Lecture 8

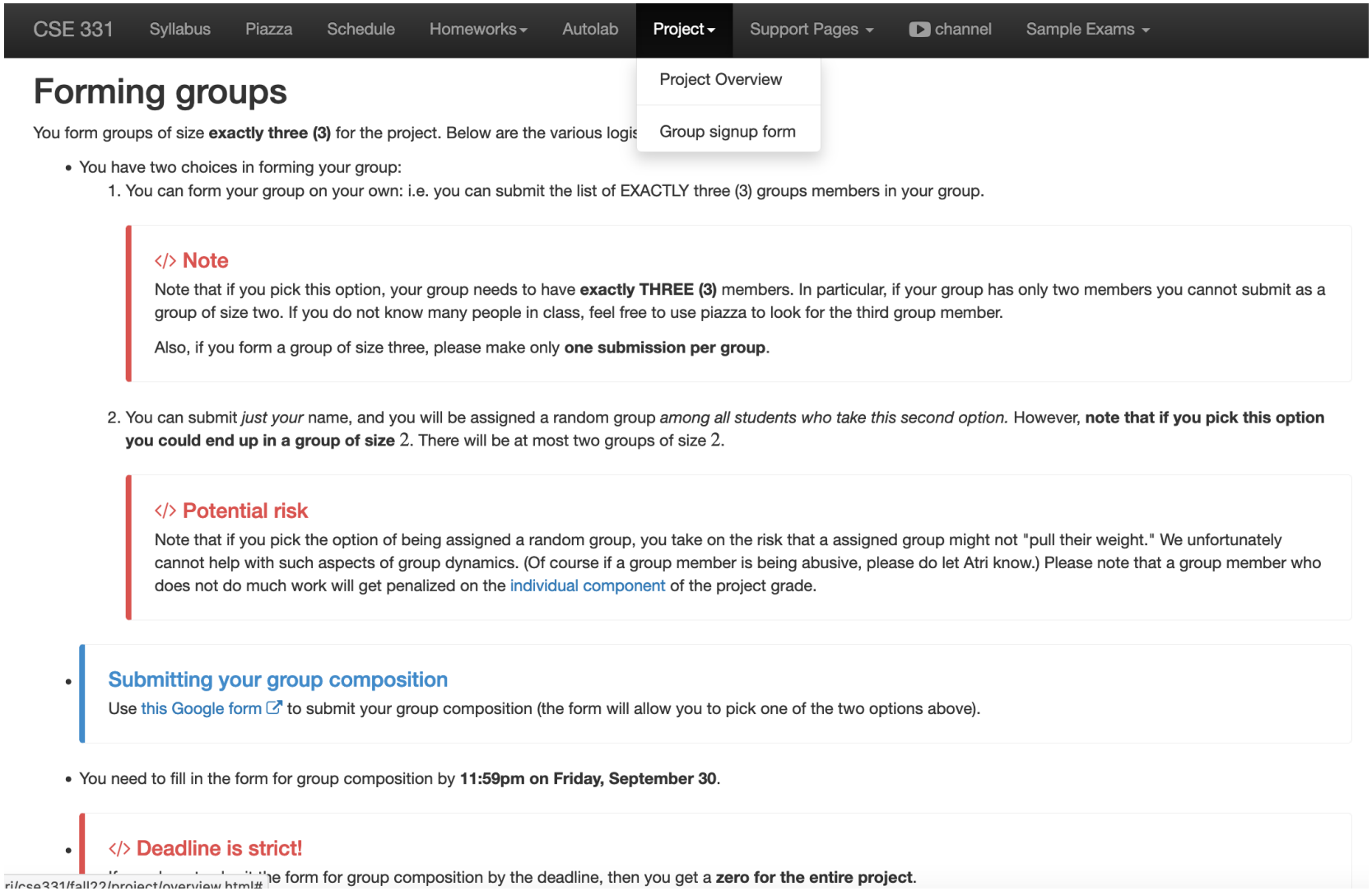
CSE 331

Sep 15, 2023

If you need it, ask for help

Register your project groups

Deadline: Friday, Sep 29, 11:59pm



Your UBIT ID is

xyz if your email ID is xyz@buffalo.edu

NOT

xyz@buffalo.edu

Your UB person number

Follow ALL instructions on HW1

! Submit part (a) and (b) separately

You need to submit **two (2) PDF** files to Autolab: one for part `(a)` and one for part `(b)`. While you can assume part `(a)` as a given for part `(b)`, to get credit for part `(a)` you have to submit your solution for part `(a)` separately from part `(b)`.

