Lecture 33

CSE 331

Nov 21, 2022

HW 7 reminders

Homework 7

Due by **11:30pm, Tuesday, November 29, 2022**.

Make sure you follow all the homework policies.

All submissions should be done via Autolab.

Question 1 (Ex 1 in Chap 6) [50 points]

The Problem

Exercise 1 in Chapter 6. The part `(a)` and `(b)` for this problem correspond to the part (**(a)+(b)**) and part **(c)** in Exercise 1 in Chapter 6 in the textbook (respectively).

Sample Input/Output

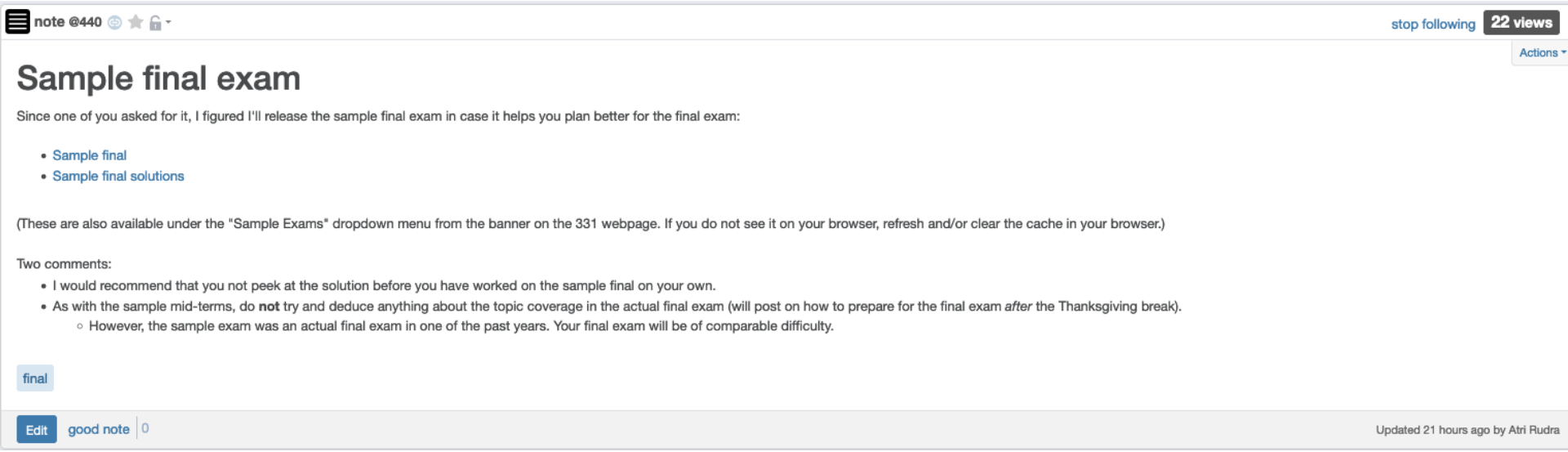
See the textbook for a sample input and the corresponding optimal output solution.

! Note on Timeouts

For this problem the total timeout for Autolab is 480s, which is higher the the usual timeout of 180s in the earlier homeworks. So if your code takes a long time to run it'll take longer for you to get feedback on Autolab. **Please start early to avoid getting deadlocked out before the submission deadline.**

Also for this problem, `C++` and `Java` are way faster. The 480s timeout was chosen to accommodate the fact that Python is much slower than these two languages.

Sample final exam



CSE 331 UTA positions for 2023

note @441 stop following 15 views

Actions

Want to be a UTA for 331 in 2023?

Prof. Akhter be teaching 331 in the upcoming Spring semester and is looking for UTAs. I expect to be teaching 331 again in Fall 2023 (though this is **not** finalized and is subject to change) and will be looking for TAs then as well. So Prof. Akhter and I are looking to jointly interviewing candidates for CSE 331 TAs for 2023 (on **zoom** tentatively the final week (Dec 13 and after) and/or the week after that (week of Dec 19), 2022).

(As an aside: I also have openings for doing research but I'll post on those once I'm done with all 331 related stuff: i.e. after the grades have been submitted.)

