

Lecture 4

CSE 331

Sep 4, 2019

Please have a face mask on

Masking requirement



UR requires all students, employees and visitors – regardless of their vaccination status – to wear face coverings while inside campus buildings.

<https://www.buffalo.edu/coronavirus/health-and-safety/health-safety-guidelines.html>

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



Syllabus Quiz (and sections)

note @64   stop following **37** views

Sections updated

If you scored at least 18 in your syllabus quiz before 8:00pm on Mon, Sep 8, your section on Autolab should have been updated to a **Y** (otherwise it should still say **N**). To receive graded material back, you should be in section **Y**.

To check your section, go to the CSE 331 page on Autolab, click your name on right top, then click *Course Profile*. It should show your section there.

If you had passed the quiz before 8:00pm on Mon but your section still says **N**, please let me know.

Otherwise, I will update the sections next weekend again, so please do fill in your syllabus quiz (and make sure you pass) if you have not done so already.

[edited](#)

[will](#) good note 0

Updated 18 hours ago by Ari Rautas

Separate Proof idea/proof details

<> Note

Notice how the solution below is divided into proof idea and proof details part. **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. (I'm presenting both the solutions but of course you only need to present one.)

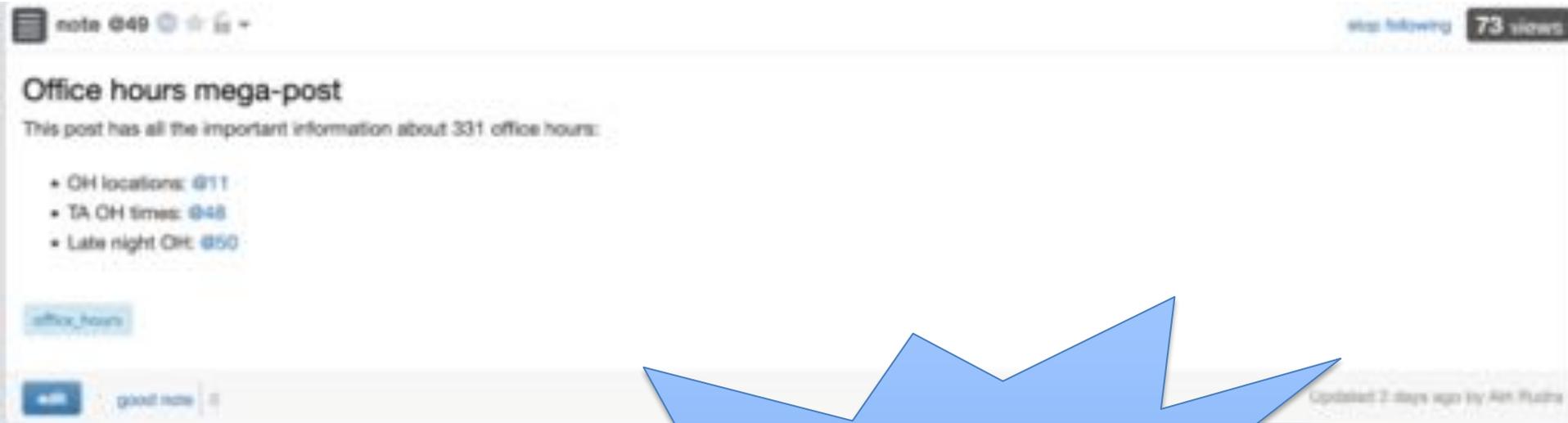
We begin with the approach of reducing the given problem to a problem you have seen earlier. \Rightarrow Build the following complete binary tree: every internal node in the tree represents a "parent" RapidGrower while its two children are the two RapidGrowers it divides itself into. After x seconds this tree will have height x and the number of RapidGrowers in the container after x seconds is the number of leaf nodes this 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 reduction. 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 reduction 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 assign the root of $T(0)$ 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 (cite your 191/250 book here with the exact place where one can find this 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.

Office hours finalized



note @49 stop following 73 views

Office hours mega-post

This post has all the important information about 331 office hours:

- OH locations: @11
- TA OH times: @48
- Late night OH: @50

office_hours

good note

Updated 2 days ago by Ash Rathi

Come ask your proof
related Questions!