Make sure you submit the correct PDF to the correct submission link on Autolab. If you do not (e.g. if you submit Q1(a) PDF to Q1(b)), then you will lose ALL points.

We recommend that you typeset your solution but we will accept scans of handwritten solution-- you have to make sure that the scan is legible.

! PDF only please

If Autolab cannot display your file, (irrespective of the reason) then you will get a zero (0) on the entire question.

Autolab might not be able to display files in formats other than PDF (e.g. Word cannot be displayed). **Note that Autolab will "accept" your submission even if you submit non-PDF file, so it is YOUR responsibility to make sure you submit in the correct format.**

Also the file size has to be **at most 3MB**.

Grading Guidelines

We will follow the usual grading guidelines for non-programming questions. Here is a high level grading rubric specific to part `(a)` of this problem:

1. `Proof idea`: `10` points.

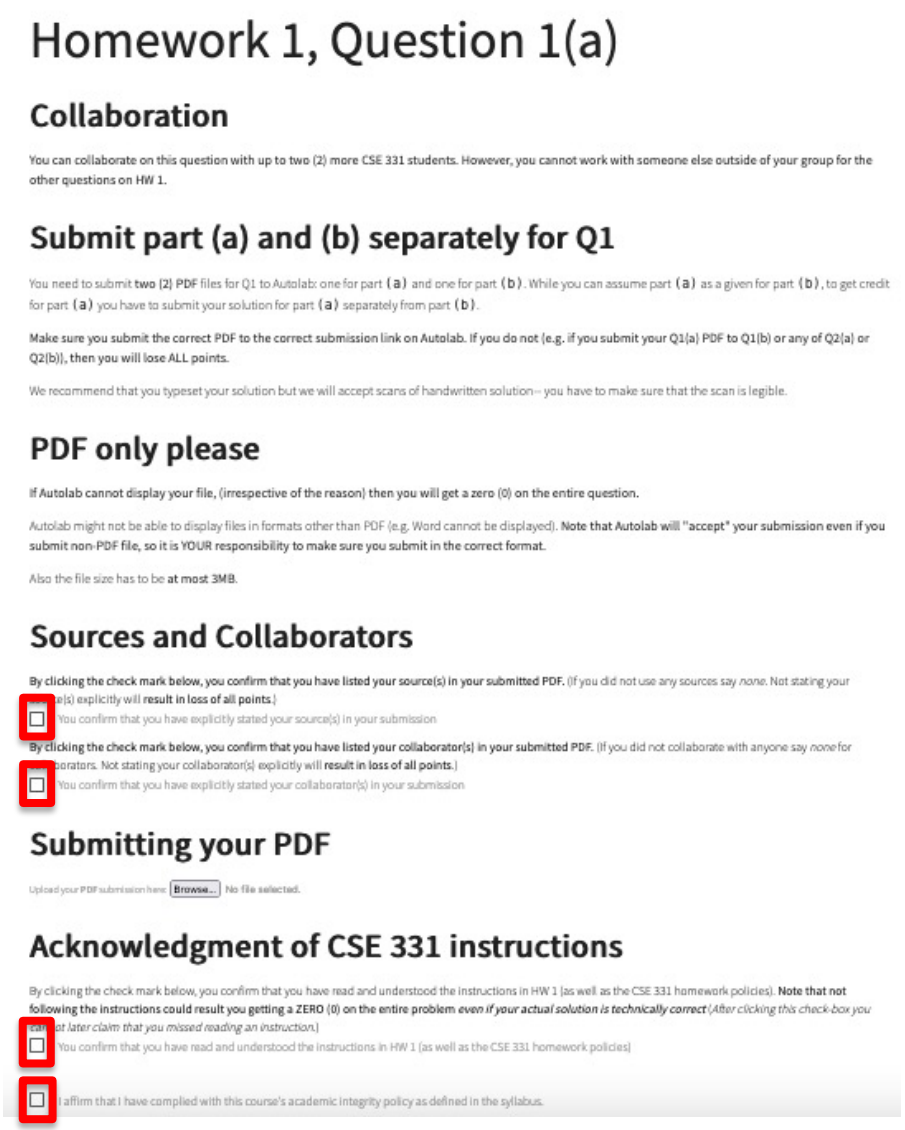
and here is the high level grading rubric for part `(b)`:

1. `Proof/Algorithm idea`: `20` points for
   If your answer is yes: the idea behind the algorithm that for any input, computes a pair of stable schedules.
   - *If your answer is no*: a counterexample idea explaining the insight behind why you think the property does not hold.
2. `Proof/Algorithm details`: `20` points for
   If your answer is yes: details of the algorithm that for any input, computes a pair of stable schedules *and* an argument as to why your algorithm will always output a pair of stable schedules for every input.
   - *If your answer is no*: a complete description of a counterexample *and* a complete proof for why the given counter example does not have any stable schedule.

! Note

If you do not have separated out proof/algorithm idea and proof/algorithm details for part `(b)`, you will get a zero (0) irrespective of the technical correctness of your solution.

They are repeated on Autolab



Homework 1, Question 1(a)

Collaboration

You can collaborate on this question with up to two (2) more CSE 331 students. However, you cannot work with someone else outside of your group for the other questions on HW 1.

Submit part (a) and (b) separately for Q1

You need to submit **two (2) PDF** files for Q1 to Autolab: one for part **(a)** and one for part **(b)**. While you can assume part **(a)** as a given for part **(b)**, to get credit for part **(a)** you have to submit your solution for part **(a)** separately from part **(b)**.

Make sure you submit the correct PDF to the correct submission link on Autolab. If you do not (e.g. if you submit your Q1(a) PDF to Q1(b) or any of Q2(a) or Q2(b)), then you will lose ALL points.

We recommend that you typeset your solution but we will accept scans of handwritten solution -- you have to make sure that the scan is legible.

PDF only please

If Autolab cannot display your file, (irrespective of the reason) then you will get a zero (0) on the entire question.

Autolab might not be able to display files in formats other than PDF (e.g. Word cannot be displayed). **Note that Autolab will "accept" your submission even if you submit non-PDF file, so it is YOUR responsibility to make sure you submit in the correct format.**

Also the file size has to be **at most 3MB**.

Sources and Collaborators

By clicking the check mark below, you confirm that you have listed your source(s) in your submitted PDF. (If you did not use any sources say *none*. Not stating your source(s) explicitly will **result in loss of all points**.)

☐ You confirm that you have explicitly stated your source(s) in your submission

By clicking the check mark below, you confirm that you have listed your collaborator(s) in your submitted PDF. (If you did not collaborate with anyone say *none* for collaborators. Not stating your collaborator(s) explicitly will **result in loss of all points**.)

☐ You confirm that you have explicitly stated your collaborator(s) in your submission

Submitting your PDF

Upload your PDF submission here: Browse... No file selected.

