CSE 135: Introduction to Theory of Computation Course overview and Administrivia

Sungjin Im

University of California, Merced

01-20-2015

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Part I

Quick Survey

(Discrete) Math Background

▶ I know how to prove $\sum_{i=1}^{n} i = n(n+1)/2$ by induction.

- Yes, sure!
- Yes, probably.
- ► No.

(Discrete) Math Background

 I understand the definition of injective and surjective functions.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへで

- Yes, sure!
- Yes, probably.
- ► No..

(Discrete) Math Background

I know that the two sets of natural numbers and integer numbers have an "equal" size, and can explain why.

▲□▶ ▲圖▶ ★ 国▶ ★ 国▶ - 国 - のへで

- Yes, sure!
- Yes, probably.
- ► No.
- No clue.

I understand the asymptotic notation. For example, I understand why $10^{10^{10}} n \log n$)-runtime is asymptotically better than n^2 -runtime.

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

- Yes.
- ► No.

Given a set S of size k, I know the number of subsets of S is 2^k , and can expalin why

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

- Yes, sure!
- Yes, probably.
- No.

Other courses taken

I've taken CSE100 Algorithm Design and Analysis.

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

- I'm taking it.
- I haven't.
- I haven't, and no plan to take it.

Other information

• I've heard of the problem "P = NP?".

► No.

Other information

I've heard of the Halting problem.

No.

Part II

Administrivia

<□ > < @ > < E > < E > E のQ @

Lecture/Discussion Time and Venue

► Lecture: Classroom 113, 4:30-5:45pm, TR.

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Discussion: SSM 104, 7:30-8:20am, R.

Instructional Staff

Instructor:

- Sungjin Im (sim3)
- Office: SE2 214
- Office Hour: TBD. See course webpage.
- Teaching Assistant:
 - Maryam Shadloo (mshadloo)
 - Office: TBD
 - Office Hour: TBD. See course webpage

* You can email the instructor or TA to ask a question. But if it is not confidential, I encourage you to use UCMCROPS.*

Electronic Bulletin Boards

Webpage:

faculty.ucmerced.edu/sim3/teaching/spring15

UCMCROPS:

ucmcrops.ucmerced.edu/portal/site/201510-14582-CSE-135-01

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Textbooks

- Prerequisites: CSE-115 Discrete Mathematics
- Main Reference: Introduction to the Theory of Computation
 ** 2nd ** edition by Michael Sipser
- Lecture Notes: Available on the web-page
- Other References
 - Introduction to Automata Theory, Languages, and Computation by Hopcroft, Motwani, and Ullman
 - Elements of the Theory of Computation by Lewis, and Papadimitriou

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Grading Policy: Overview

Total Grade and Weight

- Attendance: 5%
- Homeworks: 20%
- Midterm 1: 25%
- Midterm 2: 20%

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

► Final: 30%

Attendance

- Helps you see a high-level picture.
- You can miss up to 4 lectures and 2 discussion sections with no penalties. Here absence due to illiness, academic travel, etc. does not count, but you must provide the instructor or TA with the proof.

- Then -0.5% for every lecture class you miss.
 - Ex) Misses 5 lectures and 4 discussions sections:
 5 ((5 4)+ (4 2))/2% = 3.5%
- Attendance sheet will be circulated during the class.

Homework

- Approximately 12 homeworks will be released.
 - A student suggested "more homeworks with less problems."
- Each homework's due date will be clearly specified. But in most cases, homework will be released every Tuesday and will be due the following Tuesday, 4:30pm.
- No late homeworks. Two Lowest homework scores will be dropped.
- Homeworks may be solved in groups of size at most 3, and only one submission is needed from each group.

Read homework guidelines on the course website.

Exams

- ▶ First Midterm: TBD. Will be during the class time.
- Second Midterm: TBD. Will be during the class time.
- ► Final: 8:00-11:00am, Tue, May 12 / Classroom 113.
- Midterm2 will only cover materials Midterm1 doesn't cover.

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Final Exam will cover all the course materials.

Part III Why CSE 135?

Ever-growing Data





Different types of structured and unstructured data.

WHAT ARE THE VOLUMES OF DATA THAT WE ARE SEEING TODAY?



30 billion pieces of content were added to Facebook this past month by 600 million plus users.

🞢 zynga

Zynga processes 1 petabyte of content for players every day; a volume of data that is unmatched in the social game industry.

You Tube

More than 2 billion videos were watched on YouTube... yesterday.

LOL!