1st True/False poll

poll @62    stop following 50 votes Actions

The first true/false question

The plan is to do a weekly True/false question on piazza. The way it is going to work is that every Monday (or so) I will post a statement in a poll and ask you guys to vote True or False. (Please just vote and do not post your justification: yet.) Then after two days, I will give the correct answer (and we will see how well crowd-sourcing works in this context) and then ask for you guys to construct the correct justification. Note that this is to give you guys more practice for the true/false questions on the exams (there will be pretty much no true/false questions on the homeworks). So try and work on these on your own so that you gain some practice.

Anyhow, here is the **question for this week**. Is the following statement **True** or **False**?

Given n numbers a_1, \dots, a_n such that for every $i \in [n]$ (we will use $[n]$ to denote the set of integers $\{1, \dots, n\}$) we have $a_i \in \{0, 1\}$. That is, we are given n numbers each of which is a bit. Then we can sort these n numbers in $O(n)$ time.

True

False

You have **not yet** voted.

Revoting is **not allowed**. Select your vote and click submit to register your vote.

Your name will **not be visible** to anyone.

Register your project groups

Deadline: Friday, Oct 1, 11:59pm

CSE 331 Syllabus Piazza Schedule Homeworks + Autolab **Project +** Support Pages + channel Sample Exams +

Project Overview

Group signup form

Forming groups

You form groups of size **exactly three (3)** for the project. Below are the various options.

- You have two choices in forming your group:
 1. You can form your group on your own: i.e. you can submit the list of **EXACTLY three (3)** groups members in your group.

Note

Note that if you pick this option, your group needs to have **exactly THREE (3)** members. In particular, if your group has only two members you cannot submit as a group of size two. If you do not know many people in class, feel free to use piazza to look for the third group member.

2. You can submit *just your name*, and you will be assigned a random group among all students who take this second option. However, **note that if you pick this option you could end up in a group of size 2**. There will be at most two groups of size 2.

Submitting your group composition

Use this [Google form](#) to submit your group composition (the form will allow you to pick one of the two options above).

- You need to fill in the form for group composition by **11:59pm on Friday, October 1**.

Deadline is strict!

If you do not submit the form for group composition by the deadline, then you get a **zero for the entire project**.

Piazza Qs on your code



The screenshot shows a Piazza note interface. At the top left, it says 'note @69' with icons for search, star, and lock. At the top right, it says 'stop following' and '7 views'. The main title is 'If you are asking questions about issues with your code'. Below the title, it says 'At the minimum, please let us know:'. There are two bullet points: '• What language you are using?' and '• If you are getting a compile or runtime error: where are you running your code? Is it on Autolab or is it on your machine?' with a sub-bullet: '- If it's on your machine and you are using C++, please let us know if you are using a VM or not (if you are using an IDE we'll probably not be able to help much...)'.

At the bottom left, there is a 'homework' tag and a 'good note' button. At the bottom right, it says 'Updated 4 minutes ago by Adri Ruda'.

! If you do not follow one of our recommended C++ setups, you are on your own

We present three options for you to code in C++. You are of course welcome to use your own system **but if you do so, we will not be able to provide ANY help.**

In previous years students have reported that our C++ template code (as is) would not run on their own C++ setup (typically an IDE). If this happens we cannot help you figure out how to modify the template code on your machine.

More on coding questions

note @75

stop following 0 viewers

Actions

Few comments/reminders on programming submissions

I have mentioned the following to some of y'all in private posts so figured should post this here in case it is useful to some of you:

- Note that you cannot just use any source for programming. See here for the allowed sources— <http://www-student.cse.buffalo.edu/~str/cse331/fall21/policies/allowed-sources.html>

In particular, there is one official approved source for each language. However, there is a way to get unapproved sources "approved". Here is the relevant bit from the HW policy document:

