# Lecture 17

CSE 331

Mar 10, 2021

# Quiz 1 on Friday

note @486

## Quiz 1

The first quiz will be on **Friday, March 12, between 3:00-3:15pm**. We will have a 5 mins break after the quiz and **the lecture will start at 3:20 pm at its usual place**.

The quiz will be **in-class** and **LIVE.** There will be two T/F with justification questions (like those in the sample midterm 1: @485) Also, quiz 1 will cover all topics we cover in class till Wednesday, M
internet exam. **You can use one letter-sized cheat-sheet during the quiz (you can use both sides).**

**In this quiz, you'll read questions from your computer screen, solve on paper, and then will take the pictures of the solutions and upload them as a single pdf to Autolab's "Quiz 1" ass**

Now read the instructions below very carefully:

- The link for the Zoom room that you'll use during the quiz will be emailed to you on Thursday, Mar 11.
- The quiz starts at 3:00pm but **you must be online at the provided Zoom link at 2:50pm.**
  - Proctors will be online at 2:50pm and will do necessary checks.
- Before the exam, you must show your UB id to the proctor. If you don't have it (!), you must show your driver's license or passport (your photo must be there!)
- **The exam duration is 15 minutes**.
  - If this was in-class, we'd give you only 10 mins. But we're giving extra 5 minutes so that you can upload your solutions to AutoLab (see below for more on this).
  - The deadline for submission in Autolab will be set to **3:15pm sharp (no extensions!).**
- You'll receive an email (to your buffalo address) at **2:58pm** that contains the link to the quiz (pdf). The quiz will be put on UB Box and be accessible by this link

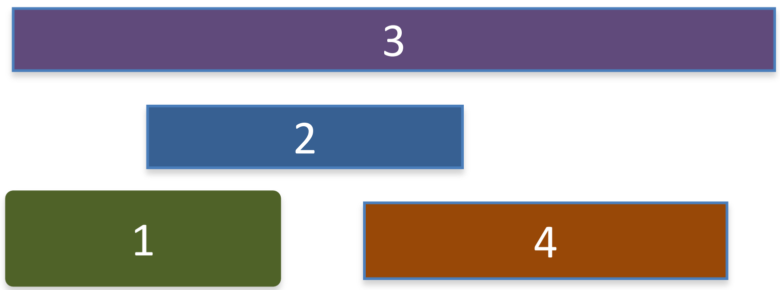
**Read this very carefully!**

# Today’s agenda

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)

In general, if jth interval is the last one chosen

Pick smallest i>j such that s[i] ≥ f(j)

