Lecture 25

CSE 331

Oct 31, 2018

Happy Halloween!

Peer Evaluation survey

Mini project video due MONDAY

note ☆

Actions ▾

You can submit mini project video now

You can now submit your mini project videos now. It is due in a bit over 2 weeks: by **11:59pm on Mon, Nov 5**.

The mini-project page has all the details on what is needed in the submission.

Some important points:

- Please make sure you read through the instructions/requirements carefully.
  - Till last year there used to be an intermediate report stage where I could give some preliminary feedback so that y'all could avoid some of the common mistakes in the video. Y'all do not have the luxury, so please make sure you read through the page very very carefully.
- This is a **group submission**. Please see the instructions at the end of this post.
  - Main thing: do **NOT** submit your report till your group is formed.
- **Check on your group**. We are getting close to the resign date. Unfortunately, some students will drop-- so make sure you check with your group mates to see if they'll be around.
  - If your group-mate(s) drop out, then it is OK for you to continue with a smaller group.
    - Even a group of size 1 is OK if you're fine with it. But if not AND if you give me enough notice, I can try and re-assign you to another group.

There are still free slots

Meetings to discuss CSE 331 performance

I will email those who have a D or below in their mid-term grade (for more details on the grade see @736). Of course you can also come and talk about your 331 performance even if you have a temp grade higher than D (though students with a D or below will get preference).

I have locked out certain times over next week or so for **10 mins** meetings. Please note that **these are NOT walk-ins:** if no one signs up for a slot, I might not be in my office then. If you want to come and talk with me, **please EMAIL me with ALL the slots below that work for you.** (Private posts on piazza will not work: please email me!) *Slots will be assigned on a first-come-first-serve basis.*

Below are all the available slots (below the start times are listed: a slot that is already taken has a strike-through):

- **Tuesday (Oct 30):** ~~11:50am~~, ~~12:00pm~~, ~~12:10pm~~, ~~12:20pm~~, ~~12:30pm~~, ~~3pm~~
- **Wednesday (Oct 31):** 11:10am, 11:20am, ~~11:30am~~, 11:40am, 11:50am, 12:00pm, 12:10pm, 12:20pm, 12:30pm
- **Thursday (Nov 1):** 11am, 11:10am, 11:20am, 11:30am, 11:40am, 11:50am, ~~1:30pm~~, 1:40pm, 1:50pm, 2:00pm, 2:10pm, 2:20pm, 2:30pm, 2:40pm, 2:50pm, 3:00pm, 3:10pm, 3:20pm, ~~3:30pm~~, ~~3:40pm~~, ~~3:50pm~~, ~~4:00pm~~, ~~4:10pm~~, ~~4:20pm~~, ~~4:30pm~~, ~~4:40pm~~
- **Friday (Nov 2):** 11:10am, 11:20am, 11:30am, 1:40pm, 1:50pm, ~~4:00pm~~, ~~4:10pm~~, 4:20pm, 4:30pm, 4:40pm
- **Monday (Nov 5):** 12:00pm, 12:10pm, 12:20pm, 12:30pm, 2:00pm, 2:10pm, 2:20pm, 2:30pm, 2:40pm, 2:50pm, 4:00pm, 4:10pm, 4:20pm, 4:30pm, 4:40pm

You can of course also stop by during my office hours (but students with Qs on the HWs will get higher priority) and you unfortunately cannot book a slot during my usual office hours.

All the meetings will be in my office (Davis 319).

Dijkstra's shortest path algorithm

$$d'(v) = \min_{e=(u,v) \text{ in } E, u \text{ in } R} d(u)+l_e$$

Input: Directed $G=(V,E)$, $l_e \geq 0$, s in V

R = {s}, d(s) =0

While there is a x not in R with (u,x) in E, u in R

  Pick w that minimizes d'(w)
  Add w to S
  d(w) = d'(w)

At most n iterations

$\sum_{x \in V} O(\ln_x+1)$ =O(m+n) time

O((m+n)n) time bound is trivial

O((m+n) log n) time implementation with priority Q

Reading Assignment

Sec 4.4 of [KT]

Building a fiber network

Lay down fibers to connect n locations

All n locations should be connected

Laying down a fiber costs money

What is the cheapest way to lay down the fibers?