The average teenager sends 4,762 text messages per month.

32 billion searches were performed last month... on Twitter.

WHAT DOES THE FUTURE LOOK LIKE?

Worldwide IP traffic will quadruple by 2015.



By 2015, nearly **3 billion people**



)

will be online, pushing the data created and shared to nearly **8 zettabytes.**

http://visual.ly/what-big-data

What to Compute?



http://aimblog.uoregon.edu/tag/social-network/

・ロト ・聞ト ・ヨト ・ヨト

э

Efficient Algorithms Needed

Speed (running time)

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Memory

▲□▶▲圖▶▲≣▶▲≣▶ ≣ のへの



▲ロト ▲理ト ▲ヨト ▲ヨト 三ヨー 釣んぐ



▲ロト ▲理 ト ▲ ヨ ト ▲ ヨ - ● ● ● ●





◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ



◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 - のへぐ

Formal Definition of Computers Needed



On January 24th, Apple Computer will introduce Macintosh. And you'll see why 1984 won't be like "1984". ... From Apple 1984 Super Bowl Commercial Introducing Macintosh Computer:

Three models

The three main computational models/problem classes in the course $% \left({{{\rm{D}}_{{\rm{D}}}}_{{\rm{D}}}} \right)$

Computational Model	Applications
Finite State Machines/ Regular Expressions	text processing, lexical analysis, protocol verification
Pushdown Automata/ Context-free Grammars	compiler parsing, software modeling, natural language processing
Turing machines	undecidability, computational complexity, cryptography

Part IV

Three Tales of Computation ...

The three Tales

- What is the nature of infinity?
- How do we learn language?
- How does the human brain work?

These diverse threads of scientific enquiry led to an understanding of computation.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

The first story

Understanding Infinity One machine to solve them all ...

◆□ ▶ < 圖 ▶ < 圖 ▶ < 圖 ▶ < 圖 • 의 Q @</p>

Infinitum actu non datur

There are no *actual* infinities; only *potential* infinities.

"I protest against the use of infinite magnitude as something completed, which is never permissible in mathematics. Infinity is merely a way of speaking, the true meaning being a limit which certain ratios approach indefinitely close, while others are permitted to increase without restriction.": Gauss

"There are more primes than in any given collection of prime numbers": Euclid



Aristotle

To Infinity and Beyond! Georg Cantor (1845–1918)

- Laid the foundations of the theory of infinite sets
- Developed the theory of infinite ordinals (numbers)
- Showed how the size of (infinite) sets could be measured
 - Showed there were more real numbers than natural numbers
 - Presaged ideas that would later show that very few problems can actually be solved computationally



Georg Cantor
Devil

Cantor's work received widespread opposition during his time

- Theologians saw Cantor's work as a challenge to the uniqueness of absolute infinity in the nature of God
- Poincaré called it the "great disease" infecting mathematics
- Kronecker called Cantor a "charlatan", a "renegade", and a "corrupter of youth"!

Devil or Messiah?

Cantor's work received widespread opposition during his time

- Theologians saw Cantor's work as a challenge to the uniqueness of absolute infinity in the nature of God
- Poincaré called it the "great disease" infecting mathematics
- Kronecker called Cantor a "charlatan", a "renegade", and a "corrupter of youth"!

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

But, it also had some supporters

"No one shall expel us from the Paradise created by Cantor": Hilbert

Crisis in Set Theory

Bertrand Russell (1872-1970)

- Mathematician, philosopher, writer, and political activist who won the Nobel prize in literature!
- Discovered disturbing paradoxes in Cantor's theory



Bertrand Russell

Russell's Paradox

Riddle: "In a small village, the barber shaves all the men who do not shave themselves (and only them). Does the barber shave himself?"

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへで

Russell's Paradox

- Riddle: "In a small village, the barber shaves all the men who do not shave themselves (and only them). Does the barber shave himself?"
- Russell: "Consider the set A of all sets that are not members of themselves. Is A a member of itself?"

Axiomatic Method to the Rescue

David Hilbert (1863-1943)

- Solution to the crises: Formalism that avoids the paradoxes
 - Define concepts precisely
 - Define axioms and rules of inference that can be used to write down formal proofs



David Hilbert

Formal Proofs

Euclid of Alexandria (around 300 BCE)

- Euclid's Elements sets out
 - Axioms (or postulates), which are self evident truths, and
 - Proves all results in geometry from these truths formally



Euclid of Alexandria

Euclid's Postulates

- A1 A straight line can be drawn from any point to any point.
- A2 A finite line segment can be extended to an infinite straight line.
- A3 A circle can be drawn with any point as center and any given radius.
- A4 All right angles are equal.
- A5 If a straight line falling on two straight lines makes the interior angles on the same side less than two right angles, the two straight lines, if produced indefinitely, will meet on that side on which the angles are less than two right angles.