? In case you are not sure

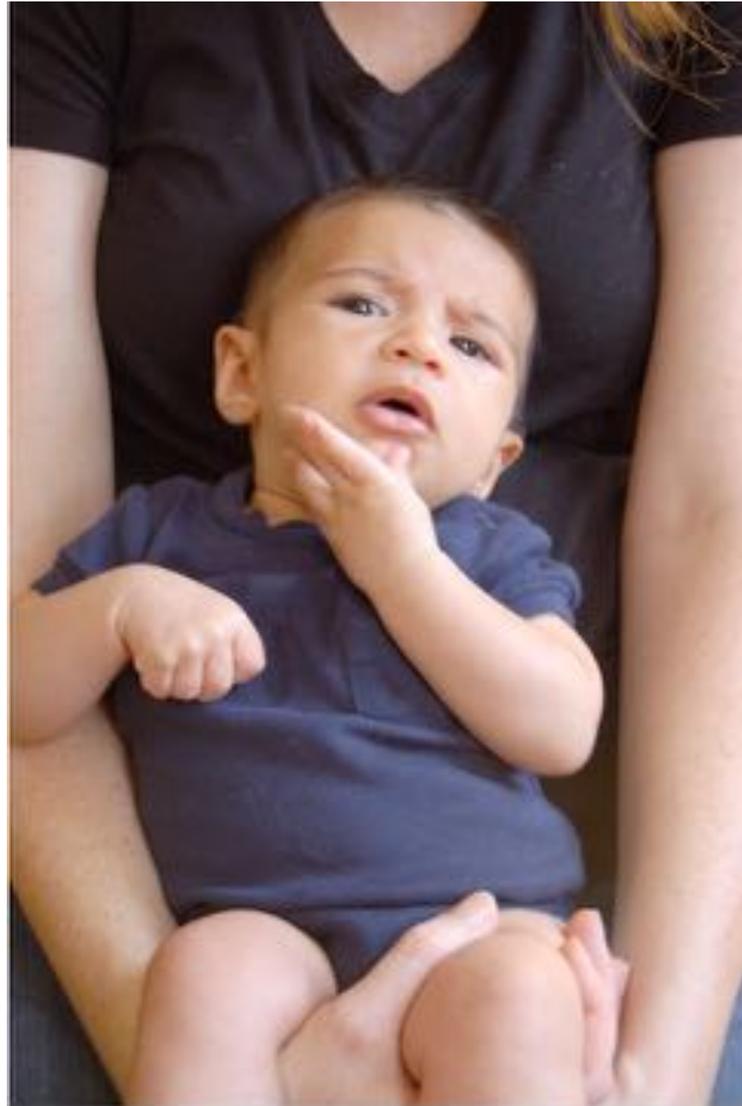
If you are not sure if you consulted with a source or someone that was not allowed, please check with the instructor before submitting your homework. If the instructor thinks that there was no inappropriate use of sources or collaboration, then you can go ahead and submit your homework. Otherwise you can just not submit your homework **without incurring any penalty**. Note that it is perfectly OK to get a source **officially approved** by sending the instructor a private post on piazza. If approved, the instructor will make the post public on piazza and then it officially becomes an allowed source. If not, then just don't submit your homework.

- If you are using a link that allows you to say figure out some basic programming stuff that I'll definitely approve (unless for some reason it gives the answer for the corresponding programming question), so please do use the above option.
- The [Autolab page](#) should have most of the answer you might have. In particular, the two most common errors that students get (other than a compilation error for their code) are explained in the page:

Dealing with Errors

- **Unexpected token error**: This usually happens when your code exceeds the global 180 second time limit for all testcases. Look for **Autodriver: Job timed out after 180 seconds** in the feedback. If you don't see this as part of the feedback please contact the course staff.
- **UnicodeDecodeError**: This is most likely because you have one or more non-ASCII characters in your code. First double check that you submitted a source file and not a binary. If the error persists, run [this Python script](#) on your Solution file by executing `python3 validate_ascii.py <filename>`. This will print the line numbers with non-ASCII characters in your code, which you should then remove.
- BTW if you are used to using a visual debugger but are programming from command line (e.g. using a VM for C++ code), I find that for code that y'all have to submit for 331, putting in appropriate print statement (and using binary search) is pretty effective in debugging from the command line.

Questions/Comments?