O(n log n) run time

# The final algo

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

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

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

For $i = 2 .. n$

    If $s[i] \ge f$

        Add $i$ to $S$

        Set $f = f(i)$

Return $S^* = S$

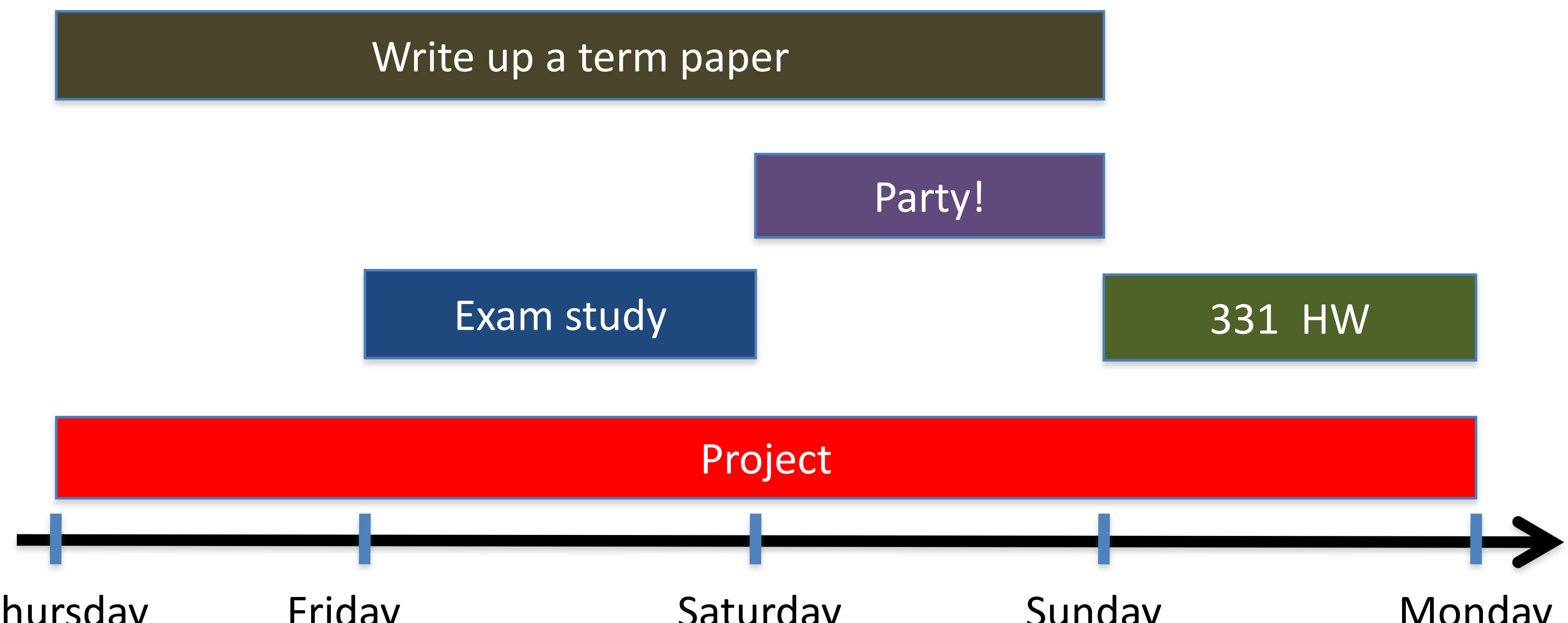
# Reading Assignment

Sec 4.1of [KT]

