CSE 105
Theory of Computation

http://cseweb.ucsd.edu/classes/sp16/cse105-ab/
CSE191 (Spring 2016)
Automata Practicum

- **Optional** Lab class to be taken concurrently with CSE 105
- **Goal:** Reinforce your understanding of Automata and Computability Theory
- **How:** implement and run mathematical definitions using a real programming language (Haskell)
- **Administrivia:**
  - Instructor: Daniele Micciancio
  - 1 Unit, P/NP grade
  - Lectures: Monday 10:00-10:50am, HSS 1128A
  - Homework: set of (short) programming assignments
Today’s Agenda

• Introductions
• What is the Theory of Computation??
• How does this course work?
• Our first computational model
Who Am I?

• **Education and Previous Experience:**
  – Ph. D., M.I.T.; CS Researcher at IBM T.J. Watson Research Center

• **Research:**
  – Compiler Optimization
  – Program Representations for Compilers

• **Teaching:**
  – CSE 105, 131, 231, ENG 100D, 100L (Global TIES)

• **When I’m not working:**
  – Playing piano, hiking, biking
And you?

• Introduce yourself to your neighbors
  – Major, year, a hobby or outside interest
  – Partner?
WHAT IS THEORY OF COMPUTATION?
What is “Theory of Computation”?

• **What is a Computer? A Computation?**
  - Current machines change over time, don’t give us limits of capability.
  - We’d like an answer that is *independent* of what we can currently build.

• **Computability**: What problems are computers capable of solving? Are there problems that are **not** solvable by any computer?

• **Complexity**: How easy or hard is a given problem to solve? Are some problems inherently harder than others?

• **Computational Models**: What resources are needed to solve a problem?
Goals for the course:

• Study and explore a series of abstracted models of computation machines
  – Start with the most limited
  – Move towards the most versatile*
  – Prove their power and limitations
  – Determine the relationships between models
  – Connect to practical applications

* Fun fact: this simple-to-complex order is not always the same order in which these were created/discovered
Incredibly Practical

- **Finite Automata**
  - Hardware design, software engineering, compilers
- **Regular Expressions**
  - A “must-have” programmer’s tool to include in your resume
  - Foundational for many programming and scripting tasks (Perl)
- **Context-Free Grammars**
  - Linguistics
  - Artificial Intelligence
  - Machine translation (English->Korean)
- **Turing Machines**
  - All modern computers and programming languages!
  - Computer architecture (CPU/instructions + RAM)
- Will help you get a head start in CSE 131 (Compilers)
- Will help you learn to communicate with precision as a computer scientist
What is “Theory of Computation”?

Breathtakingly Beautiful
HOW DOES THIS COURSE WORK?
http://cseweb.ucsd.edu/classes/sp16/cse105-ab/
Logistics

• **Exams:** Wednesday, April 20, 8 – 9:50 pm  
  Wednesday, May 11, 8 – 9:50 pm  
  Saturday, June 4, 11:30 am – 2:29 pm  
  **No Makeup exams**

• **Gradescope:** Homework submission, grades

• **Piazza:** Announcements, Q&A, contact instructors

• **TritonEd (Ted):** Register iclicker by Friday, April 1, Reading quizzes, Homework solutions

• **Office hours:** Instructors, TA and tutors on google calendar on class web site

• **Discussion sections:** Go to 1/4 Monday session each week

• **Podcast:** podcast.ucsd.edu

• **Class Web site:** For details  
  http://cseweb.ucsd.edu/classes/sp16/cse105-ab

• **No electronic device use during class (except for clickers)**
Logistics

• **Homework:** 7 assignments
  – Can work in groups of 1-3 from *either section*, can change each HW
  – Drop lowest HW score
  – **No late HW accepted**

• **Exams**
  – Drop lowest midterm score if do better on final
  – Can use one 3 in by 5 in handwritten index card for exams
  – At least 50% on Final exam to pass the class

• **Class Preparation:** Reading quizzes (9) **OR** Discussion (weekly)
  – Drop one lowest

• **Class clicker participation**
  – Credit: answer $\geq 80\%$ clicker questions in that day’s class
  – Drop 2 lowest

• **Evaluation**
  • Both *correctness* of answers and your ability to present your ideas *clearly and logically*
What do you do in this course?