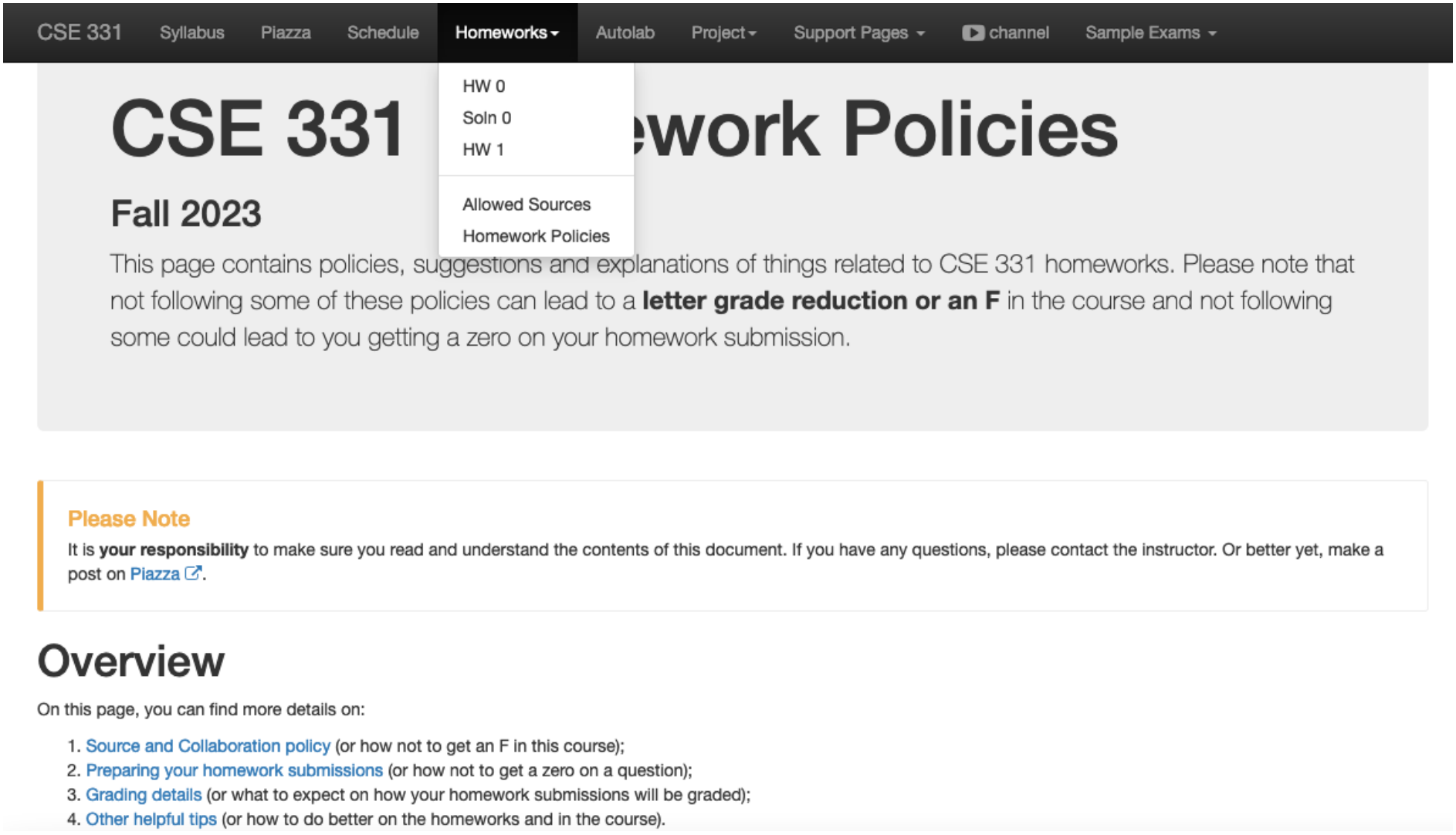
Acknowledgment of CSE 331 instructions

By clicking the check mark below, you confirm that you have read and understood the instructions in HW 1 (as well as the CSE 331 homework policies). **Note that not following the instructions could result you getting a ZERO (0) on the entire problem *even if your actual solution is technically correct*** (*After clicking this check-box you cannot later claim that you missed reading an instruction.*)

☐ You confirm that you have read and understood the instructions in HW 1 (as well as the CSE 331 homework policies)

☐ I affirm that I have complied with this course's academic integrity policy as defined in the syllabus.

Review the HW policy doc!



HW 1 (pre)post-mortem

note @97 stop following 83 views Actions

(Advance view of) post-mortem on Homework 1

The post below is from Fall 2019 that I posted *after* HW1 was due but I figured I should post it a bit earlier this time in case it is helpful to some of you as y'all work on your HW 1 submissions.

Of course this would depend pretty much on you as an individual but here are some questions, **in no particular order**, for y'all to ponder on (with some of our comments in *italics*):

- **Did you start early enough?**
  - *We recommend that you start working on the homework on Wednesday immediately after the homework is handed out itself. And distribute your hours over the week rather than wait to start till Monday (or gasp! Tuesday).*
- **Did you go to the recitations AND read the recitation notes?**
  - *Both of them help you a lot towards answering Q1(a) and Q2(a) so they are highly recommended.*
- **Did you work on the questions in correct order?**
  - *We have the current order based on what we think is most beneficial to you. In particular, we want y'all to focus more on the proof based questions, which is why they come before the programming question. But perhaps a different order would work better for you?*
- **Did you get help when you got stuck?**
  - *If you were stuck at a problem for a long time did you ask for help on piazza? Did you go to one of the office hours?*
- **Did you work on all the problems alone?**
  - *While working on all the problem by yourself will be good for you in the long run (since you are developing your proofs/algorithms skills), in the interest of time we recommend that you at least collaborate on Q2 (b).*
- **(If you submitted HW 0), did you get enough feedback?**

Questions/Comments?