# Questions?

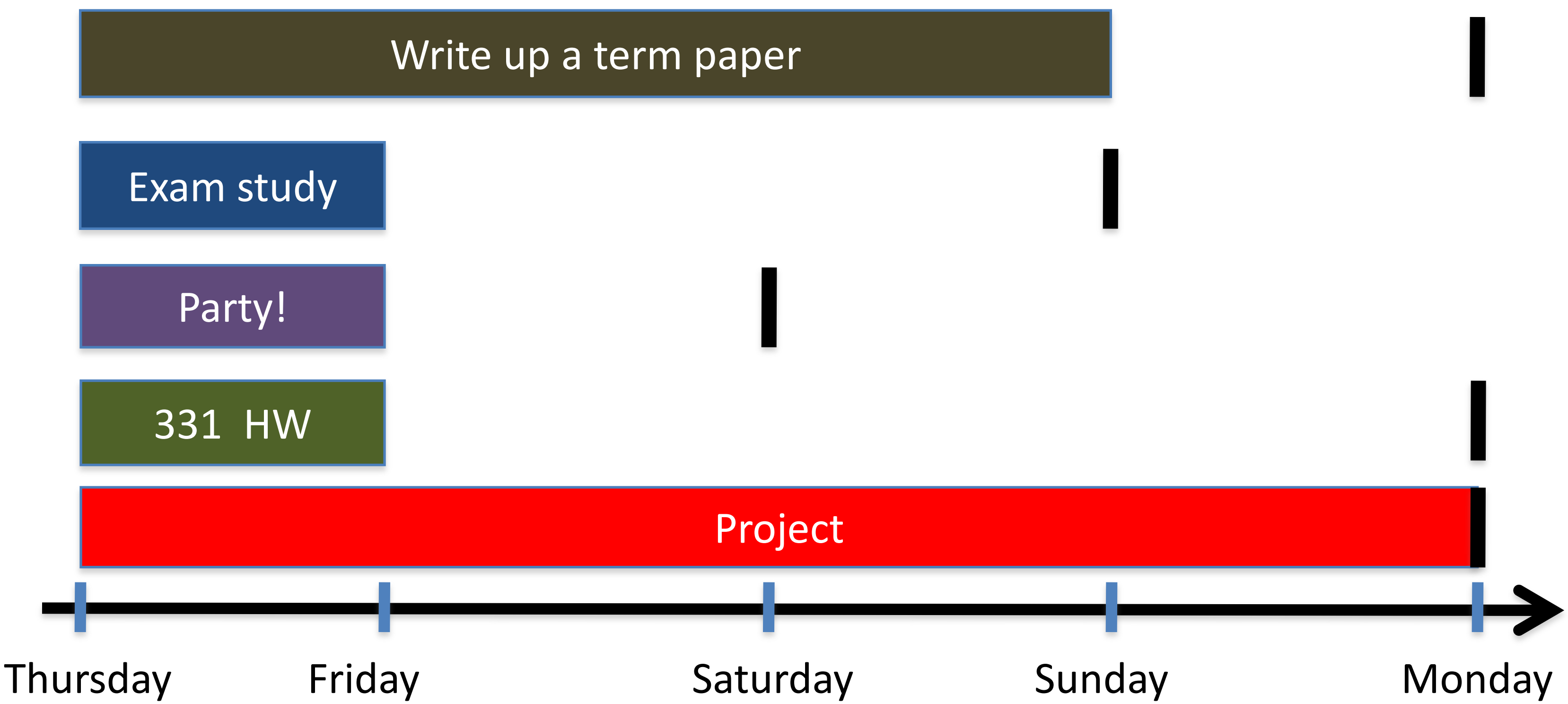
# The “real” end of Semester blues

There are deadlines and durations of tasks



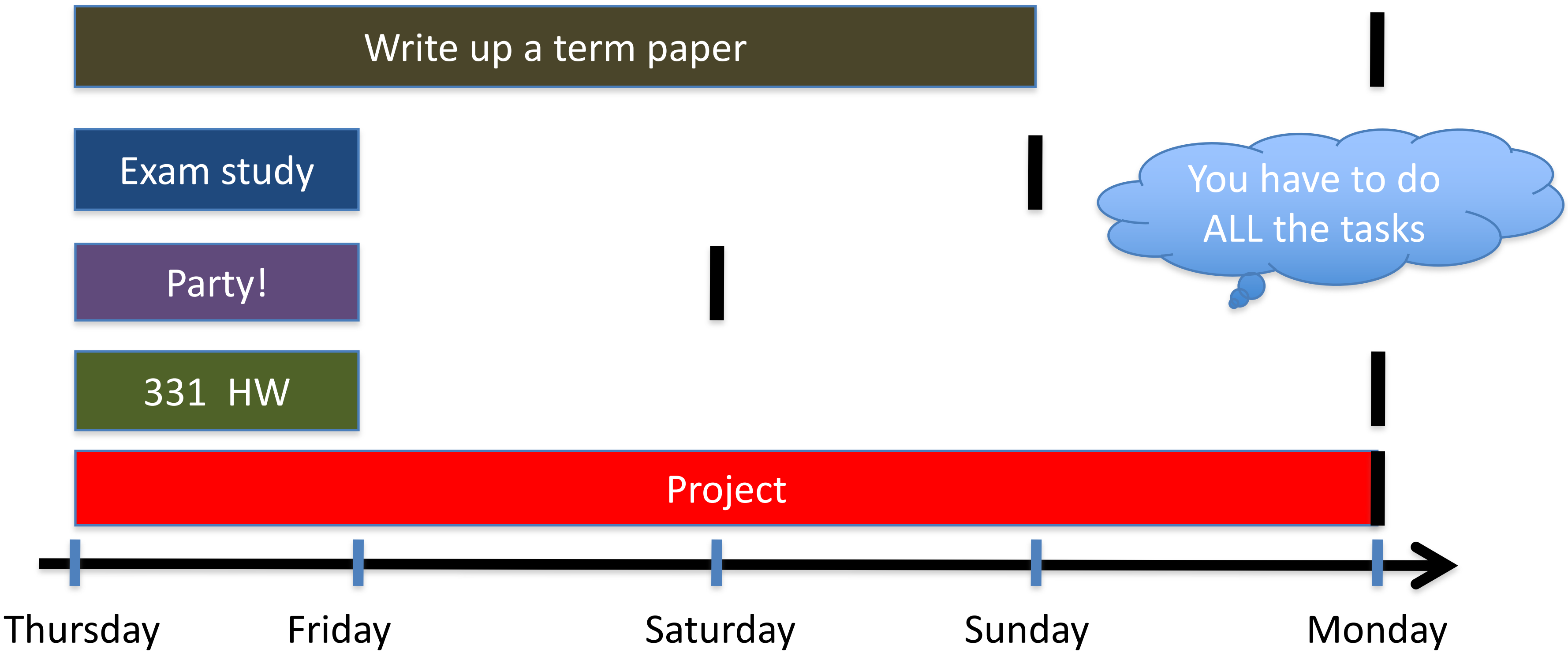
# The “real” end of Semester blues

There are deadlines and durations of tasks