Example of a Formal Proof

Elements: Proposition 32

Proposition

The interior angles of a triangle sum to two right angles.

Proof.

- 1. Extend one side (say) BC to D [A2]
- 2. Draw a line parallel to AB through point C; call it CE [P31]
- Since AB is parallel to CE, BAC = ACE and ABC = ECD [P29]
- 4. Thus, the sum of the interior angles = $ACB + ACE + ECD = 180^{\circ}$



・ロト ・ 理 ト ・ ヨ ト ・ ヨ ト ・ ヨ

Consistency and Completeness

Proof System

Precise definition of what constitutes a proof — each line is an axiom, or is derived from previous lines by rule of inference.

Correctness of proof reduced to checking if it follows the rules of the system; no ambiguity!

Consistency and Completeness

Proof System

Precise definition of what constitutes a proof — each line is an axiom, or is derived from previous lines by rule of inference.

Correctness of proof reduced to checking if it follows the rules of the system; no ambiguity!

Consistency

A proof system is consistent if it only allows true statements to be proved; no false conclusions.

Consistency and Completeness

Proof System

Precise definition of what constitutes a proof — each line is an axiom, or is derived from previous lines by rule of inference.

Correctness of proof reduced to checking if it follows the rules of the system; no ambiguity!

Consistency

A proof system is consistent if it only allows true statements to be proved; no false conclusions.

Completeness

A proof system is **complete** if every true statement has a proof that adheres to its rules.

Agenda for the 20th Century

Hilbert's Paris Lecture (1900)

- Suggested 23 open problems to be investigated in the 20th century; some remain open to this day!
- One was to obtain a consistent and complete proof system for mathematics — axioms and rules that will allow all (and only) mathematical truths to be proved

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Shocking Discovery

 In 1930, at the annual meeting of the Society of German Scientists and Physicians, Hilbert said of his program, "We must know. We will know."

Shocking Discovery Kurt Gödel (1906–1978)

- In 1930, at the annual meeting of the Society of German Scientists and Physicians, Hilbert said of his program, "We must know. We will know."
- At one of the satellite conferences of the same meeting, Kurt Gödel pronounced that Hilbert's program was fated to fail!



Kurt Gödel

Incompleteness Theorem: Gödel showed that given any consistent proof system for number theory, one can construct a statement about numbers that is true but cannot be proved!

- Relied on the Liar's Paradox which says "This statement is false."
- To get incompleteness, Gödel constructed the statement "This statement is unprovable" inside number theory!

Mechanized Computation

The heart of the axiomatic method, is that proof correctness reduces to mechanical checking.

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Mechanized Computation

The heart of the axiomatic method, is that proof correctness reduces to mechanical checking.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへで

But what is mechanical checking?

Mechanized Computation

The heart of the axiomatic method, is that proof correctness reduces to mechanical checking.

- But what is mechanical checking?
- What are limits of mechanical computation?

The Computer Revolution

Alonzo Church, Emil Post, and Alan Turing (1936)







Alonzo Church

Emil Post

Alan Turing

- Church (λ-calculus), Post (Post's machine), Turing (Turing machine) independently come up with formal definition of mechanical computation that are equivalent
- Discovered problems that cannot be solved computationally

Halting Problem

 Given a description of an arbitrary computer program, decide whether the program finishes running or continues to run forever.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

Halting problem is not solvable.

The second story

Understanding Language One grammar to generate it all

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

The Problem of Language Acquisition

- Language is an important human cognitive process that allows us to share information, thoughts and subjective experiences
- Though language is complex, it is acquired and used skillfully by children

What is the mechanism behind its acquisition and use?

Behaviorism

- All things that organisms do actions, thinking, feeling are behaviors, in response to sensory input
- Behaviors are the only measurable things
- Scientific description of such behavior should not rely on internal physiological events or hypothetical constructs such as the mind

Verbal Behavior

Burrhus Skinner (1904–1990)

- The child's mind is a blank slate, and language is learned
- The learning process is a gradual change based on sensory input provided to the organism
- Thinking is a form of "verbal behavior"



B.F. Skinner

Critique of Verbal Behavior Noam Chomsky (1928–)

Behaviorist account is flawed because the underpinnings of natural language are highly abstract principles, and children acquire language without explicit instruction or environment clues to these principles.