• Prepare your brain for maximum in-class learning
  – Reading, reading quizzes \textit{before class}

• In class: engage ideas with your neighbors and class
  – Discuss question with your neighbor(s)
    • Turn ideas upside down and sideways, think about what common errors there might be
    – Come to agreement on the answer

• After each class, briefly summarize what you learned

• Seek help and seek to help others, with integrity
  – In class, Piazza forum, homework partners, office hours, discussion sections
  – I expect each class member to contribute to an environment of mutual support and cooperation
Change Frequency to CA*

• Original i>clicker:
  – Press and hold the power button on the remote until the “Vote Status” light begins flashing.
  – Use the A-E buttons to enter the two-letter frequency code. The A-E LEDs flash green indicating the frequency change was successful.
  – BUT REDO THIS EVERYTIME YOUR CLICKER TURNS OFF!

• i>Clicker2:
  – Press and hold the Power button until the two-letter frequency on the LCD flashes.
  – Use the A-E buttons to enter the new two-letter frequency code. A checkmark appears on the LCD indicating the frequency change was successful.

*Professor must have base station plugged in and connected. Each classroom at UCSD has an assigned frequency - on base station
Have you used clickers before?

A. Yes, in our CSE department
B. Yes, not in CSE but in a different department/school
C. No
Tips for a good group discussion

• Take turns being the first one to talk
• Once you all agree on the answer, don’t stop!
  – **Go over each wrong answer** and explain why you think it is wrong
  – Think about why somebody might be tempted to choose the wrong answer
  – What is the “trick” of the wrong answer choice?
  – Even if your neighbor has said something very clearly and correctly, it’s a good idea to **repeat it yourself**
    • **Your brain will remember better if YOU say it too**
“But professor, wouldn’t it be more efficient if you just taught us the right answer to begin with?”

- Have you ever heard of an aerobics class where the instructor did all the exercises at the front of class, while the class just watched and never tried them?
  - Me neither.
- To learn, you must do the work with your own muscle (your brain).
- We can help you through the tough spots, if you let us know!
Peer Instruction: Learning Gains in Physics Nearly Double

Crouch, C., Mazur, E. Peer Instruction: Ten years of experience and results

![Graph showing learning gains in physics over years with bars for traditional instruction and peer instruction.](image)
This class is hard!

• Expect to spend 8-10 hours/week outside of class
  – Exams 70%
  – Homework 20%
  – Class Participation 5%
  – Outside of class Preparation 5%
    • Reading quizzes OR Discussion
    • Both are recommended!

You can do this hard thing!
Learning Tips

• It takes time for learning to take place
  • Can’t build “muscle” overnight!
• Start early on a topic/assignment
  • Work till you get stuck (at least 25 minutes)
  • Then take a break!
• Switch between big picture and focused thinking
• Try to learn in “chunks”
• Tap into the visual, or create a metaphor
  • “Practice makes permanent”

From “Learning to Learn”, Barbara Oakley and Terry Sejnowski, Coursera.org
Academic Integrity in our class: NO

• Clicking in for someone who is absent
• Signing attendance sheet for absent person
• Asking others to give you specific HW answers
• Sharing answers on quiz or test
• Working on HW with anyone else than HW partners
• Searching the internet or other resources not provided for the class for HW solutions
• Sharing answers or notes while taking an exam

Details on class web site; this is not a complete list
  – You are responsible for knowing and following the guidelines

Academic integrity violations will be taken seriously and reported immediately
About Academic Integrity

You are working on a HW question with your group members and are stuck. You run into a friend (not in your group) who shows you her solution. You read it but put it away before working with your group again. Is this acceptable?