Solutions to HW 0 out

note @74   

stop following **9** views

Solutions to HW 0

have been posted: <http://www-student.cse.buffalo.edu/~atri/cse331/fall21/tws/tw0/soln.html>

Please pay attention to the note on the top of the solutions about future HW solutions.

homework0

edit good note | 0

Updated 11 minutes ago by Atri Rudra

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

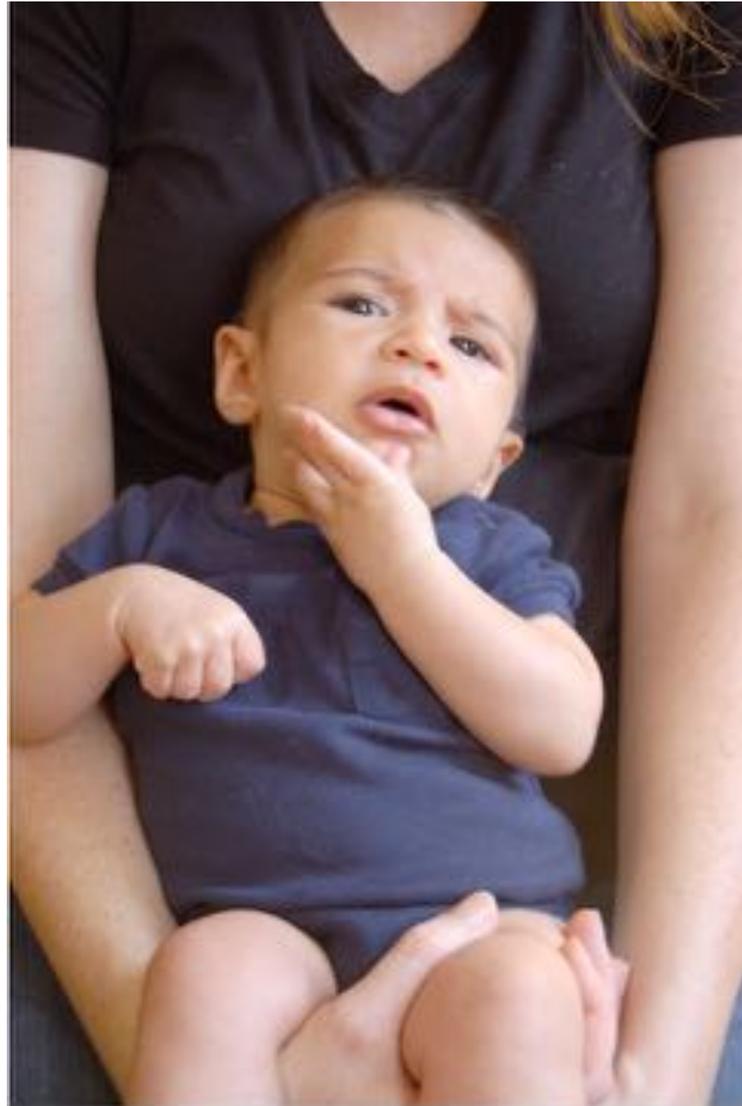
In both cases the number of perfect matchings is $1 = 1!$

Hence contradiction.

There is NO contradiction

What are the issues with the above proof?

Questions/Comments?



Questions to think about

1) How do we specify preferences?

Preference lists

2) Ratio of applicant vs employers

1:1

3) Formally what is an assignment?

(perfect) matching

4) Can an employer get assigned > 1 applicant?

NO

5) Can an applicant have > 1 job?

NO

6) How many employer/applicants in an applicants/employers preferences?

All of them

7) Can an employer have 0 assigned applicants?

NO

8) Can an applicant have 0 jobs?

