COSC 444/541: Automata

Lecture 1: Introductions, Sets, Functions

Today’s Agenda

- Introductions
- Go over Syllabus
- Why taking Automata?
  - What the course covers
- Break at 1:50 PM
- Sets, functions, graphs, proofs

About Me

- Yunting Yin
- PhD in CS from SUNY Stony Brook
- Office: 511C Pray-Harrold
- Email: yyin@emich.edu
- Office Hours:
  - MW 2-4PM
  - TR 11AM-12PM
  - or by appointment
- Research:
  - AI for health
  - Speech and language processing

Who are you?

1. Form groups of 2-3
2. Get to know each other's
    a. Name
    b. Class year
    c. MS/CS curriculum/CS applied/CS minor/Other major
3. Choose one of the three questions:
    a. What is a fun fact about you?
    b. What is your goal this semester?
    c. What's something recent in life you want to share?
4. Save your buddy's contact so that you know who to go to (in addition to the instructor) in case you miss a class/have questions
5. Introduce your buddy to me and to the class

Go over Syllabus

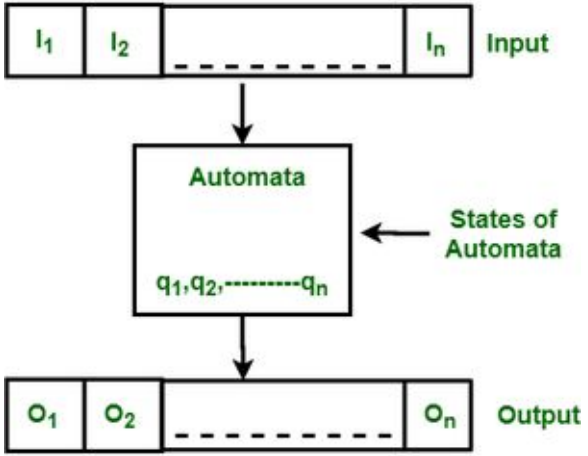
What is an automata?

An automata is an abstract mathematical model of a machine that processes input step by step and changes its internal state according to rules.

A simplified mathematical model of computation.

Think of it as a formal “robot” that:

1. Starts in an initial state.
2. Reads input symbols one by one.
3. Changes states according to a transition function.
4. Ends in either an accepting or rejecting state.



What is an automata?

Automata help us formalize what computers can and cannot do.

It answers:

- What is a computer?
- What can it do?
- What can it never do?

What is an automata?

Automata may sound abstract, but it is everywhere in real systems:

- Compilers
  - Lexical analysis, scan tokens
- Natural language processing
  - Maps between surface form -> base form, “running” -> “run”
- Search & pattern matching
- Cybersecurity
- Networking protocols
  - State machines
  - What a system should do in each state when receiving a message

Different Types of Automata

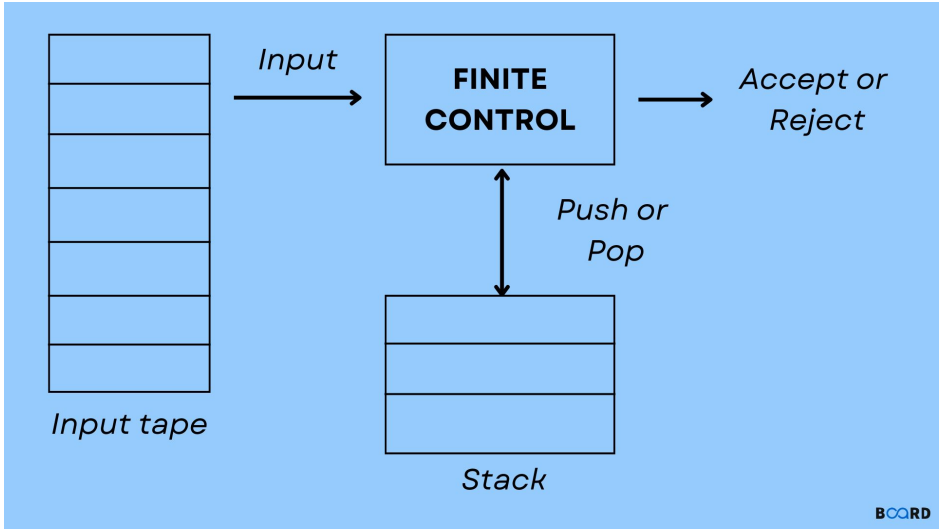
- Finite automata
- Pushdown automata
- Turing machines
- More…

Different Types of Automata

- Finite automata
  - The simplest type of automaton.
  - Has a finite number of “states” and moves between them depending on the input symbols.
    - Deterministic FA: one possible next move
    - Nondeterministic FA: multiple possible next moves
  - Example:
    - Vending machines
    - Traffic lights

Different Types of Automata

- Pushdown automata
  - Finite automata with a **stack**
  - Help you remember as you go



Different Types of Automata

- Pushdown automata
  - Finite automata with a **stack**
  - Help you remember as you go

Q: Is a stack FIFO or LIFO?

Why Automata?

Automata theory provides the mathematical backbone of computing.

Why certain problems are solvable, why some are not.