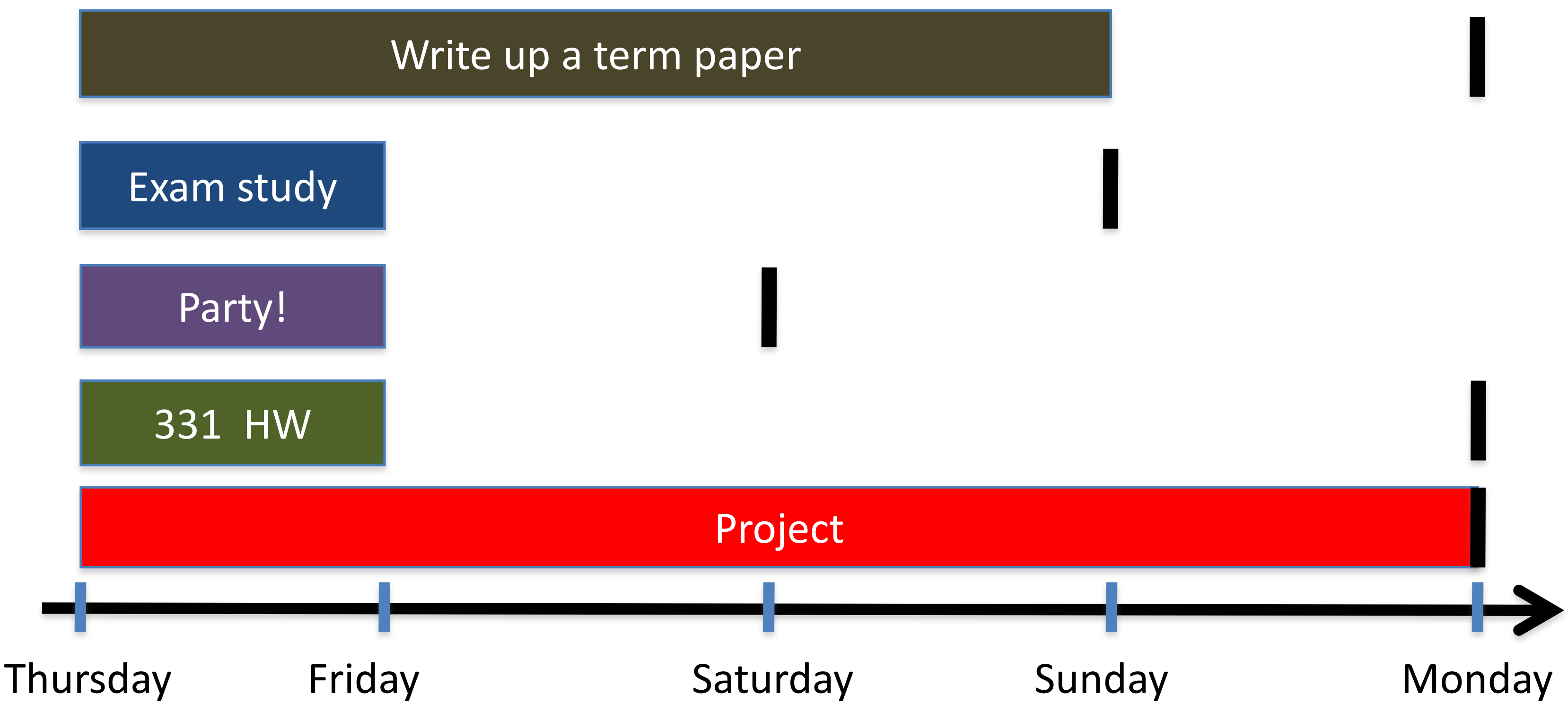
# The algorithmic task

YOU decide when to start each task



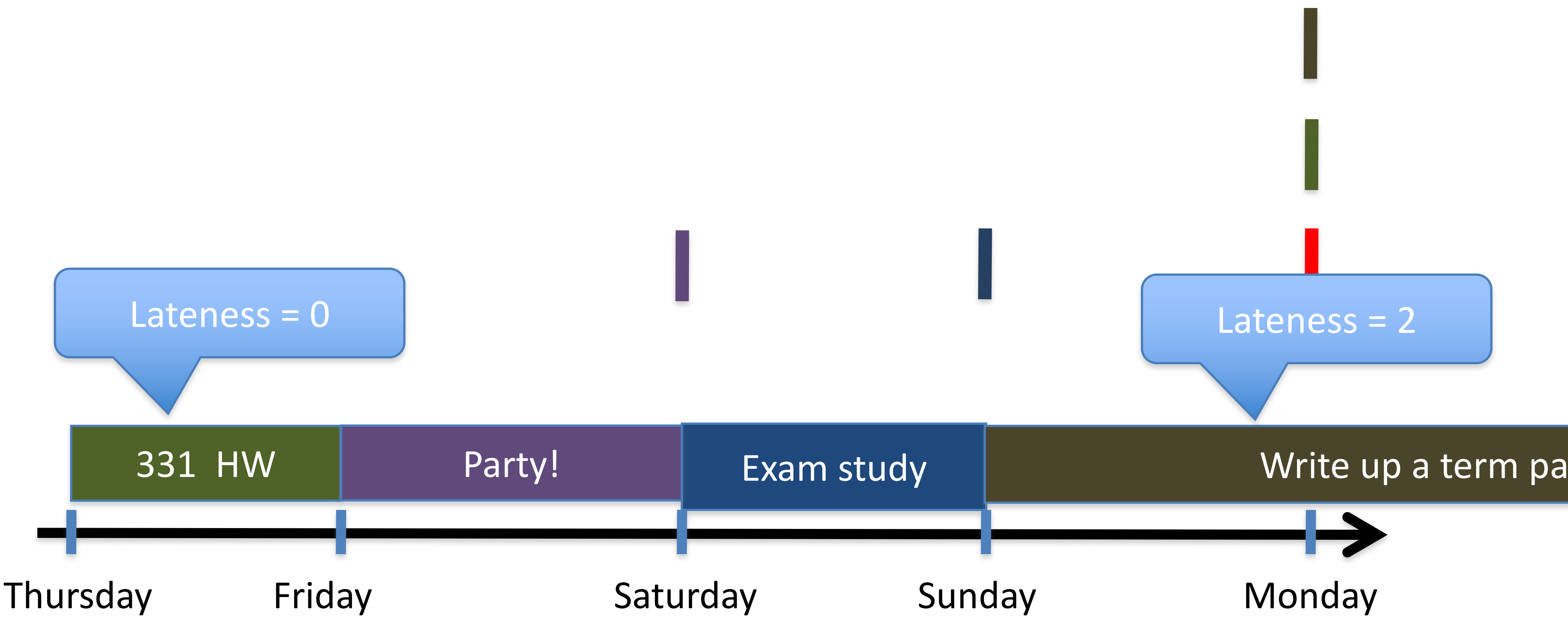
# Scheduling to minimize lateness

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



# Minimizing Max Lateness

# Minimizing Maximum Lateness

This page collects material from previous incarnations of CSE 331 on scheduling to minimize maximum lateness.

## Where does the textbook talk about this?

`Section 4.2` in the textbook has the lowdown on the problem of scheduling to minimize maximum lateness.

