

Lecture 19

CSE 331

Oct 8, 2014

Group/Algo registration deadline

BOTH TONIGHT by 11:59pm!

note ☆ 1 views

You need to form a group of size EXACTLY 6 for mini project Actions ▾

A gentle reminder that the deadline to submit your group composition and your algorithm choice is due in about 1.5 weeks (Wednesday, October 8).

Note that you need to form groups of size EXACTLY SIX. In particular, if you have a group with <6 or >6 members, then you have missed your deadline and will lose points.

Forming groups of size 6 might take some time so if you have not started on this, I strongly suggest you start immediately to avoid losing the points for the mini project. These points will be some of the easiest ones in the course so do not lose them due to procrastination.

A related post that might be of interest: @37

More details on the mini project: <http://www.cse.buffalo.edu/~atr/courses/331/handouts/mini-project.pdf>

mini_project

edit good note 0 Just now by Anil Rudra

Online OH 9-10pm tonight

 note 

Online OH #4 tomorrow at 9pm

Any and all 331 related questions are welcome. Please use the folder 'onlineoh4' to tag your questions.

[onlineoh4](#)

  | 0

Just now by Abri Rudra

0 views
Actions 

Feedback

note stop following 1 views Actions

Feedback for the course

Please give some feedback on how you think 331 is working for you:

<https://www.surveymonkey.com/r/57NTXSD>

As usual, giving feedback is voluntary and anonymous.

I encourage you to do so especially if you think something is not working well (and/or something is working well), so that I can try and make changes (and/or make sure I continue the good things respectively). I am especially interested in hearing any suggestion you might have so that I can improve. Over the years I have made many changes based on such feedback (e.g. using the microphone and the document camera as well as scanning notes from class).

Also I apologize for not keeping track of a couple of feedbacks that you guys gave in the last couple of weeks (for some reason survey monkey stopped sending me notification when new feedback came in. I hope to have something to say on those soon.

