Theory of Automata and Formal Language

Lecture-6

Dharmendra Kumar
(Associate Professor)
Department of Computer Science and Engineering
United College of Engineering and Research, Prayagraj

March 31, 2021

Finite Automata

- A finite automata (FA) is the most restricted model of automatic machine. A finite automata is an abstract model of a computer system.
- Finite automata are used to recognize patterns.
- It takes the string of symbol as input and changes its state accordingly. When the desired symbol is found, then the transition occurs.
- At the time of transition, the automata can either move to the next state or stay in the same state.
- Finite automata have two states, Accept state or Reject state. When the input string is processed successfully, and the automata reached its final state, then it will accept.

Finite Automata

Finite automata can be represented by input tape and finite control.

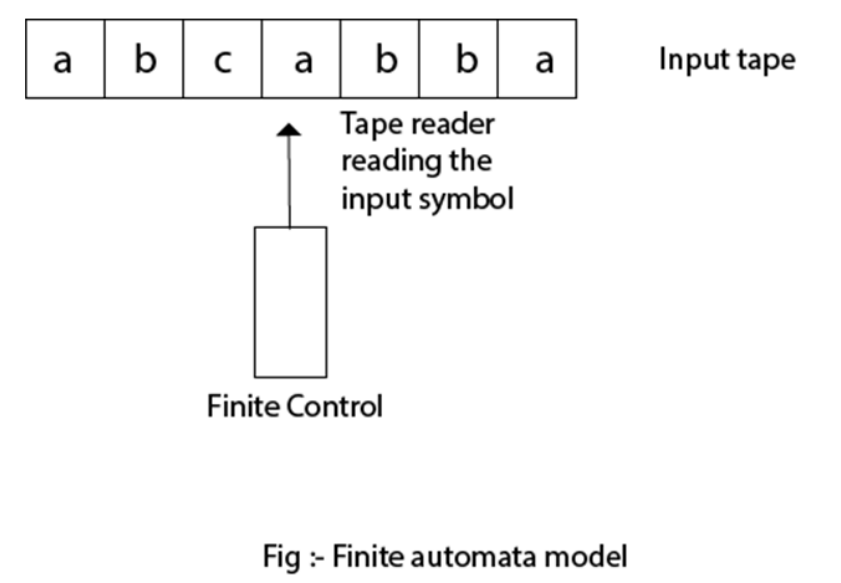


Fig :- Finite automata model

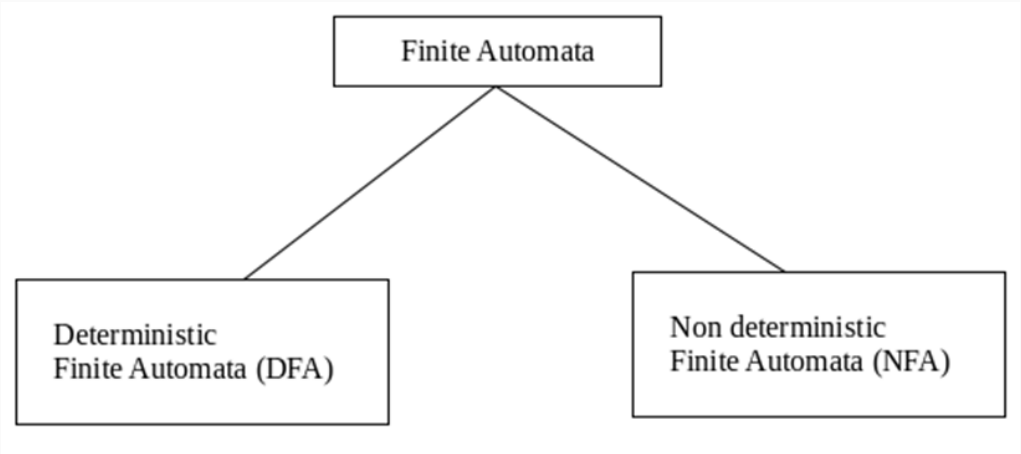
Input tape: It is a linear tape having some number of cells. Each input symbol is placed in each cell.

Finite control: The finite control decides the next state on receiving particular input from input tape. The tape reader reads the cells one by one from left to right, and at a time only one input symbol is read.

Types of Finite Automata

There are two types of finite automata:

1. **Deterministic Finite Automata (DFA):** In the DFA, the machine goes to one state only for a particular input character. DFA does not accept the null move.
2. **Non-deterministic Finite Automata (NFA):** In the NFA, the machine goes to one or more state for a particular input character. NFA can accept the null move.



Applications of Finite Automata (FA)

We have several applications based on finite automata and finite state machine. Some are given below;

- A finite automata is highly useful to design Lexical Analyzers.
- A finite automata is useful to design text editors.
- A finite automata is highly useful to design spell checkers.
- A finite automata is useful to design sequential circuit design (Transducer).

