Lecture 22

CSE 331

Oct 25, 2023

Project deadlines coming up

Tue, Oct 31		(**HW 5 in**)
Wed, Nov 1	Multiplying large integers F22 F21 F19 F18 F17 x²	[KT, Sec 5.5] *Reading Assignment:* Unraveling the mystery behind the identity
Fri, Nov 3	Closest Pair of Points F22 F21 F19 F18 F17 x²	[KT, Sec 5.4] (**Project (Problems 1 & 2** `Coding` **) in**)
Mon, Nov 6	Kickass Property Lemma F22 F21 F19 F18 F17 x²	[KT, Sec 5.4] (**Project (Problems 1 & 2** `Reflection` **) in**)

Group formation instructions

Autolab group submission for CSE 331 Project

The lowdown on submitting your project (especially the coding and reflection) problems as a group on Autolab.

Follow instructions **EXACTLY** as they are stated

The instruction below are for Coding Problem 1

You will have to repeat the instructions below for EACH coding AND reflection problem on project on Autolab (with the appropriate changes to the actual problem).

Form your group on Autolab

Groups on Autolab will NOT be automatically created

You will have to form a group on Autolab by yourself (as a group). Read on for instructions on how to go about this.

Click to add notes

HW 5 out

Homework 5

Due by **11:30pm, Tuesday, October 31, 2023**.

Make sure you follow all the homework policies.

All submissions should be done via Autolab.

Check the week 9 recitation notes for this homework.

Question 1 (Computing Set Intersection on a Network) [50 points]

The Problem

In this problem, we will take a break from trying to minimize the runtime of the algorithm and focus on an important resource in distributed computing: the total number of bits communicated over a network by the algorithm.

! For those of you who are feeling a little ambitious

For the top 3 submissions in the scoreboard in Python, the top 2 submissions in the scoreboard in Java and the top submission in the scoreboard in C++, we are offering 2.5 bonus points. But be warned! You should not be spending too much time on this. We rather you work on Questions 1 and 2 above.

I finished grading all Q3(b)s

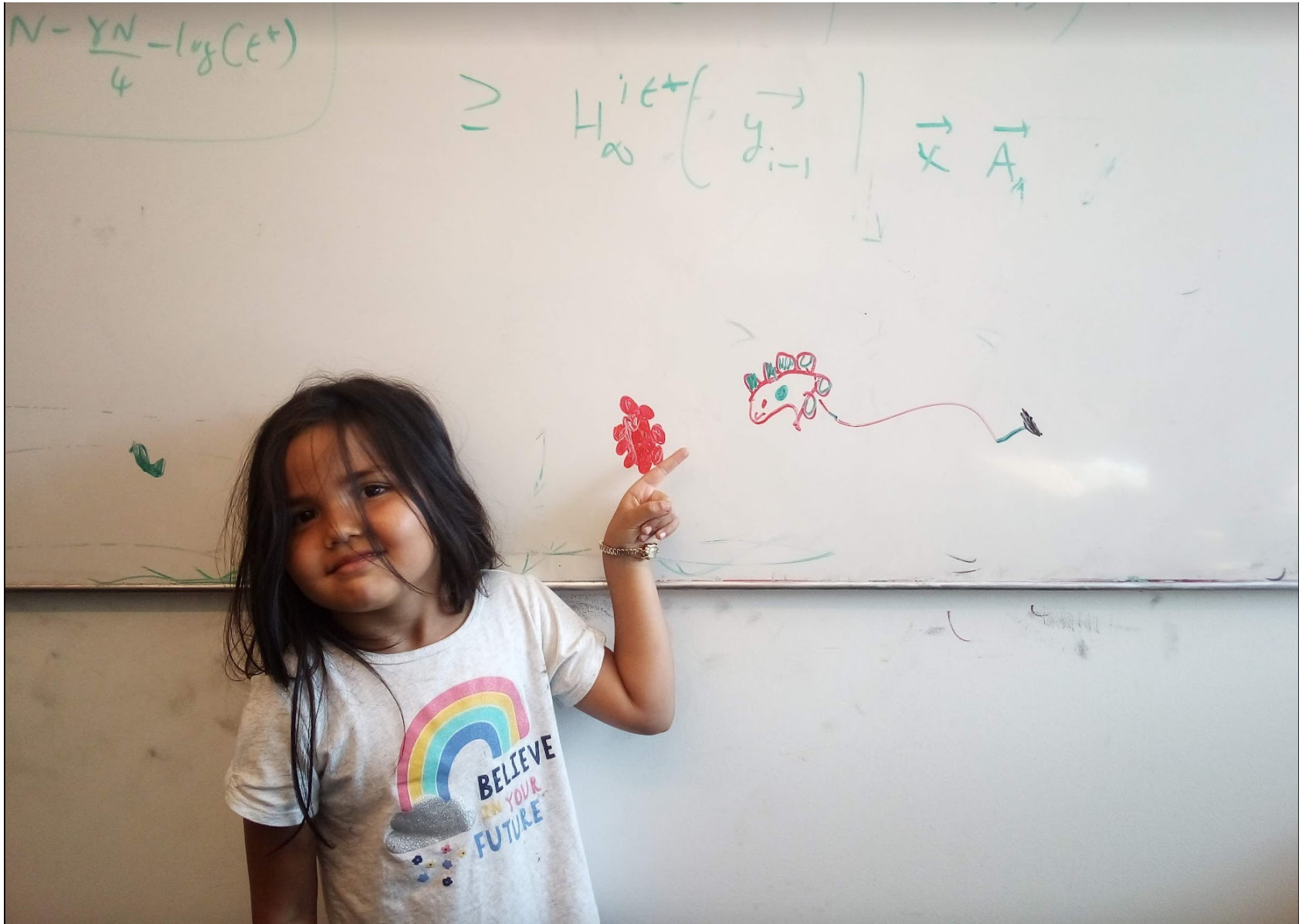
Mid-term grades should be out by <= tomorrow afternoon