#pin

grading lectures recitation

edit good note 0 2 minutes ago by Adri Fudra

Interval Scheduling Problem

Input: n intervals $[s(i), f(i)]$ for $1 \leq i \leq n$

Output: A *schedule* S of the n intervals

No two intervals in S conflict

$|S|$ is maximized

Analyzing the algorithm

R : set of requests

Set A to be the empty set

While R is not empty

 Choose i in R with the earliest finish time

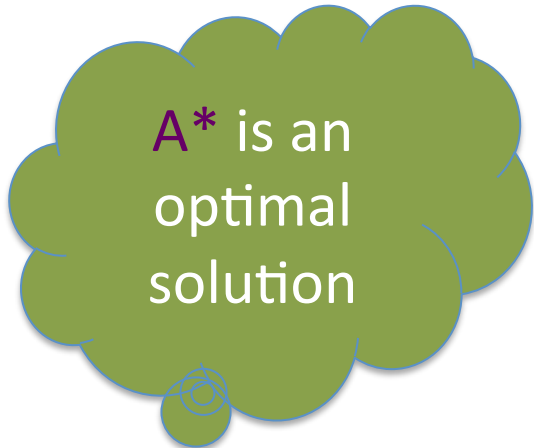
 Add i to A

 Remove all requests that conflict with i from R

Return $A^* = A$



A^* has no conflicts



A^* is an optimal solution

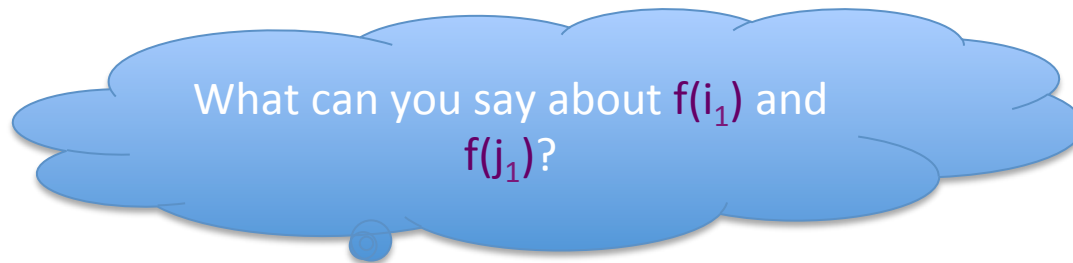
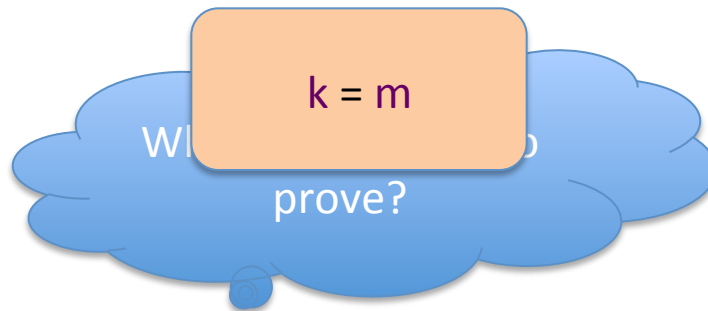
Greedy “stays ahead”



Greedy “stays ahead”

$$A^* = i_1, \dots, i_k$$

$$O = j_1, \dots, j_m$$



A formal lemma

For every $r \leq k$, $f(i_r) \leq f(j_r)$

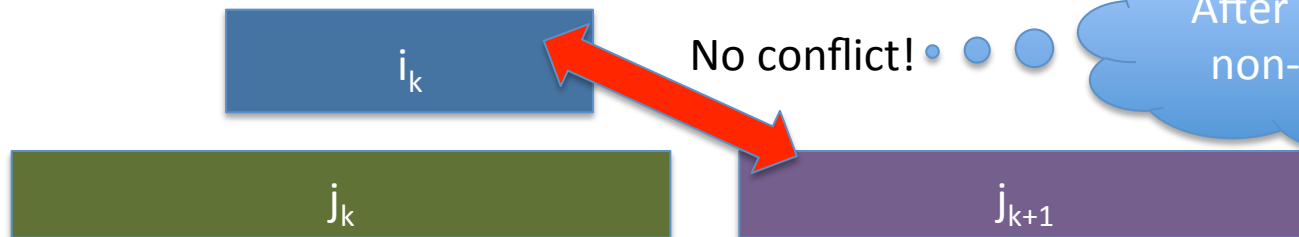
The greedy algorithm outputs an optimal A

Proof by contradiction: A is not optimal

$$A^* = i_1, \dots, i_k$$

$$O = j_1, \dots, j_m$$

$$m > k$$



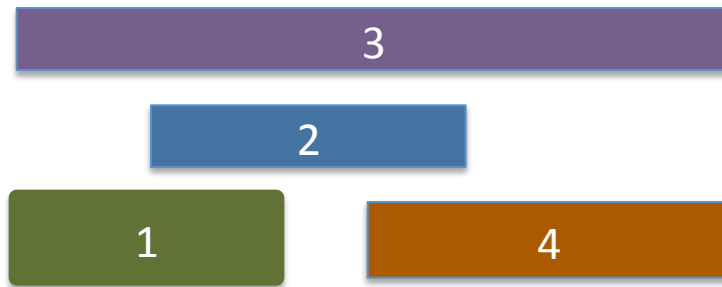
Today's agenda

Prove the lemma

Analyze run-time of the greedy algorithm

Algorithm implementation

Go through the intervals in order of their finish time



Check if $s[i] < f(1)$

with 1:

In general, if j th interval is the last one chosen

Pick smallest $i > j$ such that $s[i] \geq f(j)$. . .