Solvable problems:

- String validation with regular expressions
  - Fixed and finite structure
- Balanced parentheses
  - Stack

Why Automata?

Automata theory provides the mathematical backbone of computing.

Why certain problems are solvable, why some are not.

Unsolvable problems:

- Decide whether a string has the same number of as, bs, and cs.
  - Finite automata has no memory to keep track of all three counts
  - Pushdown automata with one stack can handle two counts, but not three
  - Needs a more powerful machine

Why Automata?

Automata theory provides the mathematical backbone of computing.

Why certain problems are solvable, why some are not.

Unsolvable problems:

- Given a string, decide whether its length is a prime number
  - Deterministic finite automata cannot count arbitrarily long and test for primality
  - Needs a more powerful machine

Why Automata?

Abstraction skills: thinking in terms of abstract models.

Can help you understand advanced CS topics.

Sets, functions, graphs, proofs (Chapter 1.1)

We need precise mathematical tools to study automata.

Sets, functions, graphs, and proofs are those tools to describe the elements, define the transitions, visualize, and explain why some things are always true.

Sets

A set is a collection of elements.

How to write:

- x is an element of the set S
- x is not an element of the set S
- A set of integers 0, 1, 2

Sets

Ellipses are used when the meaning is clear.

How to write:

- Set of lowercase English letters
- Set of positive even integers

Set Operations

Union

Intersection

Difference

Complement - needs to know Universal Set

DeMorgan’s Laws

For two subsets of the same universal set:

1. The complement of the union is the intersection of the complements.
2. The complement of the intersection is the union of the complements.

DeMorgan's Laws

For two subsets of the same universal set:

1. The complement of the union is the intersection of the complements.
2. The complement of the intersection is the union of the complements.

In the context of automata:

Sets often represent languages – collections of strings.

Sometimes it's easier to construct an automata for the complement of a language, then apply this law to derive what you want.

More on sets

Subset

Proper subset

Disjoint

Finite: contains finite number of elements

Infinite

Powerset

Cartesian product

More on sets

Subset

Proper subset

Disjoint

Finite: contains finite number of elements

Infinite

Powerset

Cartesian product

Q: set A has 2 elements, set B has 4 elements.

What is the size of their Cartesian product?

Functions

A function is a rule that assigns to elements of one set a unique element of another set.

Total function

- Example: square

Partial function

- Example: square root

Functions

A function is a rule that assigns to elements of one set a unique element of another set.

Total function

- Example: square

Partial function

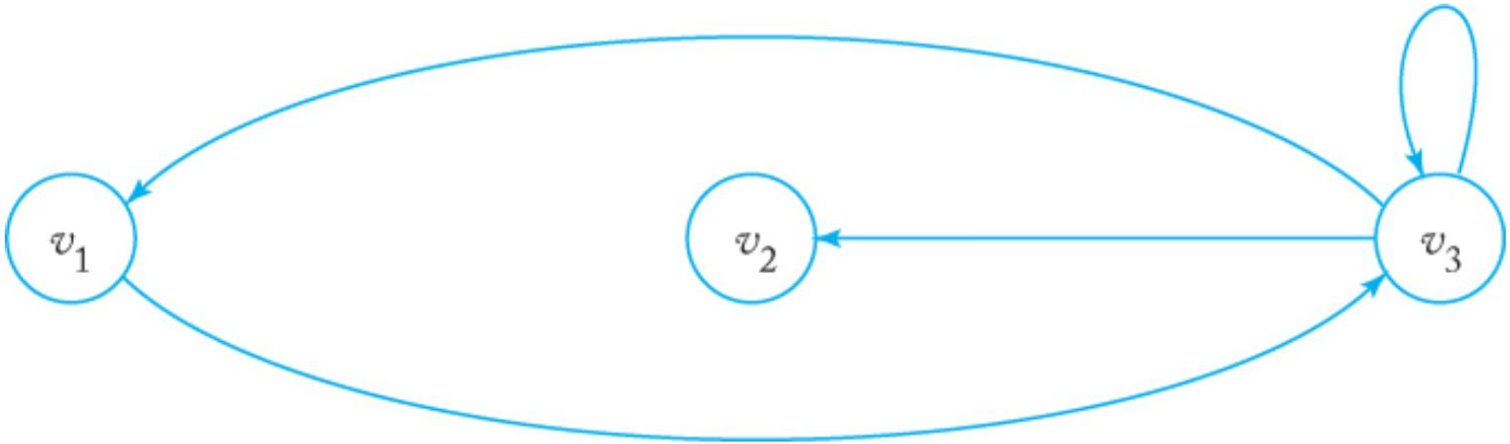
- Example: square root

In the context of automata: in DFA, every state maps to exactly one next state.

Functions guarantee determinism.

Graphs

- Vertices
- Edges



Graphs

- Vertices
- Edges

In the context of automata:

State diagrams.

Automata can be represented as directed graphs.

Graph algorithms.

In-class Exercise

Work on exercises 1-4 on page 14 on a piece of paper.

Submit at the end of class as exit token.

This will be your attendance/participation for today.

Today’s Takeaways

- We got to know each other.
- We reviewed syllabus and course expectations.
- Have an idea of what topics the course covers.
- Reviewed some math.
- No homework.

Announcements