Today's learning goals

- Use and design a finite automaton via its
  - Formal definition
  - State diagram
- Identify the strings and languages accepted by a given finite automaton
- Design a finite automaton which accepts a given language
- Define the regular operations on languages
- Prove closure properties of the class of regular languages
Review

- **Alphabet**: nonempty finite set of **symbols**
- **String over an alphabet**: finite sequence of symbols
- **Language over an alphabet**: some set of strings

- **DFA over an alphabet**: deterministic finite automaton
  - Input: finite string over a fixed alphabet
  - Output: "accept" or "reject"
  - \( L(M) = \{ w \mid M \text{ accepts } w \} \)
- **Regular language**
  - language that is \( L(M) \) for some DFA \( M \)
A finite automaton is a 5-tuple \((Q, \Sigma, \delta, q_0, F)\) where

1. \(Q\) is a finite set called the states
2. \(\Sigma\) is a finite set called the alphabet
3. \(\delta : Q \times \Sigma \rightarrow Q\) is the transition function
4. \(q_0 \in Q\) is the start state
5. \(F \subseteq Q\) is the set of accept states.

How many outgoing arrows from each state?
A. May be different number at each state.
B. Must be 2.
C. Must be \(|Q|\).
D. Must be \(|\Sigma|\).
E. I don't know.
Regular languages

- If $A$ is the set of strings that DFA $M$ recognizes (accepts)
  - We say $A$ is the **language** of $M$
  - We write $L(M) = A$
  - We have that $A$ is **regular** because...

A language is **regular** if there is some finite automaton that recognizes **exactly** it.
An example

\[(\{q_1, q_2, q_3, q_4\}, \{a, b\}, \delta, q_1, \{q_4\})\]

What's the best description of the language recognized by this DFA?

A. Start 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
D. More than one of the above
E. I don't know.

and using set notation?
An example

\( (\{q_1, q_2, q_3, q_4\}, \{a, b\}, \delta, q_1, \{q_4\}) \)

This DFA recognizes the language of all strings of the form a's followed by b's

i.e. \( \{a^n b^k \mid n, k \geq 1\} \)
What is the best description of language recognized by this automaton?

A. \{ a^n b^k | n,k \geq 1 \}
B. \{ a^n b^k | n \geq 1, b \geq 0 \}
C. \{ awb | w \text{ in } \{a,b\}^* \}
D. \{ aw | w \text{ in } \{a,b\}^* \}
E. I don't know
Specifying an automaton

( \{q_1, q_2, q_3\}, \{a, b\}, \delta, q_1, ? )

What's the best representation of $\delta$ for this DFA?
A. $q_1 \rightarrow b, q_1 \rightarrow a, q_2 \rightarrow a, q_2 \rightarrow b, q_3 \rightarrow a, b$.
B. $\{(q_1, b, q_1), (q_1, a, q_2), (q_2, a, q_2), (q_2, b, q_3), (q_3, a, q_3), (q_3, b, q_3)\}$
C. $\delta(b) =$ same, $\delta(a) =$ change
D. There's no description other than the arrows possible.
E. I don't know.
Specifying an automaton

\(( \{q1,q2,q3\}, \{a,b\}, \delta, q1, ? )\)

What state(s) should be in F so that the language of this machine is \(\{ w | ab \text{ is a substring of } w \}\)?

A. \(\{q2\}\)
B. \(\{q3\}\)
C. \(\{q1,q2\}\)
D. \(\{q1,q3\}\)
E. I don't know.
Specifying an automaton

( \{q1,q2,q3\}, \{a,b\}, \delta, q1, ? )

What state(s) should be in F so that the language of this machine is \{ w | b's never occur after a's in w\}?

A. \{q2\}
B. \{q3\}
C. \{q1,q2\}
D. \{q1,q3\}
E. I don't know.
Recall terminology

- **Alphabet**: nonempty finite set of **symbols**
- **String over an alphabet**: finite sequence of symbols
- **Language over an alphabet**: some set of strings

- **DFA over an alphabet**: deterministic finite automaton
  - Input: finite string over a fixed alphabet
  - Output: "accept" or "reject"
  - \( L(M) = \{ w \mid M \text{ accepts } w \} \)
- **Regular language**
  - Language that is \( L(M) \) for some DFA \( M \)
Regular languages: general facts

Is there an infinite regular language?

A. No: all regular languages have to be finite.
B. Yes: all regular sets are infinite.
C. Yes: all infinite sets of strings over an alphabet are regular.
D. Yes: some infinite sets of strings over each alphabet are regular and some are not.
E. I don't know.
Regular languages: general facts

Is every finite language regular?

A. No: some finite languages are regular, and some are not.
B. No: there are no finite regular languages.
C. Yes: every finite language is regular.
D. I don't know.
Regular languages: general facts

True/False: each DFA recognizes a **unique** language. I.e. if two DFA are different (different number of states or different initial state, or different transition function, etc.) then they recognize different languages.

A. True  
   **can you prove it?**

B. False  
   **can you prove it?**

C. I don't know.
Building DFA

Typical questions

* e.g. 4b, 5 on HW1

Define a DFA which recognizes the given language $L$.

or

Prove that the (given) language $L$ is regular.
Building DFA

Example
Define a DFA which recognizes

\{ w \mid w \text{ contains the substring } aba \}
Building DFA

Example
Define a DFA which recognizes

\{ w \mid \text{w does not contain the substring aba} \}
Building DFA

Example
Define a DFA which recognizes

\{ w | w has at least 2 a's \}
Building DFA

Example
Define a DFA which recognizes

\[ \{ w \mid w \text{ has at most 2 a's} \} \]
Building DFA

Remember

States are our only (computer) memory.

Design and pick states with specific roles / tasks in mind.

"Have not seen any of desired pattern yet"

"Trap state"
The regular operations

For A, B languages over same alphabet, define:

\[ A \cup B = \{ x | x \in A \text{ or } x \in B \} \]

\[ A \circ B = \{ xy | x \in A \text{ and } y \in B \} \]

\[ A^* = \{ x_1x_2 \ldots x_k | k \geq 0 \text{ and each } x_i \in A \} \]

These are operations on sets!
Closure of … on …

• Addition on \( \mathbb{Z} \).
• Multiplication on even ints.
• Concatenation on \{0\}*

Which of these is true?

A. The set of odd integers is closed under addition.
B. The set of positive integers is closed under subtraction.
C. The set of rational numbers is closed under multiplication.
D. The set of real numbers is closed under division.
E. I don't know.
Complementation

Claim: If $A$ is a regular language over $\{0,1\}^*$, then so is $\overline{A}$

Proof:
Theorem: The class of regular languages is closed under the union operation.

Proof:
For next time

Homework 1 due tomorrow
  • Set up course tools: TritonEd, Gradescope, Piazza, JFLAP
  • Keep working
  • Ask questions in office hours

Next week….  

Class of regular languages is also closed under concatenation and Kleene star, but harded to prove