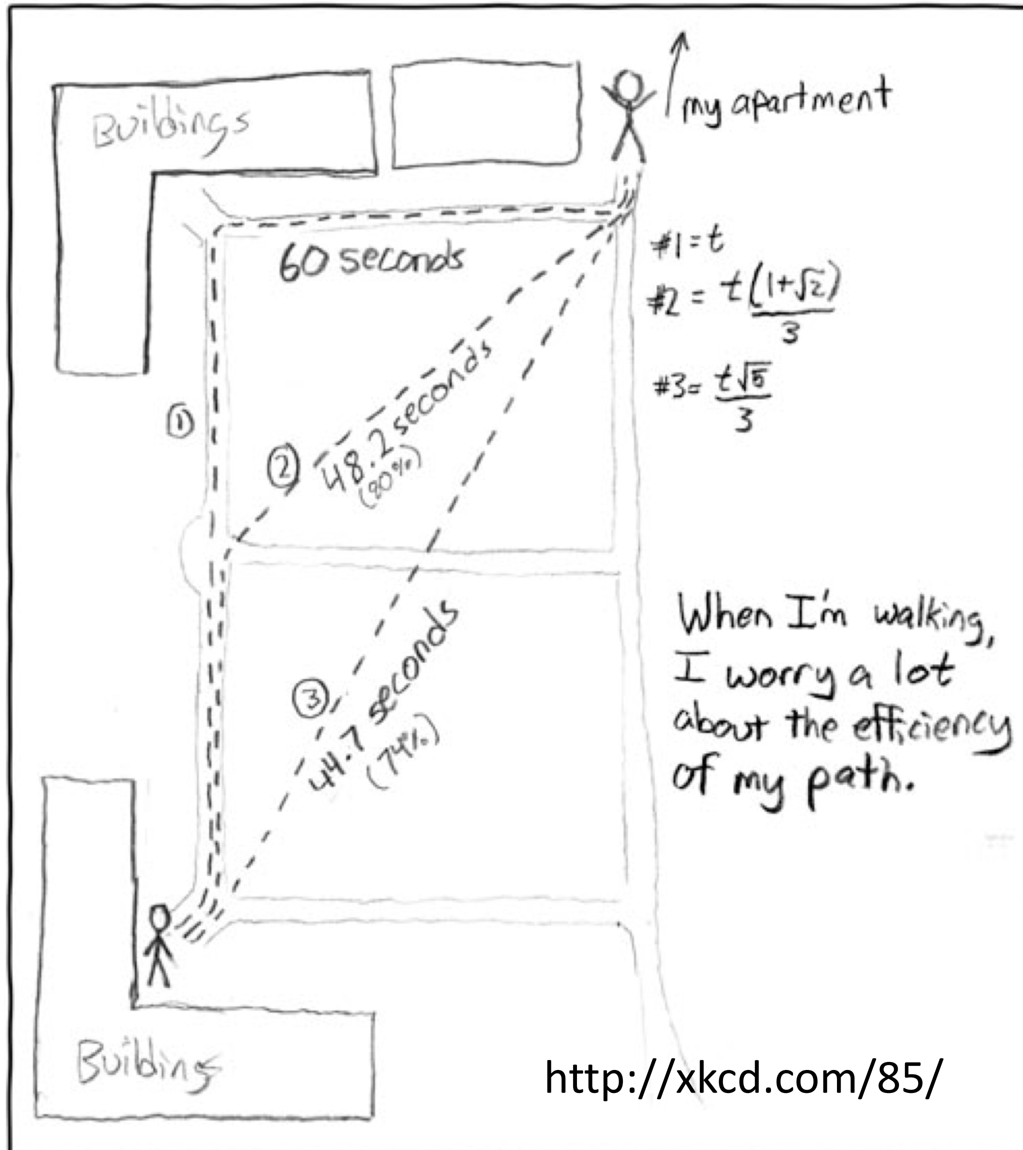
## Fall 2018 material

### First lecture

Here is the lecture video:

# Today

Shortest Path Problem

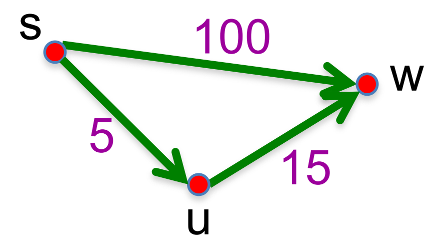



http://xkcd.com/85/

# Reading Assignment

Sec 2.5 of [KT]

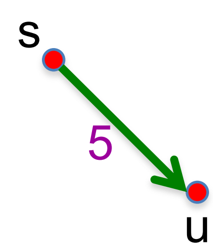
# Shortest Path problem

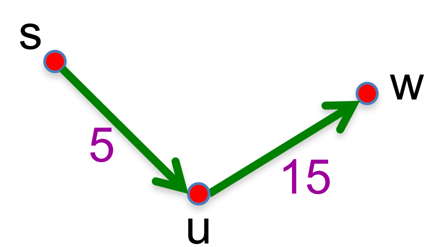



**Input:** *Directed* graph $G=(V,E)$

Edge lengths, $l_e$ for $e$ in $E$

“start” vertex $s$ in $V$




**Output:** All shortest paths from $s$ to all nodes in $V$

# Naïve Algorithm

$\Omega(n!)$ time

# Dijkstra’s shortest path algorithm