Gale-Shapley Algorithm

Intially all men and women are free

While there exists a free woman who can propose

Let w be such a woman and m be the best man she has not proposed to

w proposes to m

If m is free

(m,w) get engaged

Else (m,w') are engaged

If m prefers w' to w

w remains free

Else

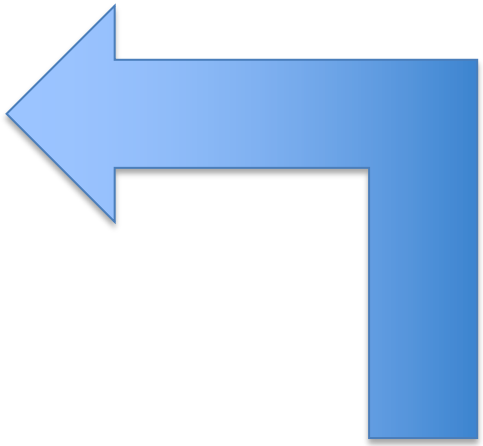
(m,w) get engaged and w' is free

Output the engaged pairs as the final output

The Lemmas

Lemma 0: The GS algorithm has at most n^2 iterations

Lemma 1: S is a perfect matching



Lemma 2: S has no instability

Lemma 3: If at the end of an iteration, w is free then w has not proposed to ALL men

Proof Details of Lemma 0

Gale Shapley algorithm terminates

This page collects material from Fall 17 incarnation of CSE 331, where we proof details for the claim that the Gale-Shapley algorithm terminates in $O(n^2)$ iterations.

Where does the textbook talk about this?

`Section 1.1` in the textbook has the argument (though not in as much detail as below).

Fall 2017 material

Here is the lecture video (it starts from the part where we d the proof details):

Proof technique de jour

Proof by contradiction

Assume the negation of what you want to prove

After some reasoning

Source: 4simpsons.wordpress.com

Two obervations

Obs 1: Once m is engaged he keeps getting engaged to "better" women

Obs 2: If w proposes to m' first and then to m (or never proposes to m) then she prefers m' to m

Questions/Comments?

Proof of Lemma 3 on the board…

Proof of Lemma 2

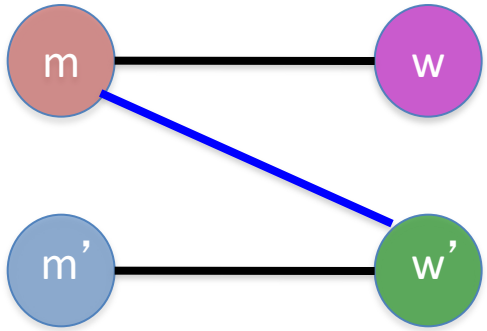
By contradiction

Assume there is an instability (m,w')

w' last proposed to m'

m prefers w' to w

w' prefers m to m'



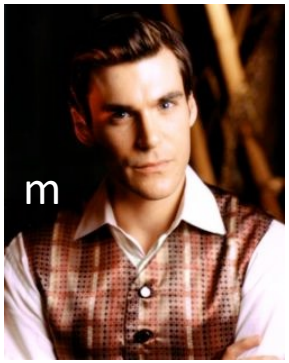
Contradiction by Case Analysis

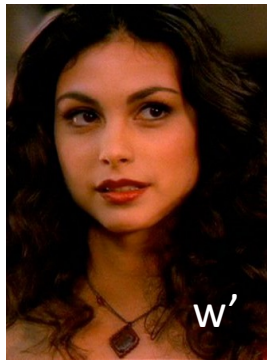
Depending on whether w’ had proposed to m or not