A. Yes
B. No
About Academic Integrity

You form a group of 3 to work on a HW with 3 questions. You split up the work so each student works on 1 question. After working individually for a while, you get together as a group and proofread the solutions, make some changes, and submit. Is this acceptable?

A. Yes but not advisable

B. No
About Academic Integrity

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

A. Yes
B. No

Do a private post to instructors!
Let’s get started!

AUTOMATA

COMPUTATIONAL MODELS
What Do Automata Do?

• Input strings, output “accept” or “reject”

• Every automaton is defined over an alphabet \( \Sigma \)
  - Ex: \( \Sigma = \{a, b, c, \ldots, z\} \) or \( \Sigma = \{0, 1\} \)

• What does a particular automaton do?
  - Can test any possible string made from \( \Sigma \) from start
  - Recognizes the set of all the strings that are accepted (wind up in a final state (double circle))
  This is referred to as the Language of the automaton
They have almost no memory!

DETERMINISTIC FINITE AUTOMATA

DFA
Example String: “001”

Since the input 001 reaches a final state (double circle); the DFA accepts the string 001
What is the sequence of states that are followed when running this DFA on the input “111001”?

A. \( q_0, q_1, q_2, q_2, q_3 \)
B. \( q_0, q_1, q_1, q_2, q_2, q_3 \)
C. \( q_0, q_1, q_1, q_1, q_1, q_2, q_2, q_3 \)
D. \( q_0, q_1, q_1, q_1, q_1, q_1, q_2, q_2, q_2, q_3 \)
E. Other/none/more than one
Acceptance in a DFA

Does the DFA accept input “1011”?  
A. Yes  
B. No  
C. I don’t know

Recall that the language of the DFA is the set of all strings that it accepts.
Which is the best description of the language recognized by DFA M1?

M1:

\[
\Sigma = \{ a, b \} 
\]

A. Starts with b and ends with a or b
B. Starts with a and ends with a or b
C. a's followed by b's

\[
\{ a^n b^k \mid n, k \geq 1 \}
\]
Communication: An important skill

• Describing languages clearly and concisely, like we did in the last problem, is a skill we work on frequently in this course.
• Also an issue in proof-writing: Being concise is a virtue; but leaving out important details is a problem... A conundrum!!

• Examples, practice, feedback, REPEAT!
• Homework and discussion sections are an official way of getting practice
Which state(s) should be in \( F \) (the set of “final” or “accept” states) so that DFA \( M_1 \) recognizes the language \( L_1 = \{ w \mid \text{b’s never appear after a’s in } w \} \) (\( \Sigma = \{a, b\} \))

\( M_1 \):

A. \( F = \{q_2\} \)
B. \( F = \{q_3\} \)
C. \( F = \{q_1, q_2\} \)
D. \( F = \{q_1, q_3\} \)
E. \( F = \{q_2, q_3\} \)
Regular Languages

• If A is the set of strings that DFA M recognizes (accepts):
  – A is the **language** of M
  – We write \( \text{L}(M) = A \)

• A language B is **regular** if there is some DFA M that recognizes B
Is every finite language regular?

A. No, if the language is too big, it can’t be recognized by a DFA.
B. Yes.
C. No, even some small languages are not recognized by a DFA.
D. I don’t know.
Getting started to-do list:

1. Go to the class web page below to read info on grading, reading assignments...
2. Join our class on TritonEd, gradescope, Piazza AND register your iClicker for our class on TritonEd by Friday April 1
3. Download and try out the J FLAP software for designing and simulating automata
4. Reading quiz 1 is due Wed. Mar 30 by 11:59 pm
5. Homework 1 is due Friday, April 1 by 11:59 pm

Remember: Late homework is not accepted.

http://cseweb.ucsd.edu/classes/sp16/cse105-ab/