INTRO TO THEORY OF COMPUTATION

LECTURE 2

Last time:
• Finite Automata

Today:
• Finite Automata
• Operations on languages
• Nondeterminism

Homework 0 due
Homework 1 out

Sofya Raskhodnikova

Sofya Raskhodnikova; based on slides by Nick Hopper
Formal definition of FA

\( M = (Q, \Sigma, \delta, q_0, F) \) where

\[ Q = \{q_0, q_1, q_2, q_3\} \]
\[ \Sigma = \{0, 1\} \]
\[ \delta : Q \times \Sigma \rightarrow Q \text{ transition function} \]
\[ q_0 \in Q \text{ is start state} \]
\[ F = \{q_1, q_2\} \subseteq Q \text{ accept states} \]
Language of FA

$L(M) = \text{the } language \text{ of machine } M$

$= \text{set of all strings machine } M \text{ accepts}$

$M \text{ recognizes the language } L(M)$
A language is **regular** if it is recognized by a finite automaton

\[ L = \{ w \mid w \text{ contains 001} \} \] is regular

\[ L = \{ w \mid w \text{ has an even number of 1s} \} \] is regular

Many interesting programs recognize regular languages

- NETWORK PROTOCOLS
- COMPILERS
- GENETIC TESTING
- ARITHMETIC
Let \( \text{TCPS} = \{ w \mid w \text{ is a complete TCP Session} \} \)

**Theorem:** \( \text{TCPS} \) is regular
Let $\Sigma_3 = \{ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \}$

- A string over $\Sigma_3$ has three ROWS.
- Each ROW $b_0b_1b_2...b_N$ represents the integer $b_0 + 2b_1 + ... + 2^Nb_N$.
- Let $ADD = \{ S \mid ROW_1 + ROW_2 = ROW_3 \}$

**Theorem.** $ADD$ is regular.
COMMENTS:

- Are delimited by /* */
- Cannot have NESTED /* */
- Must be CLOSED by */
- */ is ILLEGAL outside a comment

COMMENTS = \{\text{strings over \{0,1, /, *\} with legal comments}\}

Theorem. COMMENTS is regular.
DNA SEQUENCES are strings over the alphabet \{A,C,G,T\}.

A GENE \( g \) is a special substring.

A GENETIC TEST searches a DNA SEQUENCE for a gene.

\[ \text{GeneticTest}_g = \{ \text{strings over} \{A,C,G,T\} \text{ containing a copy of } g \} \]

**Theorem.** GeneticTest\(_g\) is regular for every gene \( g \).
Complement: \( \neg A = \{ w \mid w \notin A \} \)

Union:

Intersection:

Reverse:

Concatenation:

Star:
\[
\varepsilon \cup A \cup AA \cup AAA \cup AAAAA \cup \ldots
\]
The class of regular languages is closed under all 6 operations. If A and B are regular, applying any of these operations yields a regular language.
Theorem. The complement of a regular language is also a regular language.

Proof:

Complement: \( \overline{A} = \{ w \mid w \notin A \} \)
Closure properties

Complement: $\neg A = \{ w \mid w \not\in A \}$

Union: $A \cup B = \{ w \mid w \in A \text{ or } w \in B \}$
Theorem. The union of two regular languages is also a regular language.

Proof: Consider two regular languages $L_1$ and $L_2$. Prove that $L_1 \cup L_2$ is regular.

Let $M_1 = (Q_1, \Sigma, \delta_1, q_0^1, F_1)$ be finite automaton for $L_1$ and $M_2 = (Q_2, \Sigma, \delta_2, q_0^2, F_2)$ be finite automaton for $L_2$.

Construct a finite automaton $M = (Q, \Sigma, \delta, q_0, F)$ that recognizes $L = L_1 \cup L_2$. 
Example

M = ?

M_1

M_2

Sofya Raskhodnikova; based on slides by Nick Hopper
Idea: Run both $M_1$ and $M_2$ at the same time!

$Q = \{ (q_1, q_2) \mid q_1 \in Q_1 \text{ and } q_2 \in Q_2 \}$

$= Q_1 \times Q_2$

$q_0 = (q_0^1, q_0^2)$

$F = \{ (q_1, q_2) \mid q_1 \in F_1 \text{ or } q_2 \in F_2 \}$

$\delta( (q_1, q_2), \sigma) = (\delta_1(q_1, \sigma), \delta_2(q_2, \sigma))$
Example (continued)

\[ M_1 \]

\[ q_0 \leftrightarrow 1 \leftrightarrow 0 \]

\[ q_1 \]

\[ M_2 \]

\[ p_0 \leftrightarrow 0 \leftrightarrow 0 \]

\[ p_1 \]

\[ M \text{ intersection} \]

\[ q_0,p_0 \leftrightarrow 1 \leftrightarrow 1 \]

\[ q_1,p_0 \]

\[ q_0,p_1 \leftrightarrow 1 \leftrightarrow 1 \]

\[ q_1,p_1 \]
Closure properties

Complement: \( \overline{A} = \{ w \mid w \notin A \} \)

Union: \( A \cup B = \{ w \mid w \in A \text{ or } w \in B \} \)

Intersection: \( A \cap B = \{ w \mid w \in A \text{ and } w \in B \} \)

Reverse: \( A^R = \{ w_1 \ldots w_k \mid w_k \ldots w_1 \in A \} \)
Theorem. The reverse of a regular language is also a regular language.

Proof: Let $L$ be a regular language and $M$ be a finite automaton that recognizes it.

Construct a finite automaton $M'$ recognizing $L^R$.

Idea: Define $M'$ as $M$ with the arrows reversed. Swap start and accept states.
Closure under reverse

$M'$ IS NOT ALWAYS A DFA!

It may have many start states.

Some states may have too many outgoing edges, or none.
NONDETERMINISM

What happens with 100?

Nondeterministic Finite Automaton (NFA) accepts if there is a way to make it reach an accept state.
Nondeterministic Finite Automaton (NFA) accepts if there is a way to make it reach an accept state.
Example

$L(M) = \{1, 00\}$
$L(M) = \{w \mid w \text{ contains 101 or 11} \}$