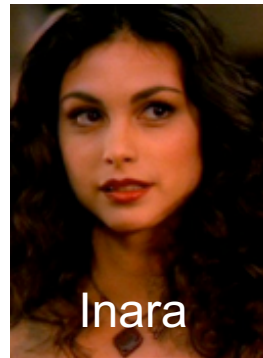


Perfect Matching

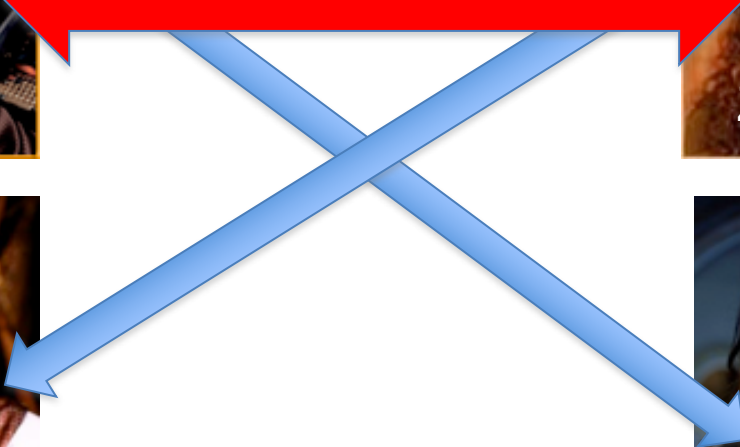
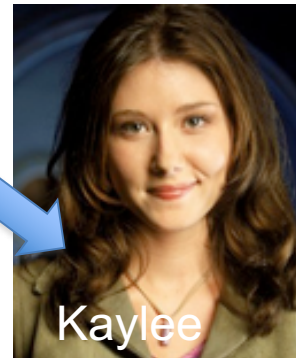
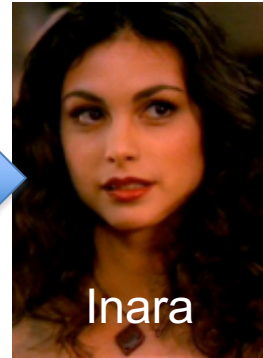


