CSE 105
THEORY OF COMPUTATION

Winter 2022

https://cseweb.ucsd.edu/classes/wi22/cse105-a/
Learning goals
Introductions (my hope)
Introductions (reality?)
Logistics

Homework: ~1 / week; 7 over the quarter
  can work in groups!

In-class: Group discussion + clicker / zoom polls ; Read book first
  Discussion section: more examples, homework warmup

Exams:
  Midterm 1, Wednesday February 2, in class
  Midterm 2, Wednesday March 10, in class
  Final Exam Monday March 14, 7-10pm

Gradescope: Homework submission, exam return
Piazza: announcements and Q&A.
One-on-one office hours: with me and your TAs.
Grading

• Your final grade has 3 components

• Final exam: 40% (must pass to pass class)

• Midterms: 30% (two midterms)

• Homework: 30% (7 homeworks)

• There are no “participation points” this time, since we will be remote for at least some portion of the quarter, and some students may be remote for longer
How to excel

• Prepare ahead of class
  • Read assigned sections, read homework questions
• Engage in class
  • Discuss questions with your neighbors (when we are in person), look for (counter) examples
  • Go over wrong choices too!
• Reinforce after class
  • Briefly summarize what you learned
• Start homework early and work in a group
  • Tackle problems together: brainstorm, plan, and solve together
• Seek help and seek to help others, with integrity
How to excel with integrity

It's an integrity violation to...

• Ask others to give you specific HW or test answers
• Share your answers on HW or test
• Work on HW with anyone other than your HW partners
• Search the internet or other resources not provided for the class for HW solutions
• Share answers or notes while taking an exam

This not a complete list ... you are responsible for knowing and following the guidelines. Academic integrity violations will be taken seriously and reported immediately
About this class: Academic integrity

You are working on a homework question with your group members and are stuck on a question. You run into a friend who solved the problem already and shows you her solution. You look at it, but put it away before continuing the group conversation. Is this acceptable?

A. Yes  
B. No
About this class: Academic integrity

You're not sure if you are interpreting a homework problem correctly. You write a post on Piazza explaining your approach to answering it, and asking if this is the correct way to interpret the question. Is this acceptable?

A. Yes
B. No
I don’t know if you already noticed…

• But I have a strong accent
• Nothing I can do about it, but it might make it harder for you to understand me (in particular as the topics we learn will be new to you)

• In fact, educational research shows that students who have a professor with a strong accent actually learn better (maybe because they need to focus more)

• Some solutions:
  • Everything I say is recorded: https://podcast.ucsd.edu/
  • Slides are available in the class website

• If you don’t understand something, don’t wait – come to office hours. Sometimes one small clarification can make all the difference.
CSE 105's big questions

• What problems are computers capable of solving?
• What resources are needed to solve a problem?
• Are some problems harder than others?

Making a decision or computing a value based on some input
Questions about algorithms

Are they correct (for given specification)?

Is there a better approach to solving the same problem?

Does each problem have a solution?

If so, can that solution be found by an algorithm?
Questions about algorithms

Are they correct (for given specification)?
CSE 20, 21, 101

Is there a better approach to solving the same problem?
CSE 100, 101, CSE 200: complexity theory

Does each problem have a solution?
CSE 105: computability theory

If so, can that solution be found by an algorithm?
CSE 105: computability theory
Goal: find treasure hidden somewhere in grid. Describe an algorithm for each robot.

*Do you need to assume anything?*

*Do you need to keep track of any information?*

Robot 1: can move North, East, West, South.
Robot 2: can move in four diagonal directions.
Computer = algorithm?

- Your experience: Java, Python, C, etc.
- Other computer models:
  - Quantum computers
  - DNA computation
  - Supercomputers / Datacenters with parallelized and distributed computation
  - Power-sensitive computation for mobile
  - Embedded circuits

Different contexts call for different algorithms + different performance constraints
Bird's eye view of CSE 105

- Pick a model of computation
- Study what problems it can solve
- Prove its limits

DFA 50%

PDA 10%

TM 40%
Bird's eye view

• Pick a model of computation

Study what problems it can solve

• Prove its limits

Classification: is input of type A or not? e.g. is n prime? is list sorted?