These will be *paid* positions. Time-commitment wise here is what we're looking for

- *Ideally,* you should be able to commit close to 10 hours/week on average. More is of course better!
- Depending on your background (e.g. if you have TAed before), we're willing to be OK with ~5 hours/week on average but no lower than that (and no more than 1-2 TAs with << 10 hrs/week).

A few important points:

- There is *no* formal minimum grade requirement to be a 331 UTA (Of course you don't know your grade by now). For now, we're basically looking for interested students who enjoyed 331 so far and would be excited to help others.
- A large fraction of your current TAs will be TAing CSE 331 this spring (but pretty much all of them will be gone by the summer) so there will be fewer slots for Spring 23 (5-10) as compared to Fall 23 (10+).
- Being a 331 UTA is definitely a great experience (feel free to ask one of your TAs!) and also **a great preparation for your interviews -- there is no better way to learn algorithms than to teach it!**
- The application process is basically you presenting an algorithm that is covered in class to a "mock recitation"-- once you apply, we will provide more details on the process.

If you are interested in a UTA position, please fill this form.

piazza logistics

Edit good note | 0

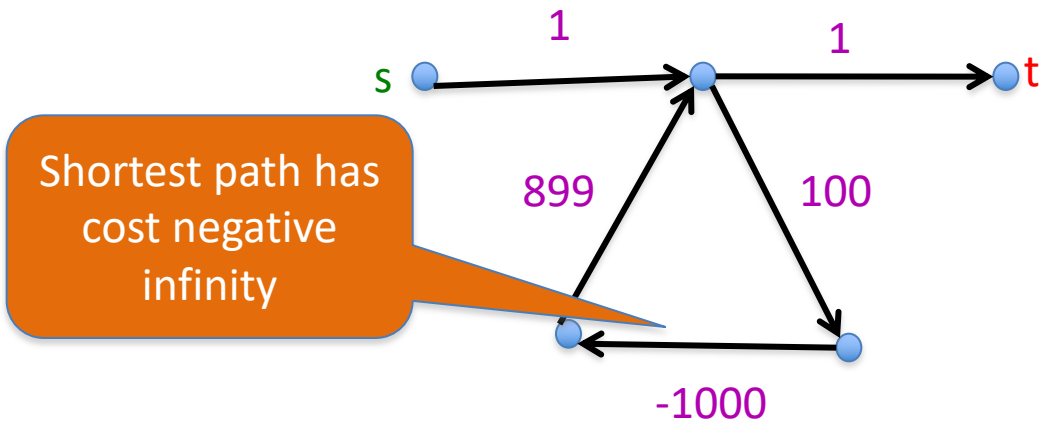
Updated 14 hours ago by Atri Rudra

Shortest Path Problem

Input: (Directed) Graph G=(V,E) and for every edge e has a cost c_e (can be <0)

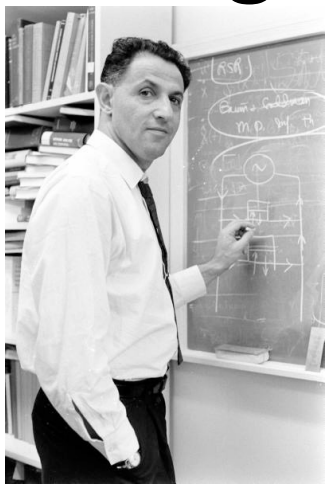
t in V

Output: Shortest path from every s to t



Assume that G has no negative cycle

When to use Dynamic Programming



Richard Bellman

There are polynomially many sub-problems

Optimal solution can be computed from solutions to sub-problems

There is an ordering among sub-problem that allows for iterative solution

Questions?

Today's agenda

Bellman-Ford algorithm

Analyze the run time

Algo on the board…

The recurrence

OPT(u,i) = shortest path from u to t with at most i edges

$$OPT(u,i) = \min \left\{ OPT(u,i-1), \min_{(u,w) \text{ in } E} \left\{ c_{u,w} + OPT(w, i-1) \right\} \right\}$$

Some consequences

OPT(u,i) = cost of shortest path from u to t with at most i edges

$$OPT(u,i) = \min \{ OPT(u, i-1), \min_{(u,w) \text{ in } E} \{ c_{u,w} + OPT(w,i-1) \} \}$$

OPT(u,n-1) is shortest path cost between u and t

Group talk time:
How to compute the shortest path between s and t given all OPT(u,i) values