Deterministic Finite Automata (DFA)

A deterministic finite automata M is a 5-tuple, M = (Q, Σ, δ, q_0, F), where

Q $\rightarrow$ Finite set of states

Σ $\rightarrow$ Finite set of input symbols

$q_0 \in Q \rightarrow$ Initial state

F$\subseteq Q \rightarrow$ Set of final states

and $\delta \rightarrow$ Transition function

It is defined as following:-

$$\delta : Q \times \Sigma \rightarrow Q$$

Deterministic Finite Automata (DFA)

Extended Transition Function

It is denoted by $\hat{\delta}$. It is defined as following:-

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

Properties of $\hat{\delta}$

1. $\hat{\delta}(q, \epsilon) = q$
2. $\hat{\delta}(q, a) = \delta(q, a), \qquad \forall a \in \Sigma$
3. $\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a), \qquad \forall a \in \Sigma \text{ and } \forall w \in \Sigma^*$

Language Accepted by DFA

Language accepted by DFA, M is denoted by L(M). It is defined as following:-

$$L(M) = \{x \in \Sigma^* \mid \hat{\delta}(q_0, x) \in F\}$$

Representation of DFA

Any DFA can be represented by the following two ways:-

(1) By transition table

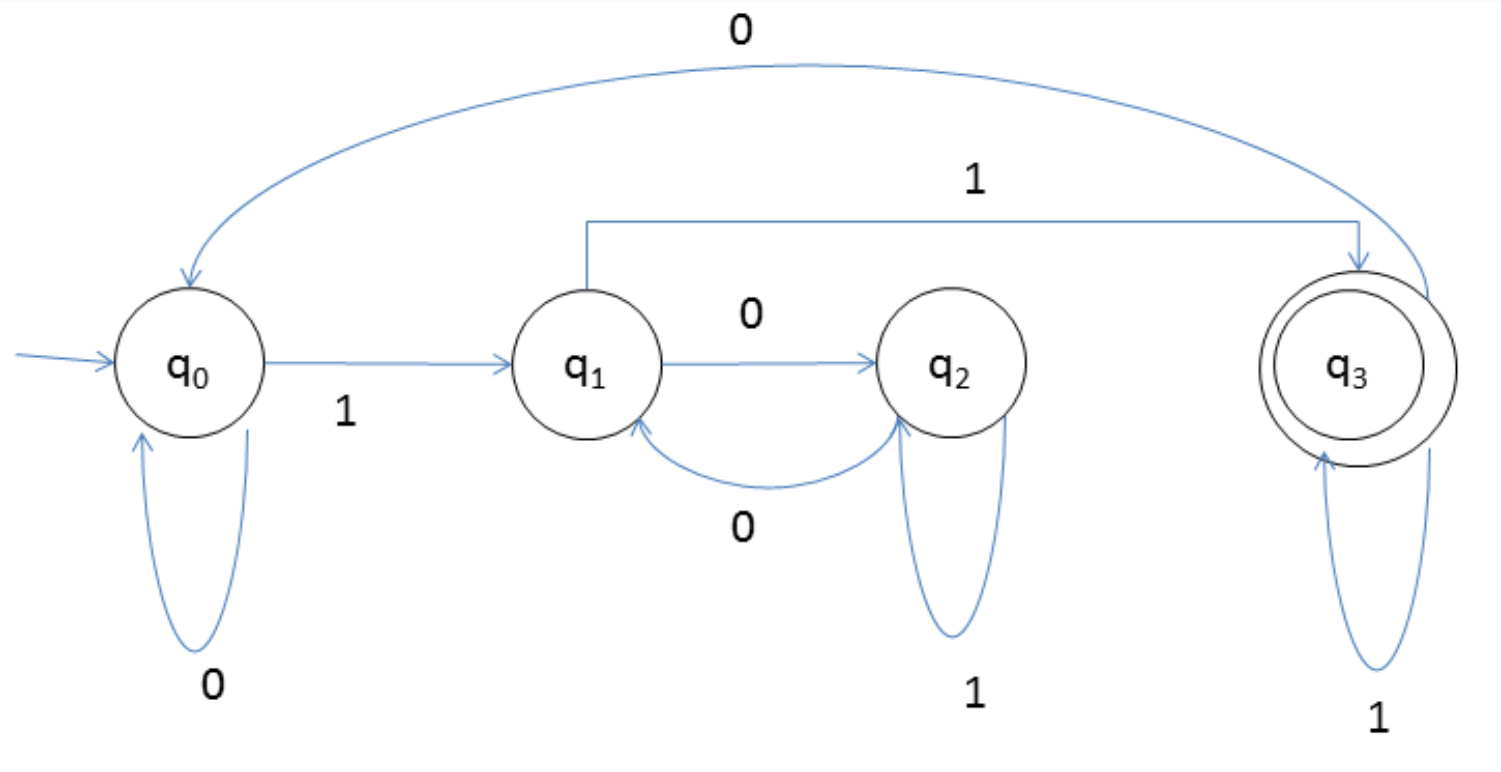
(2) By transition diagram

(1)By transition table

δ	0	1
→ q_0	q_0	q_1
q_1	q_2	q_3
q_2	q_1	q_2
(q_3)	q_0	q_3

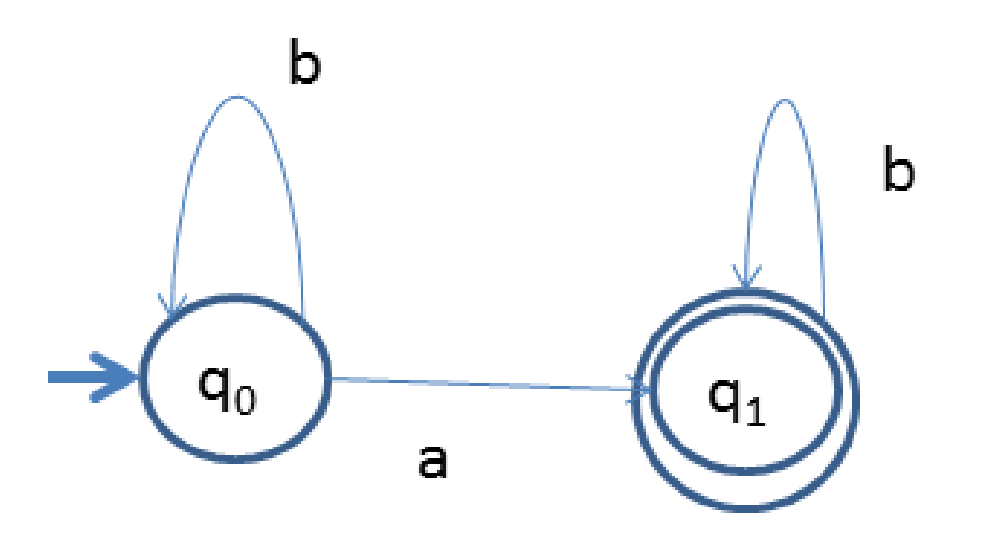
Deterministic Finite Automata (DFA)

(2) By transition diagram



Some Examples

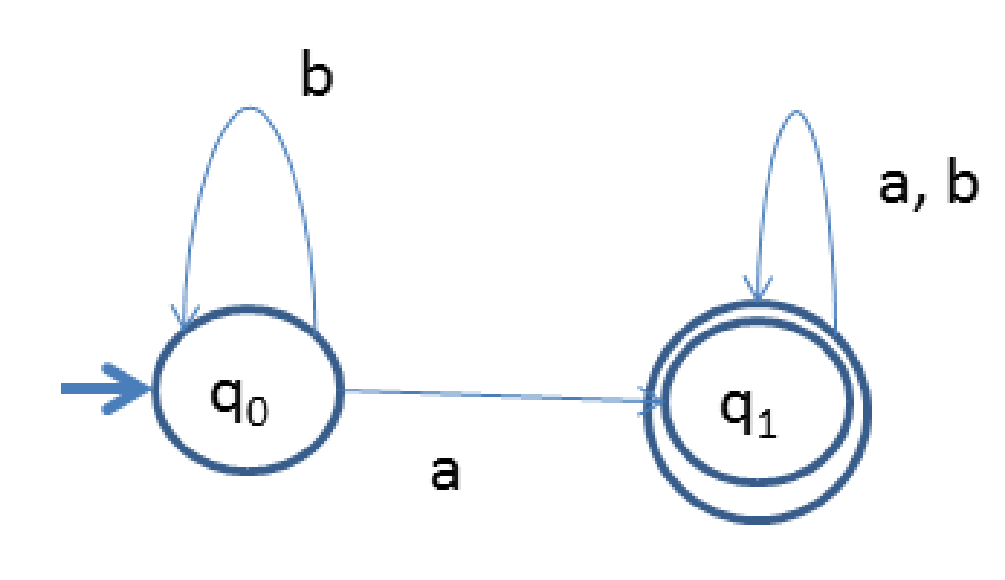
Example: Find the language accepted by following DFA:-



.

Deterministic Finite Automata (DFA)

Example: Find the language accepted by following DFA:-



Deterministic Finite Automata (DFA)

Example: Find the language accepted by following DFA:-