Lecture 6

CSE 331

Sep 10, 2018

Mini project choice due in 2 weeks

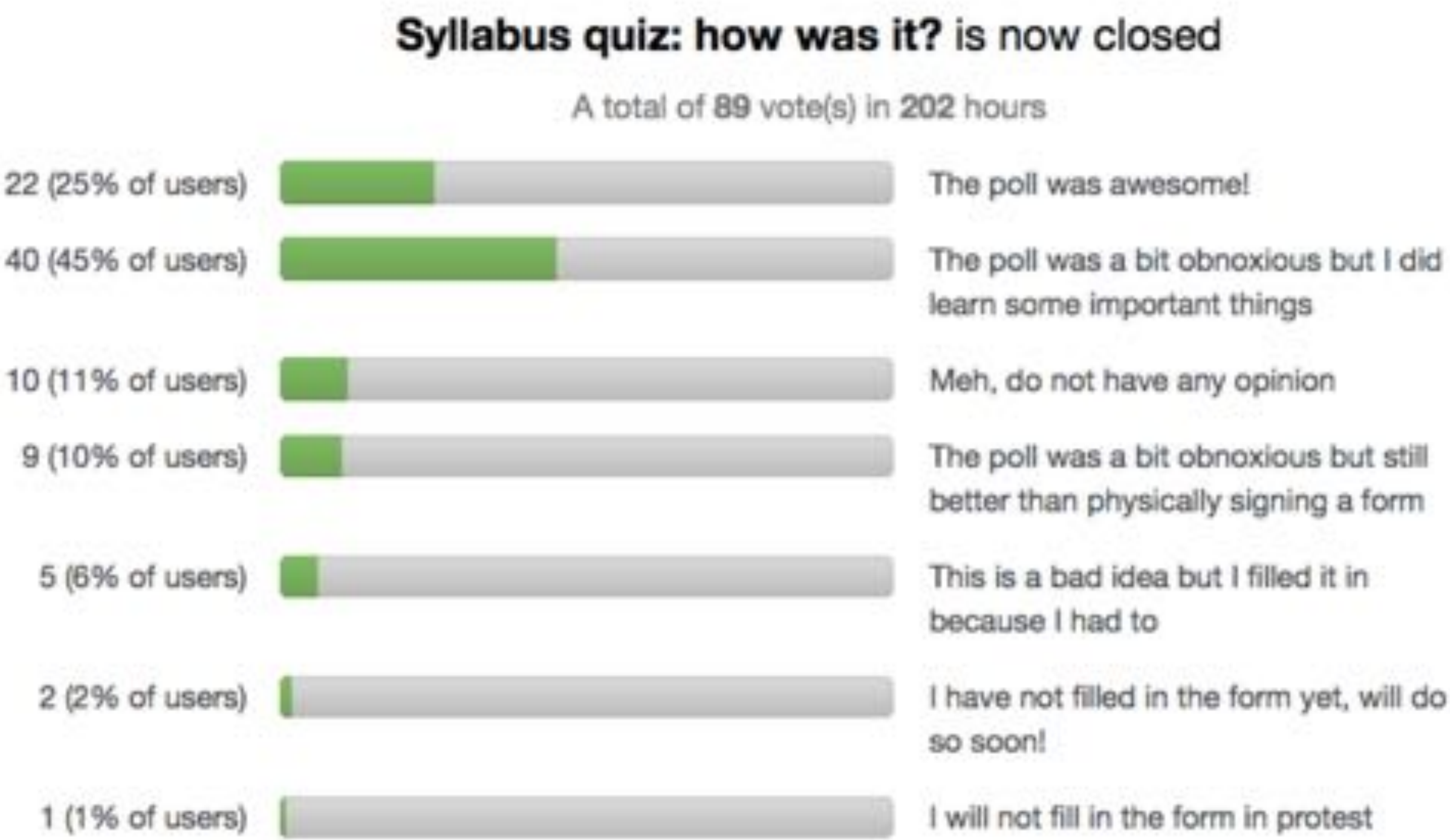
CSE 331 Mini project choices

Fall 2018

Please check the table below before submitting your mini project team composition to make sure your case study is not being used by another group. Case studies are assigned on a first come first serve basis.

