CS304: Automata and Formal Languages

Lec 1

Course Intro, Administrivia, and Refresher on Basic Maths

Rachit Nimavat

July 29, 2025

Outline

What is a Computer?

- Anything that computes....

But what is it really?

- **On the surface...** Laptops, smartphones, servers, AI, ...
- Complex machines doing amazing things
- **Under the hood...** They execute instructions, process data, make decisions.
- But how? What makes them "compute"?
- **Programming!** CS101 says its all **code**.
- So, computer is just a machine that runs code...
- **Deep down...** Computers are described by incredibly elegant, abstract mathematical models: **Automata Theory**

(Tip: Appending 'theory' to almost any term makes it sound deep and legit.)

Why Study Automata Theory? (Why this course matters!)

Theory

Understanding Computation

- What problems computers can solve?
- What problems cannot be solved (even by future supercomputers)?

Understanding Computers

- Core ideas behind programming languages, compilers, interpreters, ...
- How is ‘if (x > 3) { ... }‘ understood?
- How is ‘SELECT * FROM users;‘ processed?

Practice

ML & AI

- Finite Automata $\equiv$ state machines
- Formal Languages $\equiv$ symbolic AI and knowledge representation

Formal Verification & Security

- Proving software/hardware correctness
- Analyzing protocols for vulnerabilities

Develop Algorithmic Thinking

Learn to think rigorously about problems and solutions
Essential for any serious computer scientist!

Goals of this Course

1. Learn about mathematical abstractions for computation
2. Understand relation between machine models and classes of formal languages
3. Fundamental limits of what is computable and the distinction between tractable and intractable problems
4. Ability to write formal proofs regarding properties of languages and computation

This is largely a theoretical course with substantial mathematical lining

A Caveat

Terminology/concepts in this course may feel a bit archaic/dated

They were invented before modern computers

This is the "mother tongue" of pioneers like

Alan Turing

Kurt Gödel

John von Neumann

Administrivia (tentative)

Lectures: Tue 3-4pm, Thu 2-3pm, and Fri 10-11am. Office hours: Thu 3-5pm

Homework: Expect ≈ 4 homeworks. **No Extensions.** Submit on time! But we'll drop the lowest scored homework. (Best-$(n-1)$ of n.) It is compulsory to type the homework assignments, handwritten-on-paper ones will not be accepted. You are highly encouraged to learn LaTeX (head to overleaf.com for a quick intro.)
0^{th} **problem set:** no grade (warm-up and self-diagnostic).

Quizzes: 4 in-class ≈ 10min quizzes. (Best-$(n-1)$ of n.)

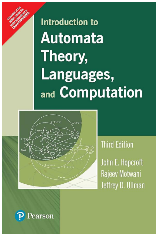
Midsems and Endsems

Have a look at the Course Policy & Grading on the course website
https://nimavat.pages.dev/courses/2025-monsoon-cs304/ You'll receive mail!

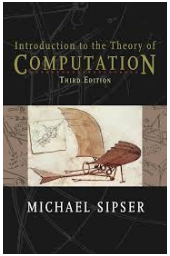
Administrivia - II (tentative)

Course website: Lecture slides and homeworks will be posted there
Google classroom: Login to your Google Classroom to submit homeworks

Textbook:
Introduction to Automata Theory, Languages, and Computation by John E. Hopcroft, Rajeev Motwani, and Jeffrey D Ullman.



You are not required to purchase the textbook. 5 copies available for in-library use.



Reference Book:
Slightly more advanced book: **Introduction to the Theory of Computation** by Michael Sipser. 5 copies should be available soon.

Sets

Defn: A **set** is an unordered collection of distinct objects (elements).

Notation:

$x \in A$ (x is an element of set A)
$y \notin A$ (y is **not** in A)
Examples:

$A = \{1, 2, 3\}$
$B = \{\text{sun}, ☁, \text{sky}\}$
$C = \{x \mid x \text{ is an even integer}\}$

$|B|$ is the number of elements in B
For $B = \{\text{sun}, ☁, \text{sky}\}$, $|B| = 3$

Special Sets:

Emptyset: $\emptyset$ or $\{\}$ (a set without any element)
Is $S = \{\{\}\}$ an emptyset?
Is $S = \{\emptyset\}$ an emptyset?

Universal Set: U or $\mathcal{U}$ (set of all elements under consideration)

Sets - II

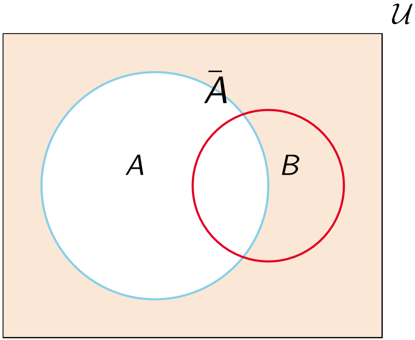
Operations

$A \cap B$

$A \cup B$

$A \setminus B$

$A^c = \bar{A} = \mathcal{U} \setminus A$



In Automata Theory, all our sets will be sets of "strings".
We call them <u>languages</u>.

Boolean Logic

Propositions

Statements that are either **True** or **False**.

"Set $\{\emptyset\}$ is not empty" (True)

"Earth is flat" (False)

"Everything happens for a reason" (**NOT a proposition!**)

For a statement to be a proposition, it must be falsifiable

Logical Connectives

Combine propositions to form more complex statements

AND: $p \wedge q$ (Both p and q)

OR: $p \vee q$ (Either p or q or both)

NOT: $\neg p$ (Also $\bar{p}$. Opposite of p's truth value)

Impl.: $p \implies q$ (pronounced p implies q) means: If p then q

Boolean Logic - II

Truth Tables

Defines truth values of a ‘statement’ for all possible truth values of its components

P	Q	$\neg P \wedge Q$
T	T	F
T	F	F
F	T	T
F	F	F

What’s in for us?

- Automata (and computers too) make decisions based on logical conditions
- Formal languages are defined using logical rules and set operations

Ensure you’re very comfortable with basic maths. Solve HW-0.

See you in the next lecture!