Today's agenda

Minimum Spanning Tree (MST) Problem

Greedy algorithm(s) for MST problem

Kruskal's Algorithm
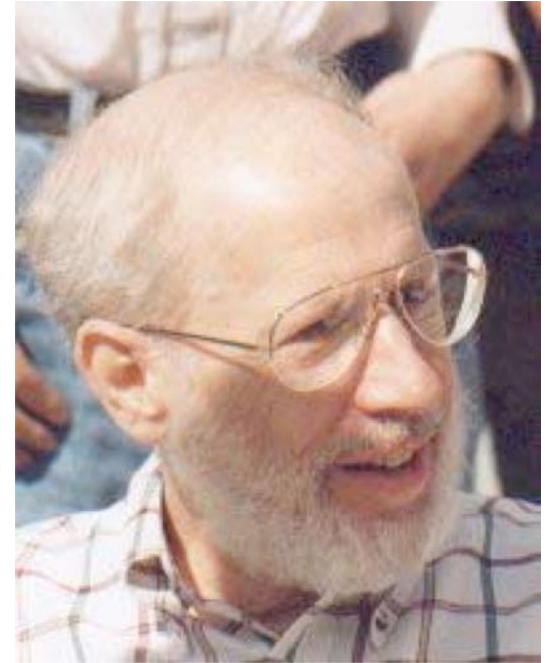

Joseph B. Kruskal

Input: $G=(V,E)$, $c_e > 0$ for every e in E

$T = \emptyset$

Sort edges in increasing order of their cost

Consider edges in sorted order

      If an edge can be added to T without adding a cycle then add it to T
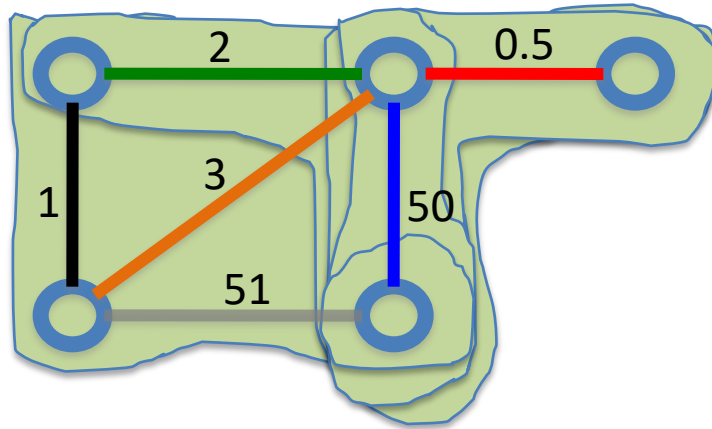
Prim's algorithm
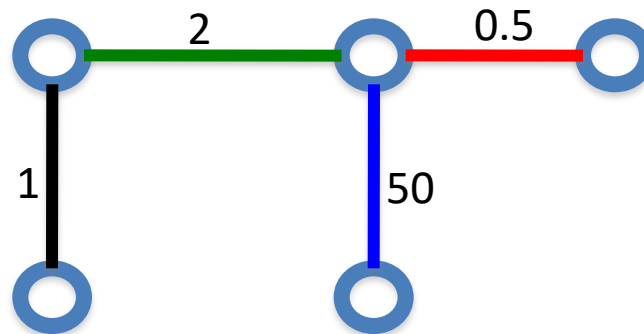
Similar to Dijkstra's algorithm

Robert Prim

Input: $G=(V,E)$, $c_e > 0$ for every e in E
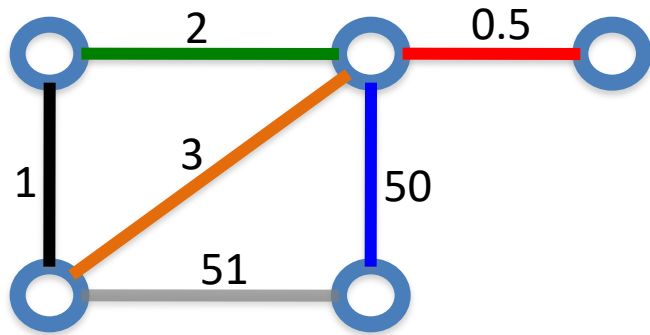
$S = \{s\}$, $T = \emptyset$

While S is not the same as V

    Among edges $e = (u,w)$ with u in S and w not in S, pick one with minimum cost

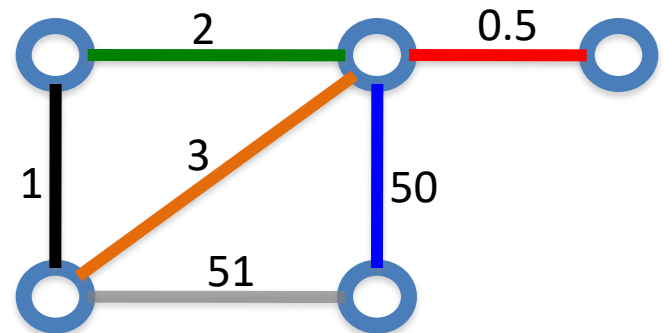   Add w to S, e to T

Reverse-Delete Algorithm

Input: G=(V,E), c_e> 0 for every e in E

T = E

Sort edges in decreasing order of their cost

Consider edges in sorted order

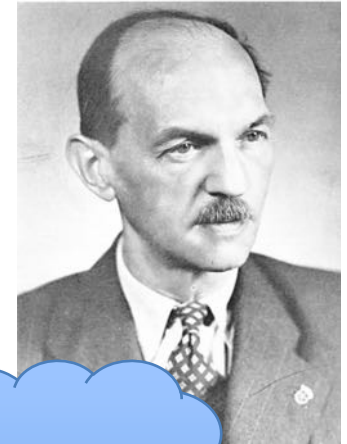 If an edge can be removed T without disconnecting T then remove it

(Old) History of MST algorithms

1920: Otakar Borůvka

1930: Vojtěch Jarník

Same algo!

1956: Kruskal

1957: Prim

1959: Dijkstra