NO

On matchings

Mal



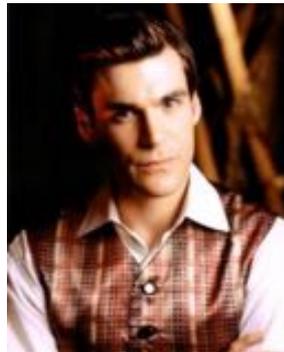
Inara

Wash



Zoe

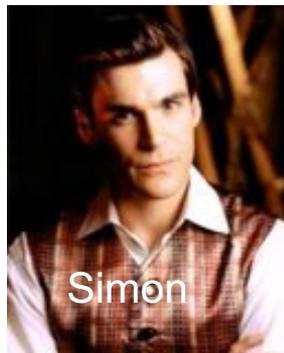
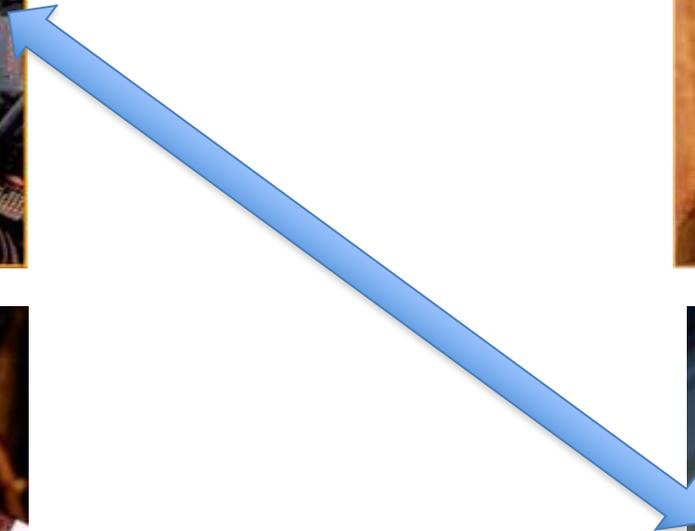
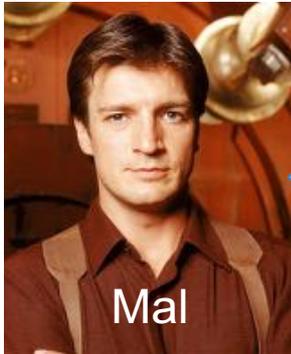
Simon



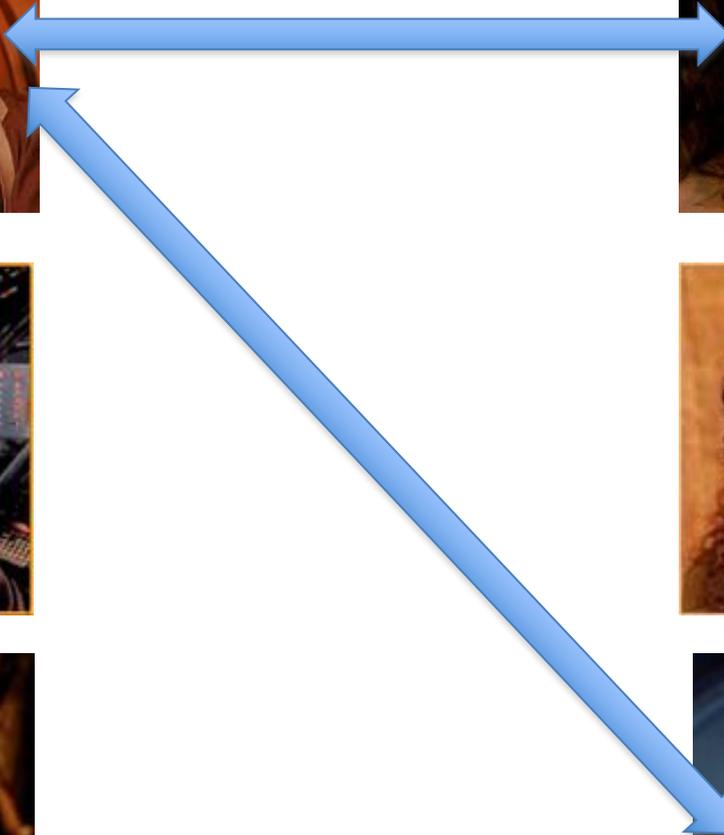
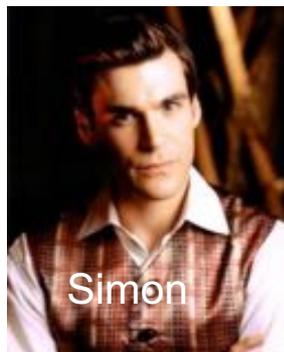
Kaylee