Noam Chomsky

Humboldt: "infinite use of finite media"

Too many sentences to learn

"A person capable of producing sentences with up to 20 words, can deal with at least 10²⁰ sentences. At the rate of 5 seconds per sentence, she would need 100 trillion years (with no time for eating or sleeping) to memorize them.": Pinker

Humboldt: "infinite use of finite media" Recursive build-up

- The longest English sentence in the Guinness Book of World Records is a 1,300 word sentence in Faulkner's Absalom, Absalom! that begins "They both bore it as though ..."
- Steven Pinker thought of submitting the following record breaker: Faulkner wrote, "They both bore it as though"

But that could be easily broken by: Pinker wrote that Faulkner wrote, "They both"

Grammatical Correctness independent of Cognition

< □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > <

Ungrammatical Comprehensible Sentences

- The child seems sleeping.
- It's flying finches, they are.
- Sally poured the glass with water.
- Who did a book about impress you?
- This sentence no verb.

Grammatical Correctness independent of Cognition

Grammatical Incomprehensible Sentences

- Chomsky: "Colorless green ideas sleep furiously."
- Edward Lear: "It's a fact the whole word knows, That Pobbles are happier without their toes."

Generative and Universal Grammars

Chomsky's Solution

The key to language acquisition is learning a Generative Grammar for the language that describes

- Word categories (like nouns, verbs, etc.)
- Rules determining how categories are put together
 Chomsky's theory: A core Universal Grammar is innate to all humans.

Chomsky Hierarchy

Chomsky found grammars of different complexity conveniently describe various aspects of languages.

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへで

Chomsky Hierarchy

Chomsky found grammars of different complexity conveniently describe various aspects of languages. Echoed in compiler design

- Lexical tokens described using "regular expressions"
- Language syntax described using "context-free grammars"

The third story

Understanding the Brain One machine to think it all

The Human Brain

- Central organ in the body that controls and regulates all human activity
- Believed the seat of "higher mental activity": thought, reason and abstraction

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

How does it work?

Neurons

Santiago Ramón y Cajal (1852–1934)

- Neurons are the primary functional unit of the brain and the central nervous system
- Neurons receive information at dendrites and transmit via axons
- They communicate with each other through junctions called synapses



Santiago Ramón y Cajal
Mathematical Model of Neural Nets

McCullough and Pitts (1943)





Warren Sturgis McCullough

Walter Pitts

- Came up with a mathematical model of a neuron
- Motivation to compare the computational power of networks of neurons with Turing Machines

Simple Neural Net



A Simple Neural Net

- Neurons have excitatory (circles) or inhibitory (dots) synapses
- Neuron produces 1 if the number of excitatory synapses with 1-input exceeds the number of inhibitory synapses with 1-input by at least the threshold of the neuron (number inside triangle)

Finite State Automata

- Finite automata model introduced by Huffman (1954), Moore (1956) and Mealy (1955) to model sequential circuits
- Kleene, 1956: Neural Nets of McCullough-Pitts the same as Finite Automata

Nondeterminism

Michael Rabin and Dana Scott (1959)





Michael Rabin

Dana Scott

- Introduced the notion of Nondeterministic Finite Automata
- Understanding the power of nondeterministic computation has remained a fundamental problem since then

Nondeterminism

Can explore multiple choices



Part V

Course Overview

Three Tales of Computation

- What is the nature of infinity?
- How do we learn language?
- How does the human brain work?

Computation underlies all these diverse scientific enquiries and they laid the foundations of a theory that we will explore.

Objectives

Understand the nature of computation in a manner that is independent of our understanding of physical laws (or of the laws themselves)

- Its a fundamental scientific question
- Provides the foundation for the science of computationally solving problems

Problems through the Computational Lens

Mathematical problems look fundamentally different when viewed through the computational lens

- Not all problems equally easy to solve some will take longer or use more memory, no matter how clever you are Not all problems can be solved!
- The "complexity" of the problem influences the nature of the solution
 - May explore alternate notions of "solving" like approximate solutions, "probabilistically correct" solutions, partial solutions, etc.

Course Overview

The three main computational models/problem classes in the course

Computational Model	Applications
Finite State Machines/ Regular Expressions	text processing, lexical analysis, protocol verification
Pushdown Automata/ Context-free Grammars	compiler parsing, software modeling, natural language processing
Turing machines	undecidability, computational complexity, cryptography