It was not pretty……

And I don't get why many of you just left Q3(b) blank!

You get better by struggling and not giving up!

Questions/Comments?



Minimum Spanning Tree Problem

Input: Undirected, connected $G = (V,E)$, edge costs c_e

Output: Subset $E' \subseteq E$, s.t. $T = (V,E')$ is connected
$C(T)$ is minimized

If all $c_e > 0$, then T is indeed a tree

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

$S = \{s\}$, $T = \emptyset$

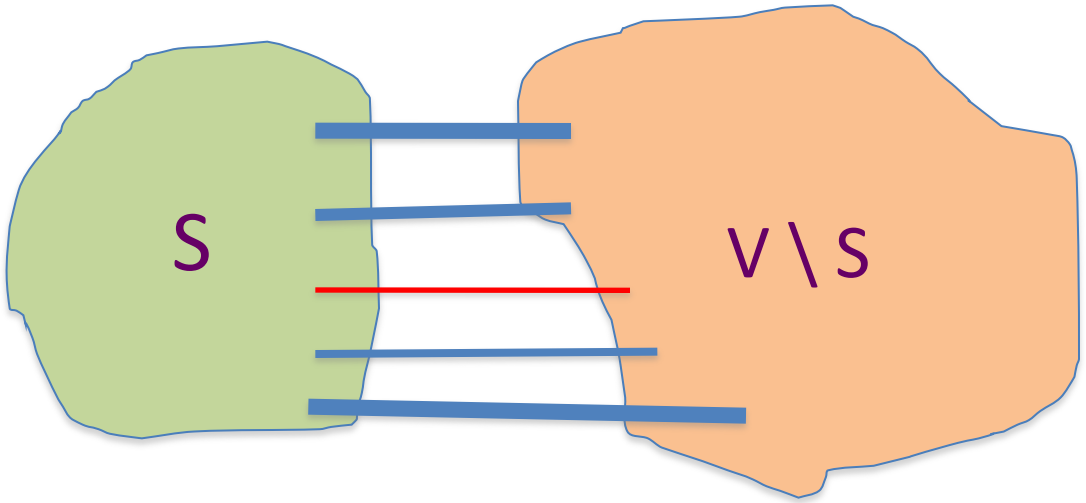
While S is not the same as V

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

Cut Property Lemma for MSTs