Case 1: w’ never proposed to m

w’ prefers m’ to m — By Obs 2

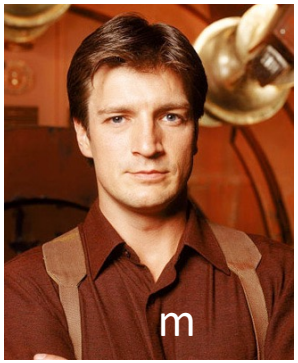
Assumed w’ prefers m to m’

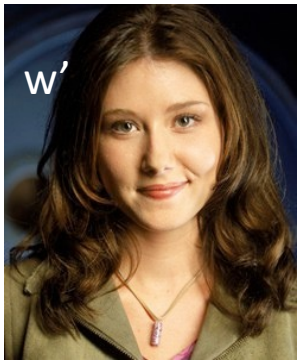




Source: 4simpsons.wordpress.com

Case 2: w’ had proposed to m





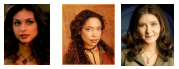
Case 2.1: m had accepted w’ proposal

m is finally engaged to w

Thus, m prefers w to w’

4simpsons.wordpress.com

By Obs 1



Case 2.2: m had rejected w’ proposal

m was engaged to w’’ (prefers w’’ to w’)

By Algo def

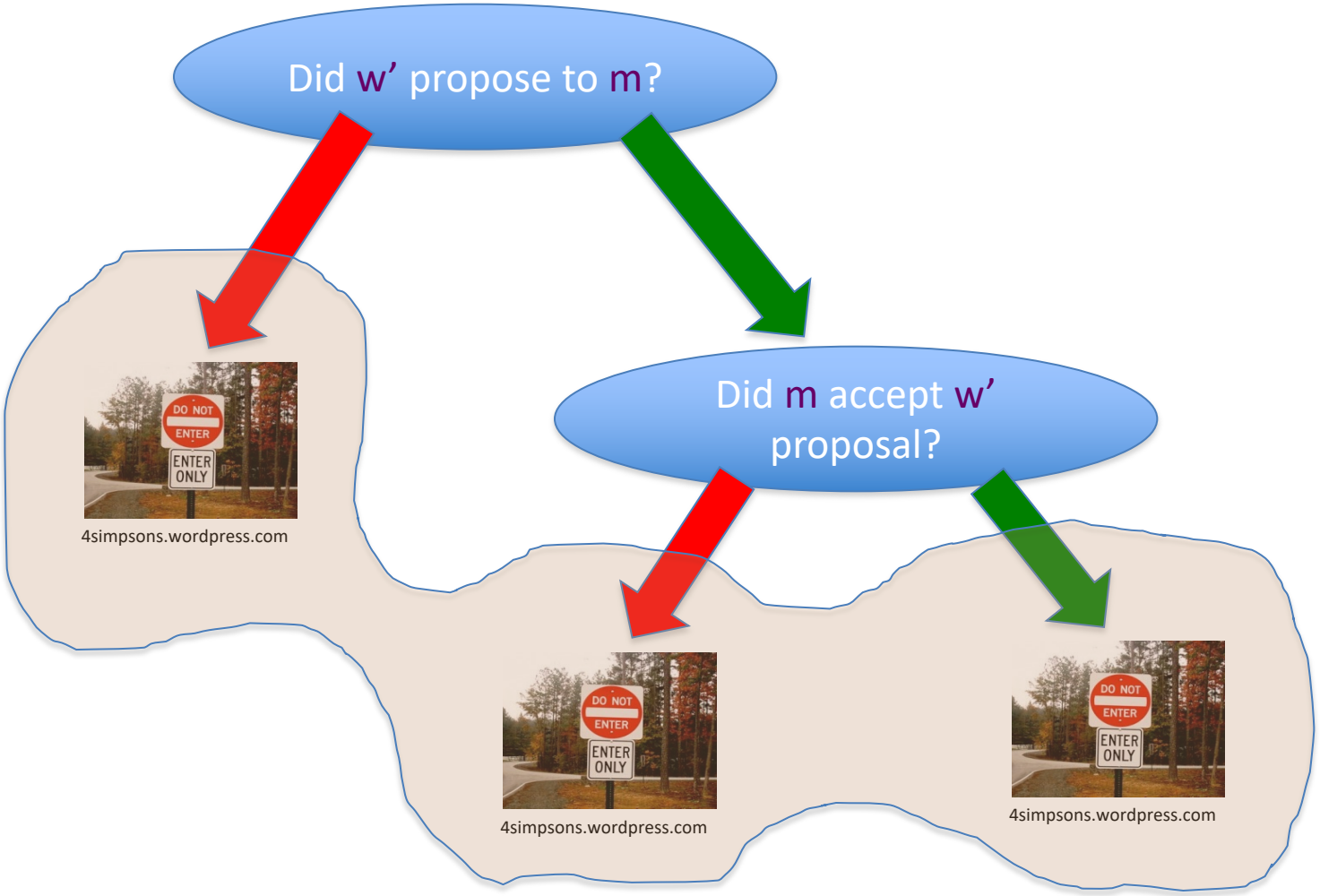
m is finally engaged to w (prefers w to w’’)

