Lecture 3 — Deterministic Finite Automata

CSE205: Theory of Computation

Minseok Jeon

2026 Fall

A Finite Automaton is a String Recognizer

A finite automaton is a simple machine that reads an input string and decides whether to *accept* or *reject* it.

We study two basic kinds:

- **Deterministic Finite Automata (DFA)**
- **Nondeterministic Finite Automata (NFA)**

This lecture focuses on the DFA.

Deterministic Finite Automata

Definition (DFA)

A *deterministic finite automaton* (DFA) is a 5-tuple

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

where

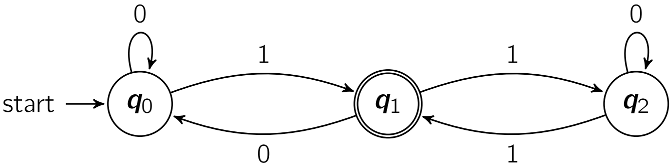
- Q: a finite set of states
- Σ: a finite set of input symbols (the input alphabet)
- $\delta : Q \times \Sigma \to Q$: a *total* function called the transition function
- $q_0 \in Q$: the initial state
- $F \subseteq Q$: a set of final (accepting) states

Example

$$M = (\{q_0, q_1, q_2\},\ \{0,1\},\ \delta,\ q_0,\ \{q_1\})$$

$$\begin{aligned}
\delta(q_0,0) &= q_0, & \delta(q_0,1) &= q_1\\
\delta(q_1,0) &= q_0, & \delta(q_1,1) &= q_2\\
\delta(q_2,0) &= q_2, & \delta(q_2,1) &= q_1
\end{aligned}$$

Transition graph:



Example: Transition Table

The same DFA can be described by a *transition table*. The arrow $\rightarrow$ marks the initial state, and $*$ marks final states.

δ	0	1
$\rightarrow q_0$	q_0	q_1
$*q_1$	q_0	q_2
q_2	q_2	q_1

Exercise

Design a DFA

Design a DFA that accepts the language

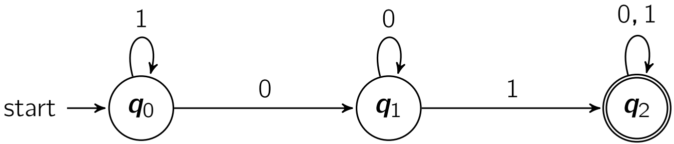
$$\{\boldsymbol{x}01\boldsymbol{y} \mid \boldsymbol{x} \text{ and } \boldsymbol{y} \text{ are any strings of 0's and 1's}\}.$$

Exercise

Design a DFA

Design a DFA that accepts the language

$$\{x01y \mid x \text{ and } y \text{ are any strings of 0's and 1's}\}.$$



Extended Transition Function

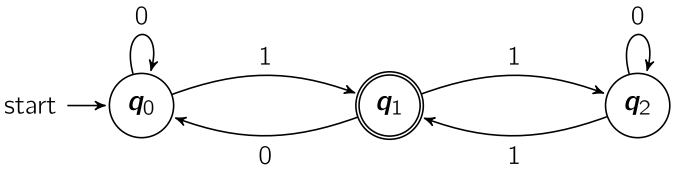
We extend $\delta : Q \times \Sigma \to Q$ from single symbols to whole strings:

$$\hat{\delta} : Q \times \Sigma^* \to Q$$

- **(Basis)** $s = \varepsilon$:
$$\hat{\delta}(q, \varepsilon) = q$$
- **(Induction)** $s = wa$ with $a \in \Sigma$:
$$\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w),\ a)$$

Example

$$M = (\{q_0, q_1, q_2\},\ \{0, 1\},\ \delta,\ q_0,\ \{q_1\})$$



Compute $\hat{\delta}(q_0, 011)$:

$$\hat{\delta}(q_0, 011) = \delta(\delta(\delta(q_0, 0), 1), 1) = \delta(\delta(q_0, 1), 1) = \delta(q_1, 1) = q_2.$$

Language of an Automaton

Definition

A DFA $\boldsymbol{M} = (\boldsymbol{Q}, \Sigma, \boldsymbol{\delta}, \boldsymbol{q}_0, \boldsymbol{F})$ *accepts* a string $\boldsymbol{w}$ if

$$\hat{\boldsymbol{\delta}}(\boldsymbol{q}_0, \boldsymbol{w}) \in \boldsymbol{F}.$$

The *language* of $\boldsymbol{M}$, denoted $\boldsymbol{L}(\boldsymbol{M})$, is the set of all strings accepted by $\boldsymbol{M}$:

$$\boldsymbol{L}(\boldsymbol{M}) = \{\boldsymbol{w} \in \Sigma^* \mid \hat{\boldsymbol{\delta}}(\boldsymbol{q}_0, \boldsymbol{w}) \in \boldsymbol{F}\}.$$

Language of an Automaton

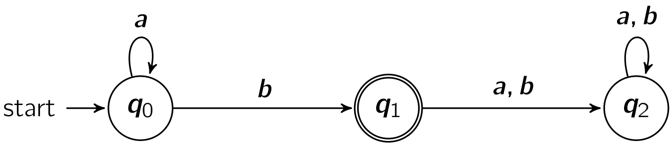
Definition

A language L is said to be *regular* iff there exists some DFA M such that

$$L = L(M).$$

Exercises

1. What is the language of the following automaton?



2. Design a DFA that accepts $L(M) = \{abw \mid w \in \{a, b\}^*\}$.
3. Design a DFA that accepts strings ending with 01:

$$L = \{w01 \mid w \in \{0, 1\}^*\}.$$