Computation: for specific input, what value should I output? e.g. what's min cost of Hamiltonian tour?
Bird's eye view

- Pick a model of computation
- Study what problems it can solve
- Prove its limits

Classification: is input of type A or not?  
Decision problem

Computation: for specific input, what value should I output?  
Function problem
Bird's eye view

- Pick a model of computation
- Study what problems it can solve
- Prove its limits

Classification: is input of type A or not?
  Decision problem

Computation: for specific input, what value should I output?
  Function problem
Bird's eye view

• Pick a model of computation

Study what problems it can solve

• Prove its limits

**Classification**: is input of type A or not?

**Decision problem**

\{ w \mid w \text{ is of type A}\}

PRIME = \{ 2, 3, 5, 7, \ldots \}

SORTED = \{ <1,3>, <-1, 8, 17> \ldots\}

Decision problems are coded by sets of strings
Models of computation

- Finite automata
- Context-free grammars
- Turing machines

← algorithms with constant-size memory
Models of computation

- Finite automata
- Context-free grammars
- Turing machines
So let's get going

- **Textbook** reference: Chapter 0, Section 1.1

- **First topic:** Automata (DFA)
Automata

• Text processing
  grep, regexp

• Natural language processing

• Hardware design
  Moore machines, Mealy machines: CSE 140

• Controllers / Robots
  SPIS!

Code input as strings
Model memory using states
Example: subway turnstile

• A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.

When you approach the turnstile, will it open?

How can we model this problem as a classification question?

states: locked, unlocked
actions: token, push
Example: subway turnstile

- A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.

When you approach the turnstile, will it open?

How can we model this problem?

Alphabet: \{token entered, push\}  
Input: sequence of moves

Language: \{sequences of moves giving unlocked gate\}
Example: subway turnstile

- A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.
Example: subway turnstile

A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.

What happens if the turnstile is pushed while locked?

A. Transition to unlocked
B. Stay locked
C. Send error message
D. None of the above
Example: subway turnstile

- A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.
Example: subway turnstile

• A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.

What's the initial state of the turnstile?
A. Locked
B. Unlocked
Example: subway turnstile

- A subway turnstile is locked until a token is entered, at which point it unlocks in response to a single push, after which it locks again.
Finite automata and decision problems

- Decision problem "is w of type A?" coded by the set
  \[ \{ w \mid w \text{ is of type A} \} \]

A problem is **solved** by a finite automaton if
the machine **accepts** all w for which the answer is yes and
**rejects** all w for which the answer is no
Example: subway turnstile

Problem: Which sequences of moves will unlock turnstile?

Alphabet: \{t, p\}

Language: \{sequences of moves giving unlocked gate\} = \{ t, ppt, …\} = \{strings over alphabet that, when input to this automaton, end in Unlocked state\}
Vocabulary review

- \{ a,b,c,d,e \}  
  The set whose elements are a,b,c,d,e

- \{ a,b \}^*  
  The set of finite strings over a,b
  - Includes empty string \( \varepsilon \)
  - Includes a, aa, aaa
  - Includes b, bb, bbb
  - Includes ab, ababab, aaaaaaabbb
  - Does not include infinite sequences of a's and b's
  - Has infinitely many different elements

- | ababab | = 6  
  The length of the string ababab is 6

- | \{ a,b,c,d,e \} | = 5  
  The size of the set \{a,b,c,d,e\} is 5

- A language (over alphabet \( \Sigma \)) is a set of strings (over \( \Sigma \))
Deterministic Finite automaton

- Input: finite string over a fixed alphabet
- Output: "accept" or "reject"

The language recognized by the machine is the set of strings it accepts.

$L = \{ x \in \Sigma^* : \text{DFA accepts } x \}$
Computation of the machine on an input string

Sequence of states in the machine, starting with the initial state, determined by transitions of the machine as it reads additional input symbols.

How does the computation of the machine on a string relate to whether the string is accepted by the machine?

A. If the first state in the computation is an accept state, the string is accepted.
B. If any state in the computation is an accept state, the string is accepted.
C. If the last state in the computation is an accept state, the string is accepted.
D. If all of the states in the computation are accept states, the string is accepted.
E. None of the above
Finite automaton

- Input: finite string over a fixed alphabet
- Output: "accept" or "reject"

Example input: 0001
Finite automaton

- Input: finite string over a fixed alphabet
- Output: "accept" or "reject"

What is the computation of this machine on input 110?

A. q1, q1, q0  
B. q1, q1, q2  
C. q0, q1, q2, q3  
D. q0, q1, q1, q2  
E. None of the above
Finite automaton

- Input: finite string over a fixed alphabet
- Output: "accept" or "reject"

Is the empty string accepted?
Is the string 001 accepted?
Is the string 0011010101010 accepted?

*What is the language recognized by this machine?*
For next time

• Set up course tools: Gradescope, Piazza
• Find group members for HW1
• Read all the questions + relevant examples in the book
• Review CSE 20 / Math 109 / CSE 21 / Sipser Ch 0 as needed