$O(n \log n)$ run
time

The final algo

$O(n \log n)$ time sort intervals such that $f(i) \leq f(i+1)$

$O(n)$ time build array $s[1..n]$ s.t. $s[i]$ = start time for i

Add 1 to A and set $f = f(1)$

For $i = 2 .. n$

 If $s[i] \geq f$

 Add i to A

 Set $f = f(i)$

Return $A^* = A$

Reading Assignment

Sec 4.1of [KT]



Questions?



The “real” end of Semester blues

There are deadlines and durations of tasks



Write up a term paper

Party!

Exam study

331 HW

Project

Monday

Tuesday

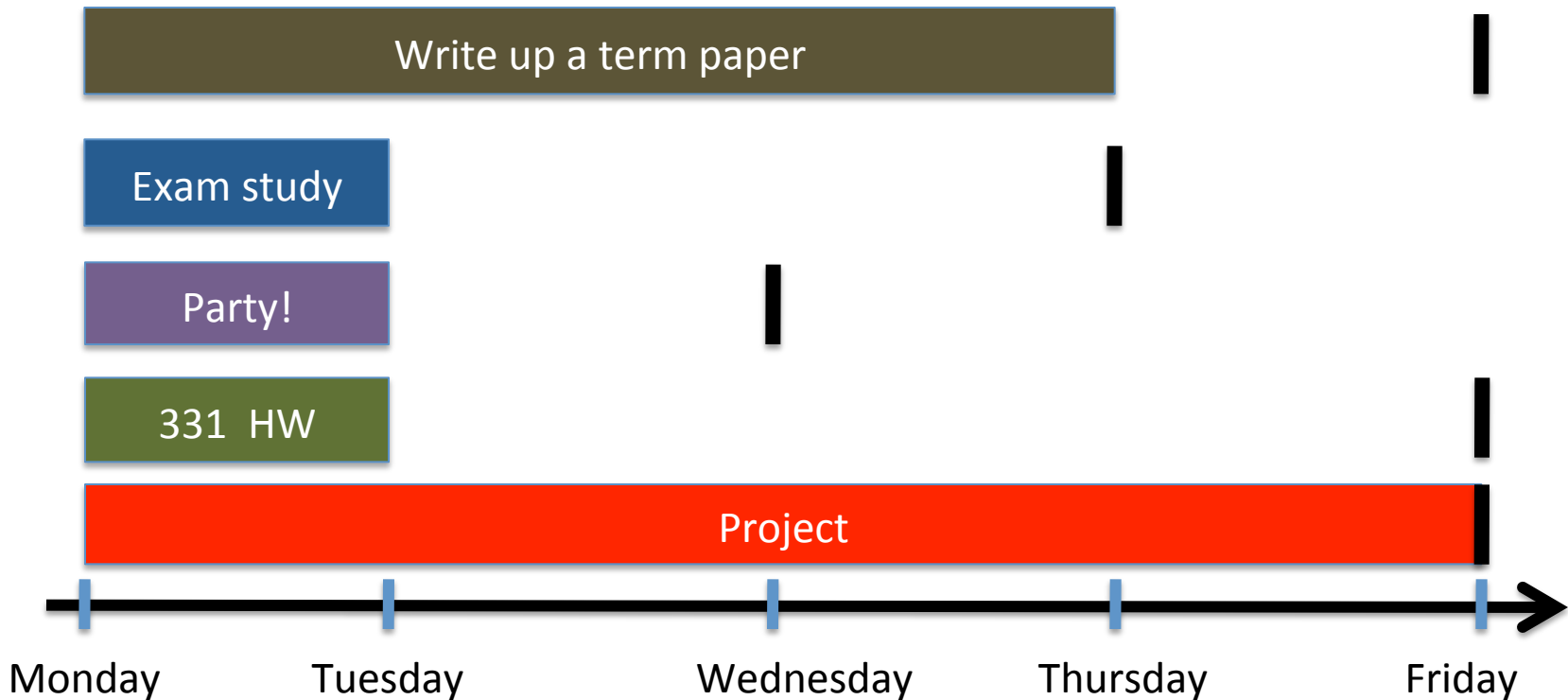
Wednesday

Thursday

Friday

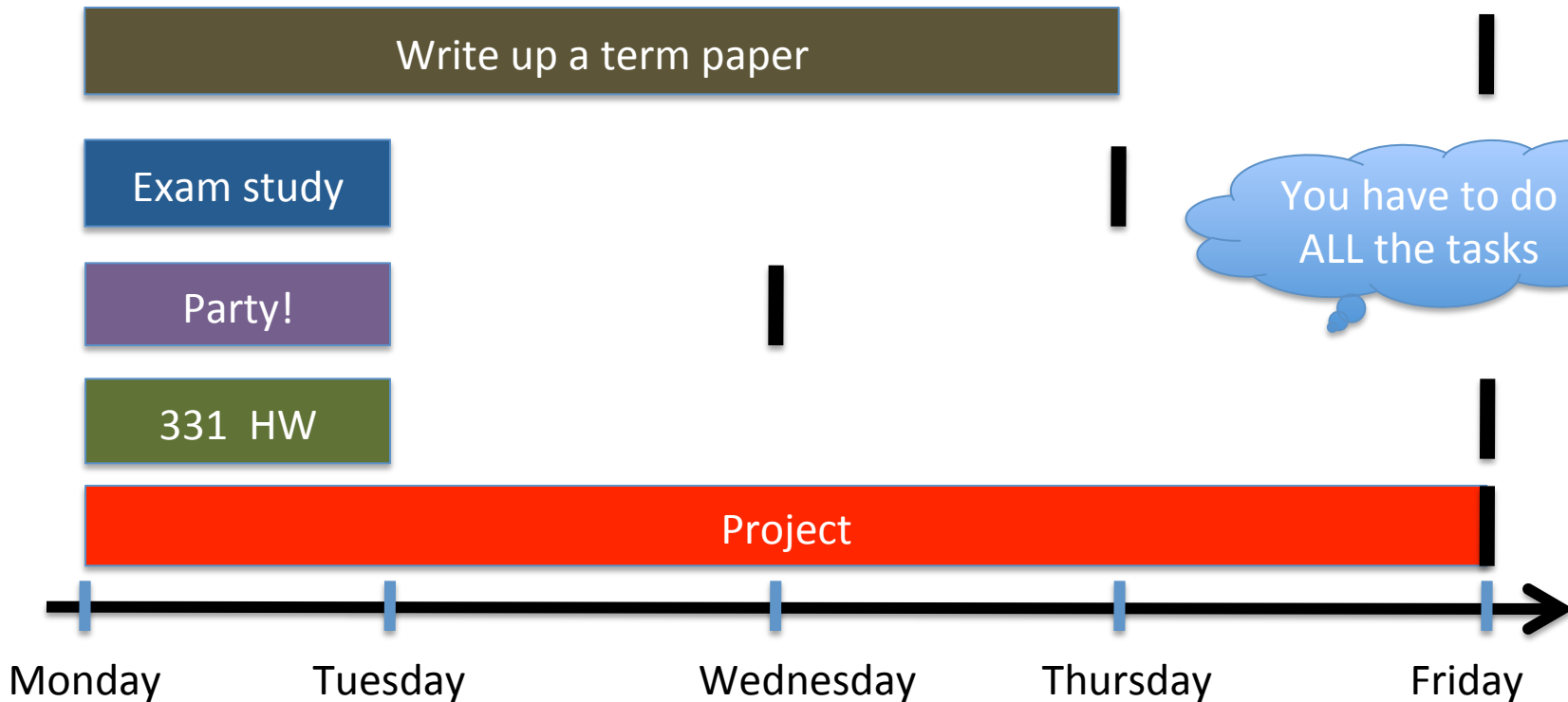
The “real” end of Semester blues