Group	Chosen Algorithm	Case Study	Links
Chinmayee Bandal, Sarah Peters, Tracy Zheng ()	Dijkstra's Algorithm	Google Maps	Link 1, Link 2
Kai Bustos, Brian Balayon, Hans Bas ()	Deep Neural Networks	Youtube Recommendations	Link 1, Link 2
Abdulrahman Alsammarraie, Jared Boswell, Peter Klotzbach (Abdu, Jared, Peter)	PageRank Algorithm	Google Page Rank	Link 1, Link 2, Link 3, Link 4
Justin Cole, Aunik Ahmed, Andrew Oslica (Team 7)	On-Road Integrated Optimization and Navigation (ORION)	UPS	Link 1, Link 2, Link 3
Alvin Lin, Tyler Gelinas, Rena Rodriguez (Don't care)	Pagerank**ALREADY TAKEN-- PLEASE CHOOSE ANOTHER CASE STUDY**	Google Search	Link 1, Link 2

Thanks for the feedback!



Peer Notetaker needed

note

stop following

129 views

Peer Notes Request

A student in your class is eligible for the services of a Peer Notetaker. Notetakers provide an essential service that helps ensure equal access to education for students who receive accommodations. Notetakers who qualify may also be paid a stipend by Accessibility Resources at the end of the semester. If you are interested in becoming a Peer Notetaker for this course, please contact 716-645-2608 or stu-notes@buffalo.edu as soon as possible. Notetakers are accepted on a first come, first serve basis.

(If you do end up volunteering for being a peer notetaker, please also let me know so that I know I do not have to send more reminders. --Atri)

#pin

logistics lectures

edit good note 0

Updated 4 days ago by Atri Rudra

What not to do on piazza

note ☆ stop following **20 views**

Actions

Do not ask about your solution in a public post on piazza

I'm copying my comment in @115 here so that this remains pinned:

Piazza is **not** the place to check whether your solution is correct or not. Doing so will be considered to be academic violation.

I know the line can sometime be fine, so here are two rules of thumb to follow:

- Asking questions to understand what the problem is saying is perfectly legit.
    - And indeed, this is how @115 started out as.
- If your question or the discussion veers towards the solution: e.g. if you catch yourself trying to say, "So does this mean I have solved the problem" or saying things like "This is how I think the problem can be solved", then you should ask your question in a private post on piazza for the instructors. And then we'll make a class whether we can make it public or not. (In either case, we'll definitely answer your question.)

#pin

piazza

edit · good note | 0

Updated 6 minutes ago by Atri Rudra

note

stop following 85 views

Reading Assignment for Monday's lecture

Here are some reading assignments for Monday. Please do the following to be prepared for the lecture:

- Read through the support page on pigeon-hole principle: we will be using it as a given during the lecture on Monday.
- On Monday, we will not prove in any detail that there are at most n^2 iteration of the GS algorithm. There are few ways for you to catch-up on this:
  - Watch the video from this lecture last year, where we did go over the argument in detail (starts around the 26 min mark)
  - Read the proof in the textbook
  - The support page on progress measure could also be helpful here.

#pin

lectures

edit good note 0

Updated 1 day ago by Atri Rudra

Gale-Shapley Algorithm

Intially all men and women are free

While there exists a free woman who can propose

> Let w be such a woman and m be the best man she has not proposed to
>
> w proposes to m
>
> If m is free
>
> (m,w) get engaged
>
> Else (m,w’) are engaged
>
> If m prefers w’ to w
>
> w remains free
>
> Else
>
> (m,w) get engaged and w’ is free

Output the engaged pairs S as the final output

Observation 1

Intially all men and women are free

While there exists a free woman who can propose

Let w be such a woman and m be the best man she has not proposed to

w proposes to m

If m is free

(m,w) get engaged

Else (m,w') are engaged

If m prefers w' to w

w remains free

Else

(m,w) get engaged and w' is free

Once a man gets engaged, he remains engaged (to "better" women)

Output the engaged pairs S as the final output

Observation 2

Intially all men and women are free

While there exists a free woman who can propose

Let w be such a woman and m be the best man she has not proposed to

w proposes to m

If m is free

(m,w) get engaged

Else (m,w') are engaged

If m prefers w' to w

w remains free

Else

(m,w) get engaged and w' is free

If w proposes to m after m', then she prefers m' to m

Output the engaged pairs S as the final output

Today’s lecture

GS algorithms always outputs a stable marriage

The Lemmas

Lemma 1: The GS algorithm has at most n^2 iterations

Lemma 2: S is a perfect matching

Lemma 3: S has no instability

Questions/Comments?

