Welcome to COMS 3261 (CS Theory)!

Toniann (Toni) Pitassi

1. <u>Course website</u>:

www.cs.columbia.edu/~toni/courses/CSTheory2023

2. <u>Courseworks</u>:

Lecture recordings

office hour Links (see calendar)

gradescope: submit homeworks

Discussion (edstem)

3. <u>Lectures</u>   MW 8:40 – 9:55

MW 10:10 – 11:25

4. <u>office hours / Review Sessions</u>

See course Calendar

5. <u>Book</u>: Sipser Intro to Theory of Computation

3rd edition (or any edition)

6. <u>Prereqs</u>

Discrete math (graphs, directed, undirected)
See chap 0

7. <u>Evaluation</u>

| | | |
|---|---|---|
| 4 homeworks | 7% each | (28%) |
| 2 tests (in class) | 36% each | (72%) |
| 10/23, 12/11 | | |

* Late homeworks not accepted
Submit to gradescope. Deadline midnight
can submit early r resubmit until deadline

## 8. Collaboration / Academic honesty

Don't search internet for solutions, related Q's

Don't copy from anyone / any source

Can collaborate with other students taking class

at end of collaboration, erase white board or paper. Don't take pics

only keep ideas from discussion.

Write down solns by yourself

Write Names of collaborators on HW

# Intro

This course is about how hard problems are

"languages" are problems. For now think of languages as problems.

We will characterize problems into some classes

- Regular languages / DFAs
- Context-free languages / PDAs
- computable/decidable languages / TMs
- Complexity theory

# Challenges

Lots of proofs

Abstract

Some concepts may not seem natural (at first)

ie. Nondeterminism

# Background Material

I. Not covered here but you need to know: (Chap 0)

Boolean Logic, graphs, sets

II Alphabet, strings/words, languages

An **alphabet** $\Sigma$ is a finite set of elements

Finite

Examples: $\Sigma = \{0,1\}$

$\Sigma = \{0,1,2,\dots,8,9\}$

A **string** $w$ (ls or) over $\Sigma$ is a finite sequence of elements from $\Sigma$

Finite

Examples $(\Sigma = \{0,1\})$: $w = \varepsilon$

$w = 0011$

A **language** $\mathcal{L}$ over $\Sigma$ is a set of strings over $\Sigma$

finite or infinite

Examples $\mathcal{L} = \emptyset$ (the empty language)

$\mathcal{L} = \{w \mid w \text{ has length } \leq 10\}$

$\mathcal{L} = \{w \mid w \text{ has an odd number of } 1\text{'s}\}$

$\mathcal{L} = \Sigma^* = \{w \mid w \text{ is any string over } \Sigma\}$

$\mathcal{L} = \{w \mid w \text{ has length} \leq 10\}$

$= \{\varepsilon, 0, 1, 10, 01, 11, 00, 000, 001, 010, \ldots, 111,$

---

$L = \{w \mid w \text{ has an odd number of 1's}\}$

$w = 001110 \in L$

$w = 10101011 \in L$

$w = \varepsilon \notin L$

$w = 11110000 \notin L$

$$\Sigma = \{0, 1\}$$

$$\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots, 111, \ldots\}$$

# String Notation

The length of a string $s \in \Sigma^*$, $|s|$, is the number of characters in the string

The empty string, denoted by $\varepsilon$, is the string of length zero

The concatenation of strings $s$ and $t$, is denoted by $s \cdot t$ or $st$

$\Sigma^k$ is the set of all strings over $\Sigma$ of length $k$

$$\Sigma^2 = \{00, 01, 10, 11\}$$

$\Sigma^*$ is the set of all strings over $\Sigma$

A language $L$ over $\Sigma$ is a subset of $\Sigma^*$

smallest language: $L = \emptyset$

Largest language: $L = \Sigma^*$

Why are languages problems?

Fix alphabet $\Sigma$ (i.e. $\Sigma = \{0,1\}$)

Problem associated with $\mathcal{L}$:

given $w \in \Sigma^*$ as input, decide if $w \in \mathcal{L}$

Input: $w \in \Sigma^*$

Output: $\begin{cases} \text{yes (accept)} & \text{if } w \in \mathcal{L} \\ \text{No (reject)} & \text{if } w \notin \mathcal{L} \end{cases}$

# Examples of Languages

$L1 = \{w \in \{0,1\}^* \mid w \text{ has an even number of 1's}\}$

$L2 = \{w \in \{0,1\}^* \mid w \text{ ends with } 011\}$

example:

$w = 1101011 \in L$

$w = 111110 \notin L$

$L3 = \{w \in \{0,1\}^* \mid w = 0^n 1^n, n \geq 1\}$

$w = 000111 \in L$

$w = 00110011 \notin L$

$L4 = \{w \in \{0,1,2\}^* \mid w = 0^n 1^n 2^n, n \geq 1\}$

$L5 = \{w \in \{0,1\}^* \mid w \text{ encodes a connected graph}\}$

$L6 = \{w \in \{0,1\}^* \mid w \text{ encodes a program that runs forever}\}$

# Examples of Languages

$L1 = \{ w \in \{0,1\}^* \mid w$ has an even number of 1's $\}$

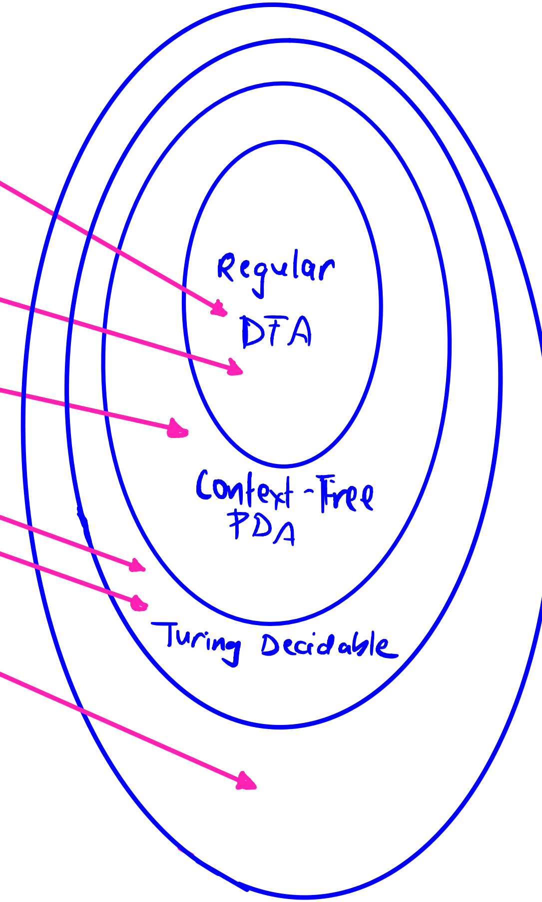
$L2 = \{ w \in \{0,1\}^* \mid w$ ends with 011 $\}$

$L3 = \{ w \in \{0,1\}^* \mid w = 0^n 1^n,\ n \geq 1 \}$

$L4 = \{ w \in \{0,1,2\}^* \mid w = 0^n 1^n 2^n,\ n \geq 1 \}$

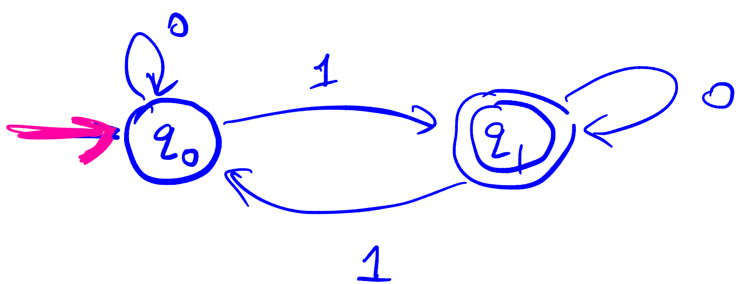
$L5 = \{ w \in \{0,1\}^* \mid w$ encodes a connected graph $\}$

$L6 = \{ w \in \{0,1\}^* \mid w$ encodes a program that runs forever $\}$



# Regular Languages and Finite Automata

Example of a DFA (over $\Sigma = \{0,1\}$)



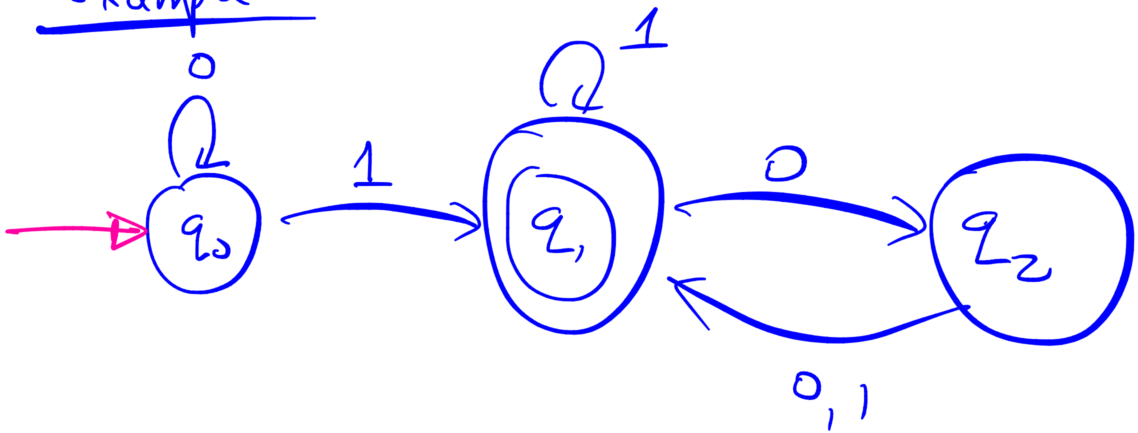

$w = 01101$ accepted
$w = 1100$ rejected
$w = 1110$

$\delta$ :

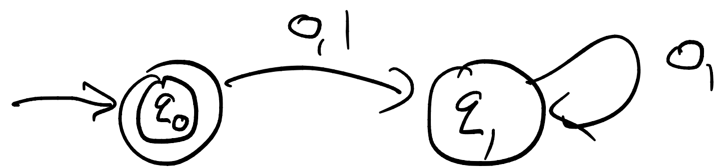
| | 1 | 0 |
|---|---|---|
| $q_0$ | $q_1$ | $q_0$ |
| $q_1$ | $q_0$ | $q_1$ |

$$M = \{ \underbrace{\Sigma = \{0,1\}}_{\text{alphabet}},\ \underbrace{Q = \{q_0, q_1\}}_{\text{set of states}},\ \underset{\text{start state}}{q_0},\ \underset{\text{accept states}}{F = \{q_1\}},\ \underbrace{\delta : Q \times \Sigma \to Q}_{\text{transition function}} \}$$

## Example 2




$L = \{\varepsilon\}$




$L = \emptyset$