Lecture 33

CSE 331

Nov 15, 2019

HW 9 out

Homework 9

Due by **11:00am, Friday, November 22, 2019**.

Make sure you follow all the homework policies.

All submissions should be done via Autolab.

Question 1 (Ex 2 in Chap 6) [50 points]

The Problem

Exercise 2 in Chapter 6. The part `(a)` and `(b)` for this problem correspond to the part **(a)** and part **(b)** in Exercise 2 in Chapter 6 in the textbook.

Sample Input/Output

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

Hint

HW 8 solutions

At the end of the lecture

HW 7 Grading

Hopefully by tonight

Coding project Java templates

note

stop following 59 views

Please re-download Java zips for Problem 1-4

If you are not using Java for your mini-project, then you can safely ignore this post. Otherwise read-on.

tl;dr: **Please download the Java zips for Problem 2-5 and work on the updated zips from now on**.

Longer version: There was a bug in the template code that could show a different revenue for your solution than the revenue on Autolab. (*The grader code on Autolab has been fine and so no need to worry about your scores on Autolab changing*.) The updated zips should fix this and issue and you should now see the same revenue on both the template and Autolab.

If you are still reading: the issue at a high level was the following-- if your code was changing the input (e.g. re-sorting the clients for P2), then the Autolab grader code was ignoring these changes (as it should). Unfortunately, the previous template code was **not** ignoring these changes, leading to a discrepancy between the revenue from the template code and Autolab. Anyhow, this should be fixed now!
#pin

coding_mini_project

edit · good note | 0

Updated 16 hours ago by Atri Rudra

Coding project Problem 2

Due in (bit less than) a week

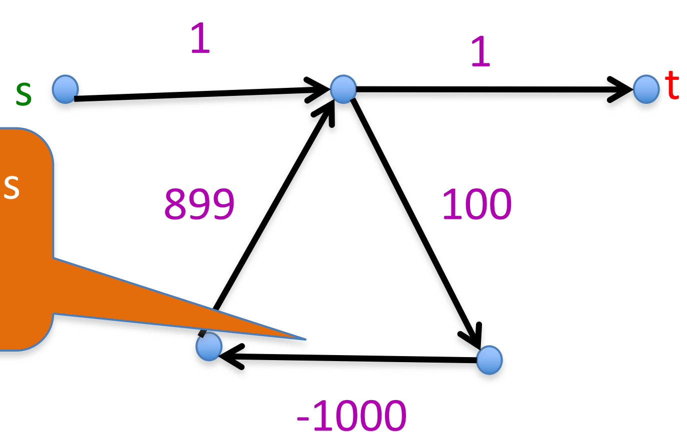
Questions?

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

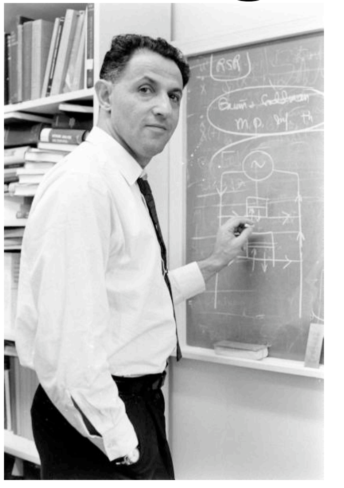
Output: Shortest path from every s to t



Shortest path has cost negative infinity

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

Today's agenda

Bellman-Ford algorithm

Analyze the run time

The recurrence

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

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

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