Extensions

Fairness of the GS algorithm

Different executions of the GS algorithm

Main Steps in Algorithm Design

Problem Statement

↓

Problem Definition

↓

Algorithm

↓

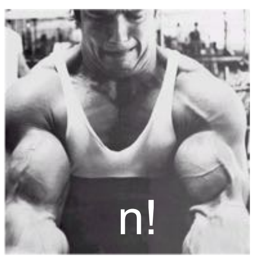
"Implementation"

↓

Analysis

NRMP

National Resident Matching Program



Correctness Analysis

Definition of Efficiency

An algorithm is efficient if, when implemented, it runs quickly on real instances

Implemented where?

Platform independent definition

What are real instances?

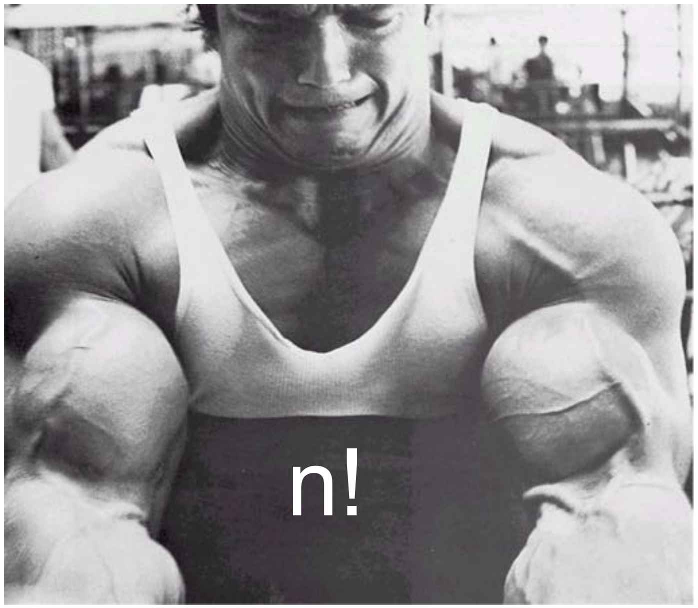
Worst-case Inputs

$N = 2n^2$ for SMP

Efficient in terms of what?

Input size N

Definition-II



Analytically better than brute force

How much better? By a factor of 2?

Definition-III

Should scale with input size

If N increases by a constant factor, so should the measure

Polynomial running time

At most $c \cdot N^d$ steps ($c>0$, $d>0$ absolute constants)

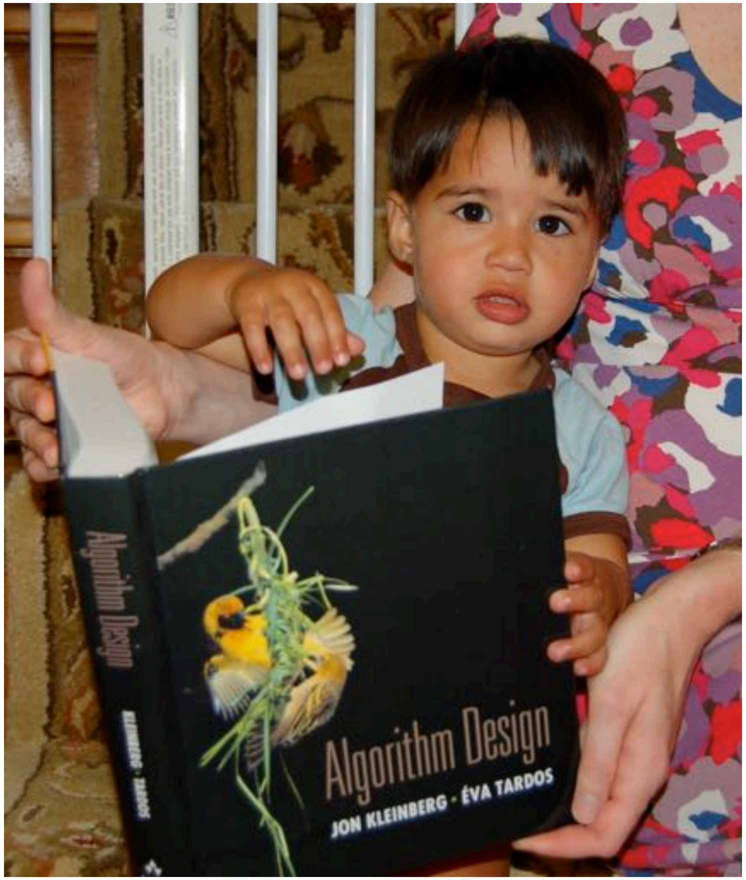
Step: "primitive computational step"

More on polynomial time

Problem centric tractability

Can talk about problems that are not efficient!

Reading Assignments



Sections 1.2, 2.1, 2.2 and 2.4 in [KT]