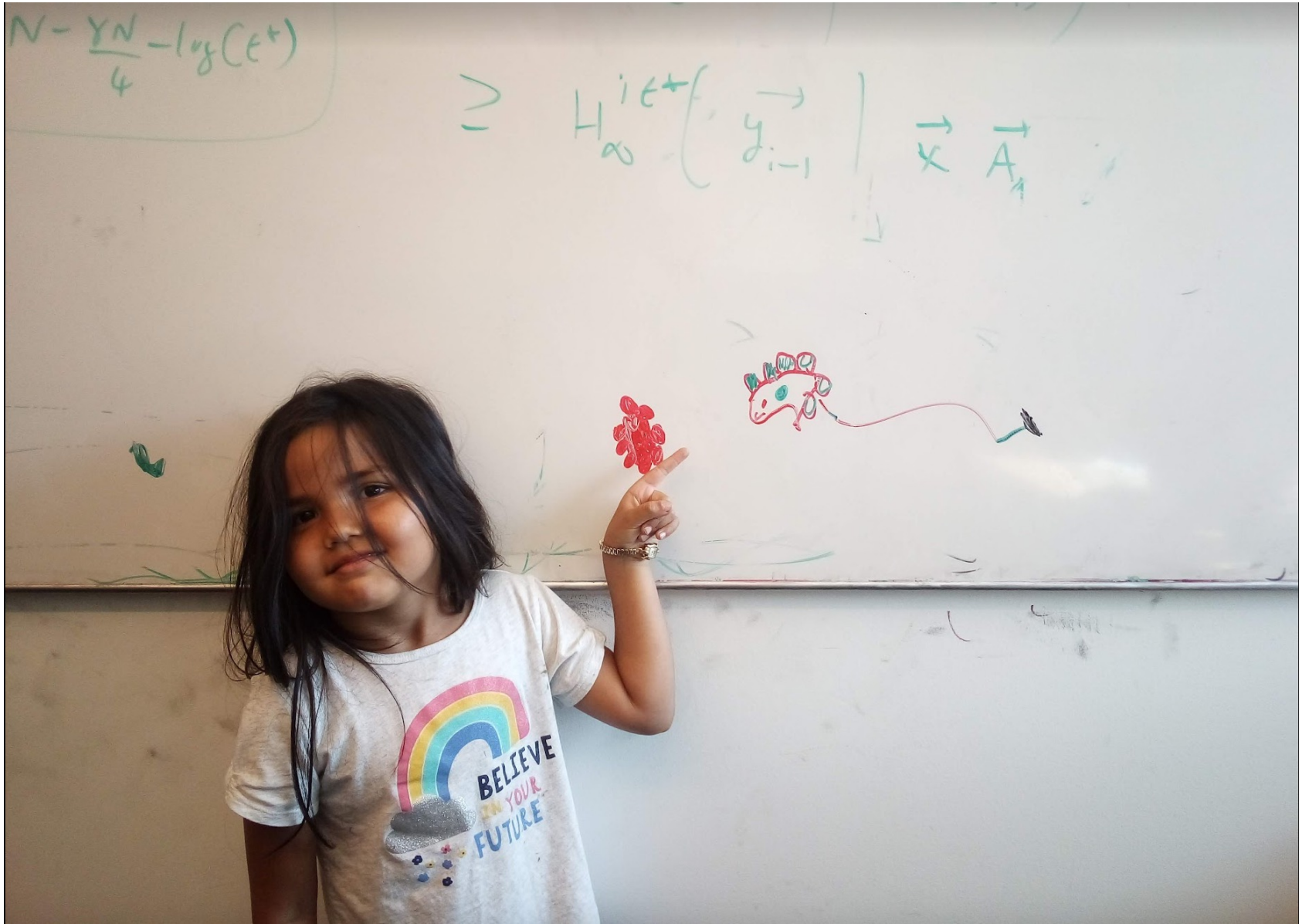
Condition: S and V\S are non-empty



Cheapest crossing edge is in **all** MSTs

Assumption: All edge costs are distinct

Questions/Comments?



Today's agenda

Optimality of Prim's algorithm

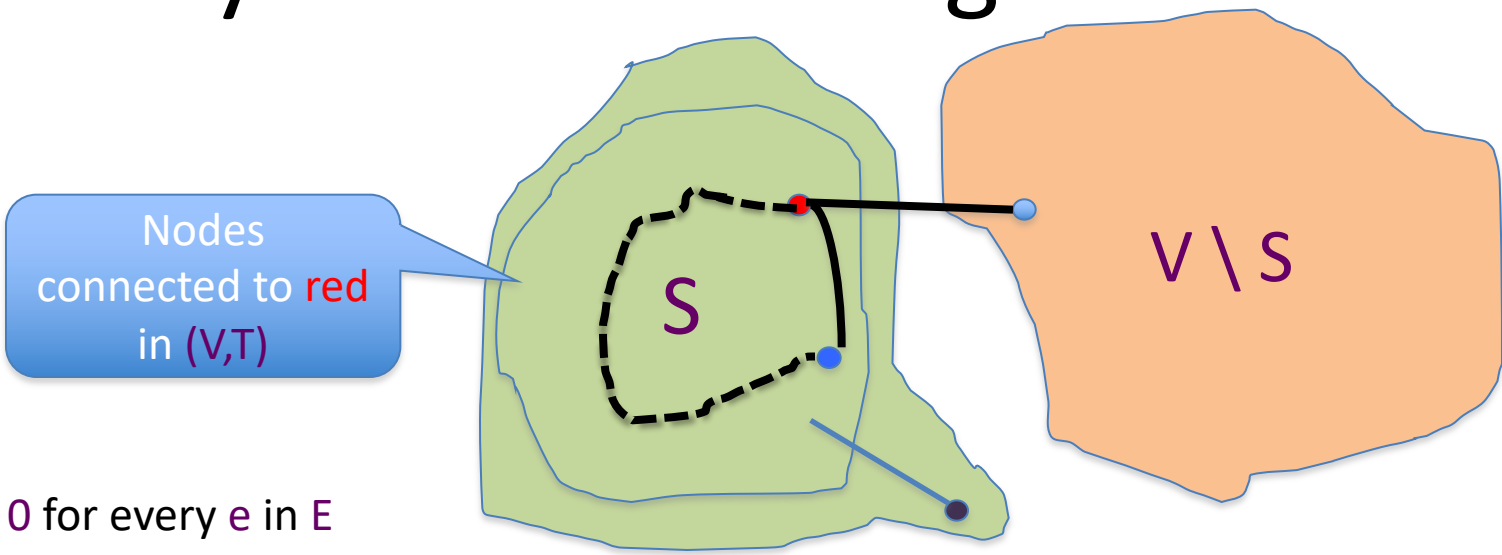
Prove Cut Property Lemma

Optimality of Kruskal's algorithm

Remove distinct edge weights assumption

On to the board…

Optimality of Kruskal's Algorithm



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

T = Ø

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

S is non-empty

V\S is non-empty

First crossing edge considered