Back to couple more definitions

Preferences

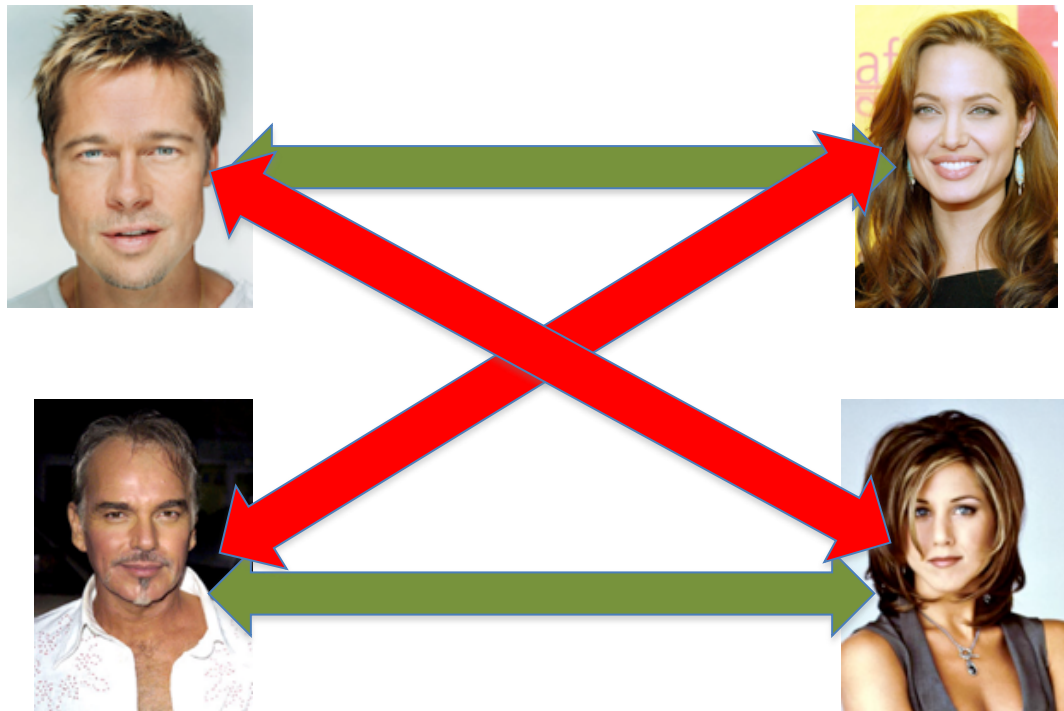


Instability

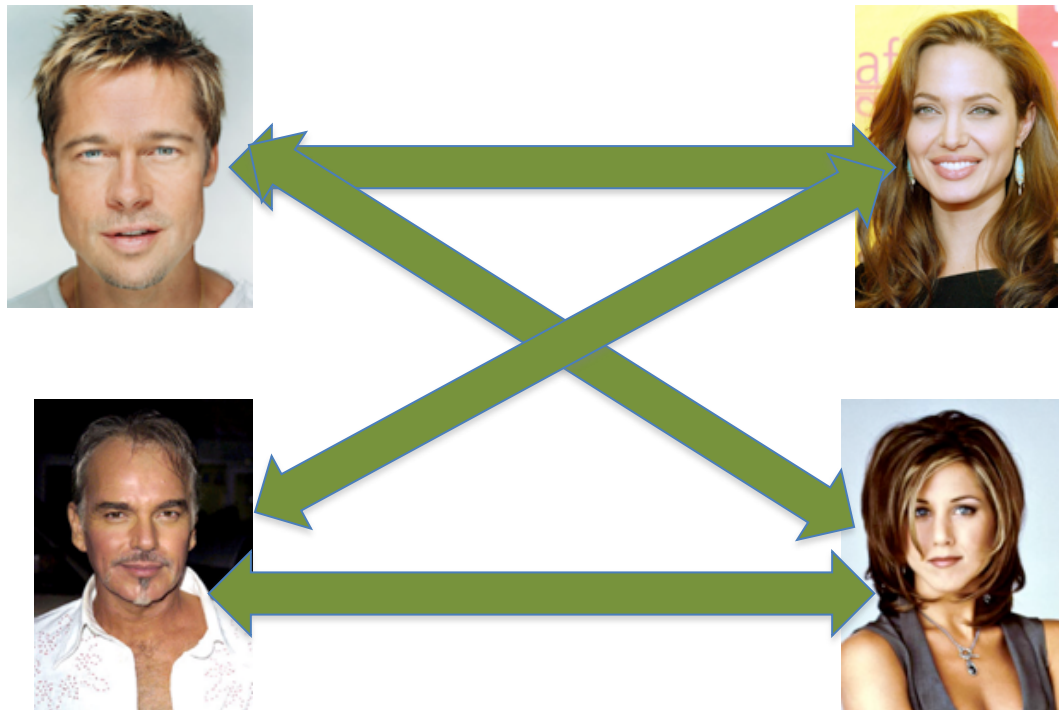


A stable marriage

Even though BBT and JA are not very happy



Two stable marriages



Stable Marriage problem

Set of men M and women W

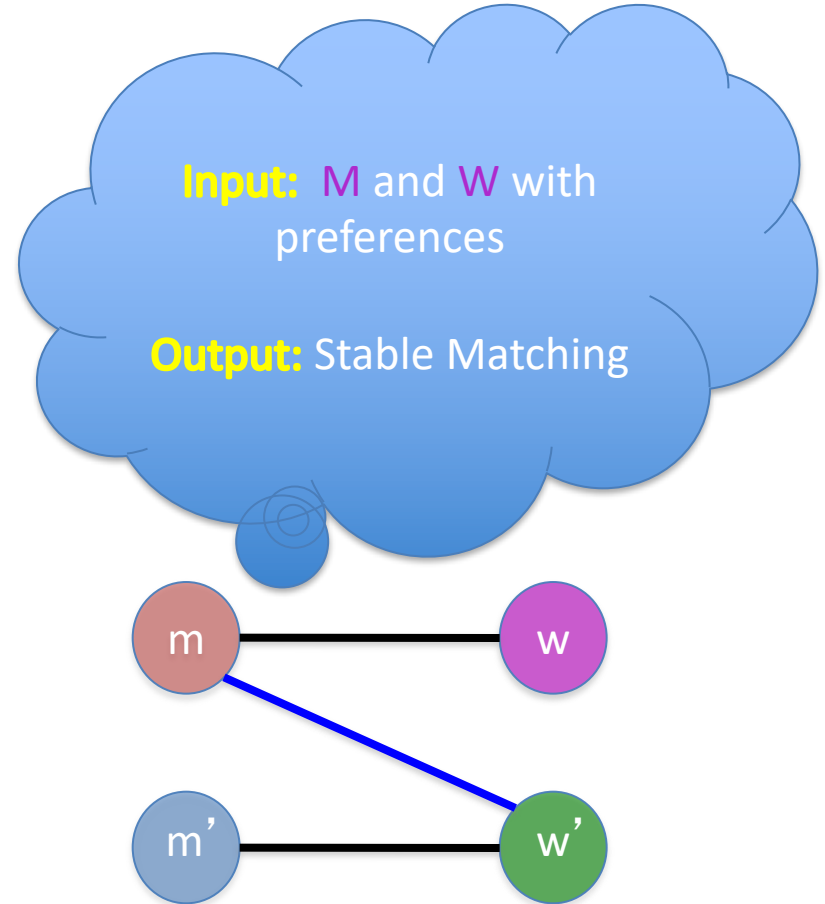
Preferences (ranking of potential spouses)

Matching (no polyandry/gamy in $M \times W$)

Perfect Matching (everyone gets married)

Instability

Stable matching = perfect matching + no instability



Questions/Comments?



Two Questions

Does a stable marriage always exist?

If one exists, how quickly can we compute one?

Today's lecture

Naïve algorithm

Gale-Shapley algorithm for Stable Marriage problem

Discuss: Naïve algorithm!



The naïve algorithm

Incremental algorithm to produce all $n!$ perfect matchings?

Go through all possible perfect matchings S

If S is a stable matching

then Stop



Else move to the next perfect matching

Gale-Shapley Algorithm



David Gale



Lloyd Shapley

$O(n^3)$ algorithm

Moral of the story...



Questions/Comments?