There are deadlines and durations of tasks



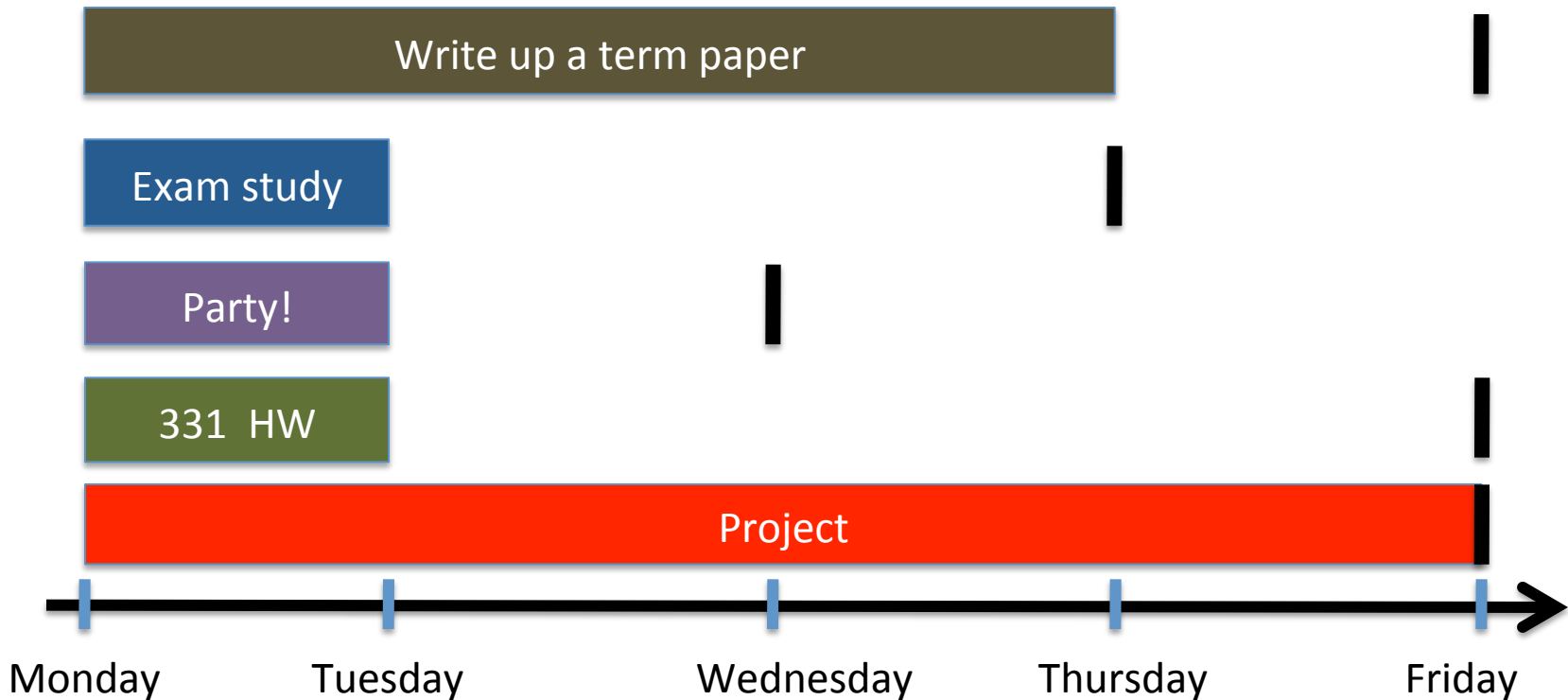
The algorithmic task

YOU decide when to start each task



Scheduling to minimize lateness

All the tasks have to be scheduled
GOAL: minimize maximum lateness



One possible schedule

All the tasks have to be scheduled
GOAL: minimize maximum lateness

