Automata theory (also known as the Theory of Computation) is a theoretical branch of Computer Science and Mathematics, which mainly deals with the logic of computation with respect to simple machines, referred to as automata.

- The word automata comes from the Greek word, which means "self-acting, self-moving".
- An automaton with a finite number of states is called a Finite Automaton (FA) or Finite-State Machine (FSM). This automaton consists of states (represented in the figure by circles) and transitions (represented by arrows).
- Automata theory is closely related to formal language theory. In this context, automata are used as finite representations of formal languages that may be infinite. Automata are often classified by the class of formal languages they can recognize, as in the Chomsky hierarchy, which describes a nesting relationship between major classes of automata
- Automata in many fields such as the theory of computation, compiler construction, artificial intelligence, parsing and formal verification.

Finite Automata(FA) is the simplest machine to recognize patterns. The finite automata or finite state machine is an abstract machine that has five elements or tuples. It has a set of states and rules for moving from one state to another but it depends upon the applied input symbol. Basically, it is an abstract model of a digital computer. The following figure shows some essential features of general automation.

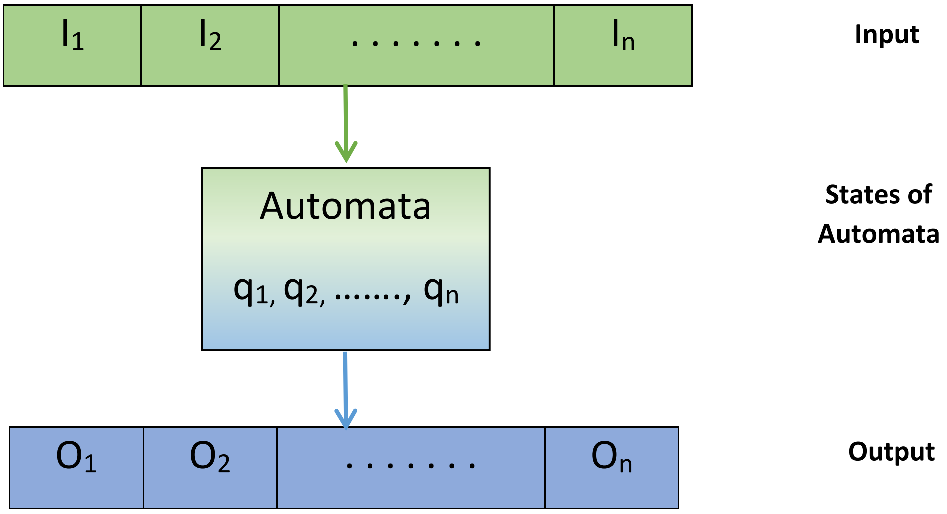


Figure: Features of Finite Automata

The above figure shows the following features of automata:

1. Input
2. Output
3. States of automata
4. State relation
5. Output relation

A Finite Automata consists of the following:

Q: Finite set of states.
Σ: a set of Input Symbols.
q: Initial state.
F: the set of Final States.
δ: Transition Function.

The formal specification of the machine is:

$$\{ Q, \Sigma, q, F, \delta \}$$

FA is characterized into two types:

1. **Deterministic Finite Automata (DFA):**

DFA consists of 5 tuples {Q, Σ, q, F, δ}.

Q: a set of all states.

Σ: a set of input symbols. (Symbols that which machine takes as input)

q: Initial state. (Starting state of a machine)

F: the set of the final state.

δ: Transition Function, defined as δ: Q X Σ --> Q.

In a DFA, for a particular input character, the machine goes to one state only. A transition function is defined on every state for every input symbol. Also in DFA null (or ε or λ) move is not allowed.

For example, below DFA with Σ = {0, 1} accepts all strings ending with 0.

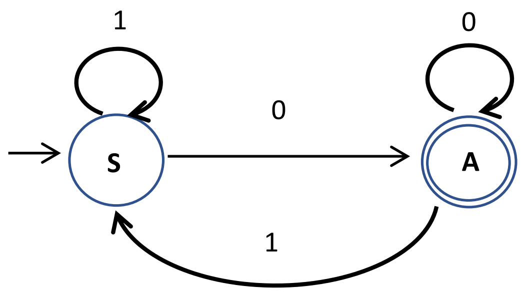


Figure: DFA with Σ = {0, 1}

2. **Nondeterministic Finite Automata(NFA):** NFA is similar to DFA except following additional features:
   - Null (or ε or λ) move is allowed i.e., it can move forward without reading symbols.
   - Ability to transmit to any number of states for a particular input.

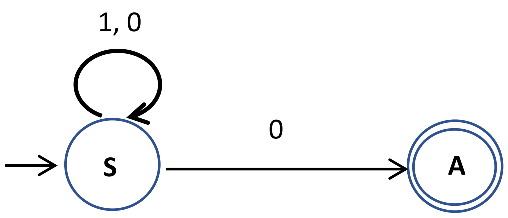
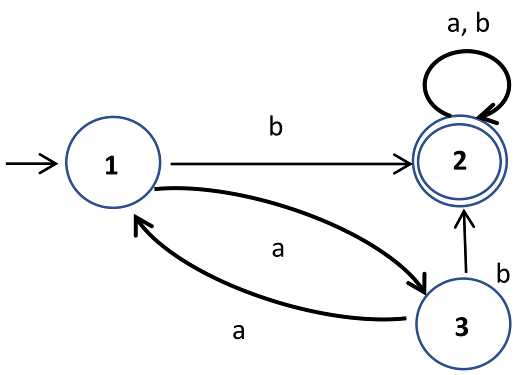


Figure: NFA with Σ = {0, 1}

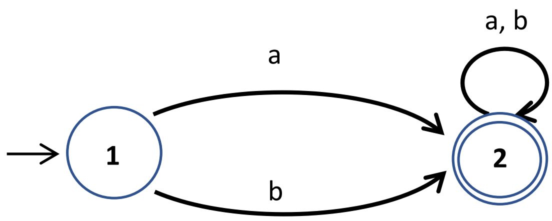
Examples of finite automata

Σ = {a, b}

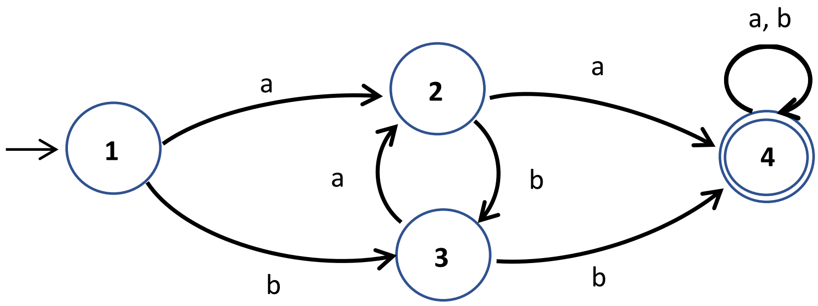
1. At least b



2. Must start with a, b



3. Must contains with aa, bb



4. Third letter b?
5. End with a?
6. End with a or nothing?
7. Double letter aa?
8. Must start with a and end with b or start with b and end with a?