By Obs 1

m prefers w to w’

4simpsons.wordpress.com

Overall structure of case analysis



Questions?

Extensions

Fairness of the GS algorithm

Different executions of the GS algorithm

Main Steps in Algorithm Design

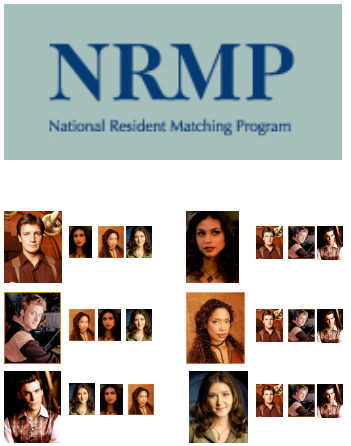
Problem Statement

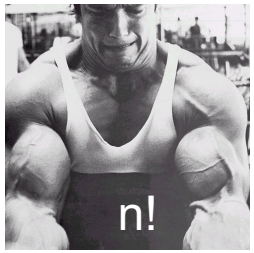
Problem Definition

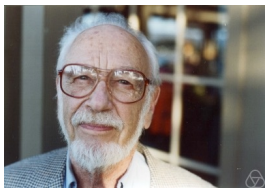
Algorithm

"Implementation"

Analysis







Correctness Analysis

Definition of Efficiency

An algorithm is efficient if, when implemented, it runs quickly on real instances

Implemented where?

Platform independent definition

What are real instances?

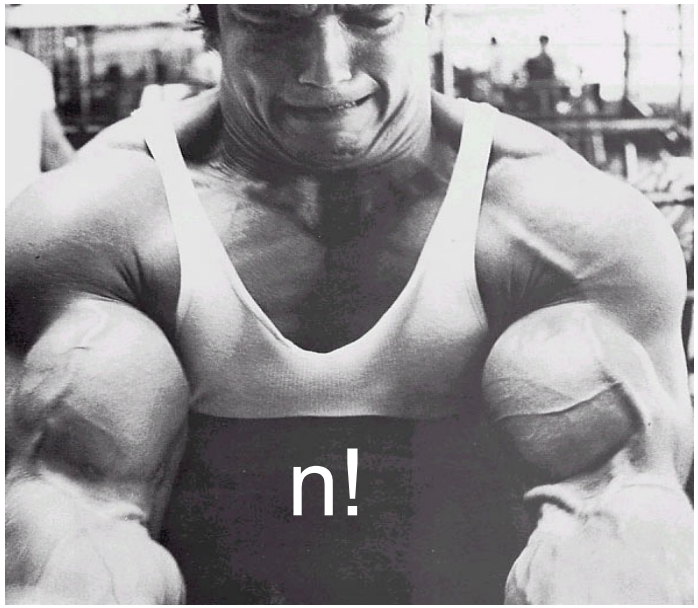
Worst-case Inputs

Efficient in terms of what?

$N = 2n^2$ for SMP

Input size N

Definition-II



Analytically better than brute force

How much better? By a factor of 2?

Definition-III

Should scale with input size

If N increases by a constant factor, so should the measure

Polynomial running time

At most $c \cdot N^d$ steps ($c>0$, $d>0$ absolute constants)

Step: “primitive computational step”

More on polynomial time

Problem centric tractability

Can talk about problems that are not efficient!