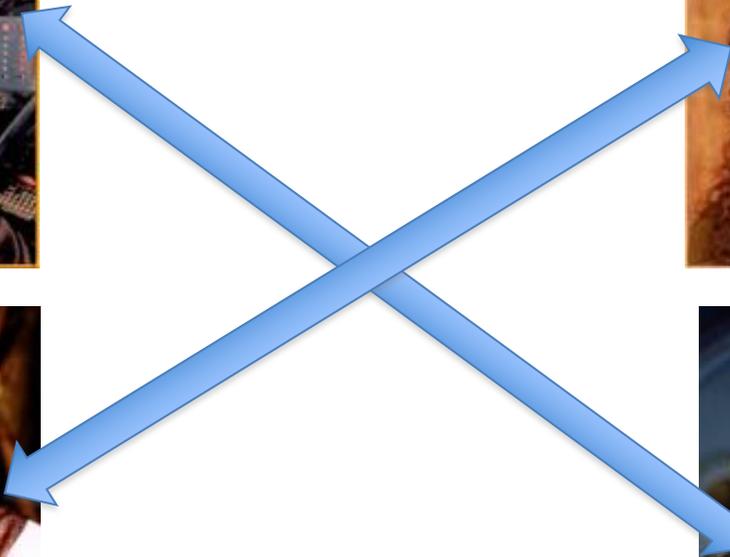
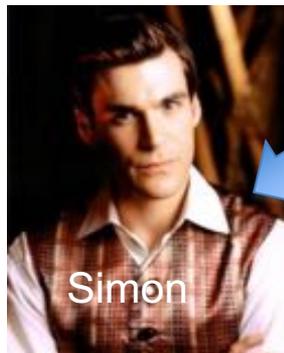
A valid matching



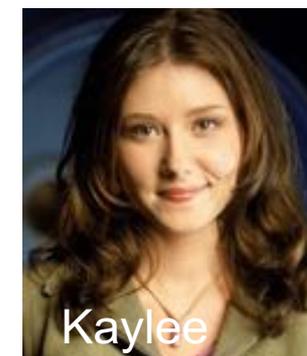
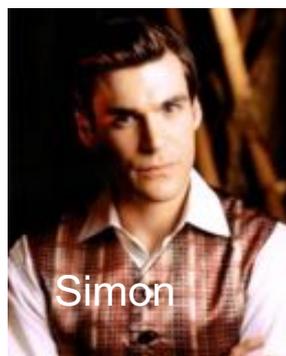
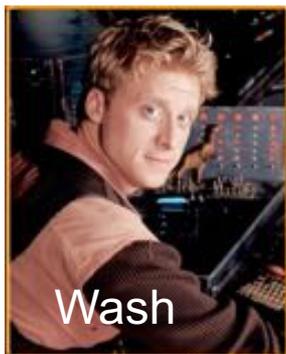
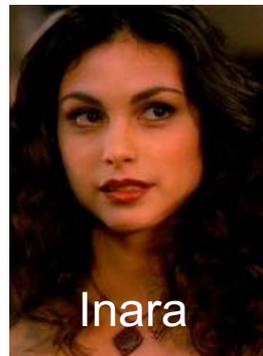
Not a matching



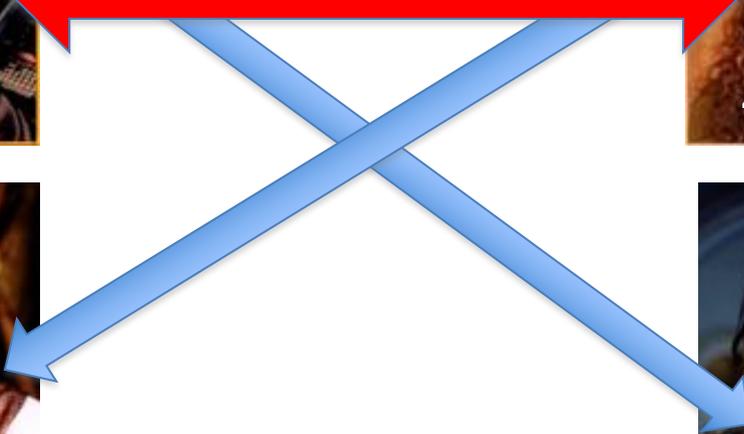
Perfect Matching



Preferences



Instability



Back to the board...

