Theory of Computation: CS-202
Deterministic Finite Automata

Content

Finite Automata

Deterministic Finite Accepters

- Deterministic Accepters
- Transition Graphs
- Languages and DFAs

Deterministic Finite Accepters

Definition

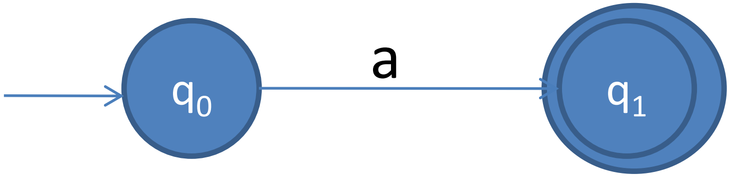
A deterministic finite accepter or dfa is defined by the quintuple

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

where Q is a finite set of internal states,
Σ is a finite set of symbols called the input alphabet,
$\delta: Q \times \Sigma \rightarrow Q$ is a total function called the transition function,
$q_0 \in Q$ is the initial state,
$F \subseteq Q$ is a set of final states.

Transition Graph

- To visualize and represent finite automata, we use transition graphs, in which vertices represents states and the edges represents transitions.
- The labels on the vertices are the name of the states, while the labels on the edges are the current values of the input symbol.



- The initial state is identified by an incoming unlabeled arrow not originated at any vertex.
- Final states are drawn with a double circle.

Example

Transition Graph of a dfa M = (Q, Σ, δ, q_0, F)

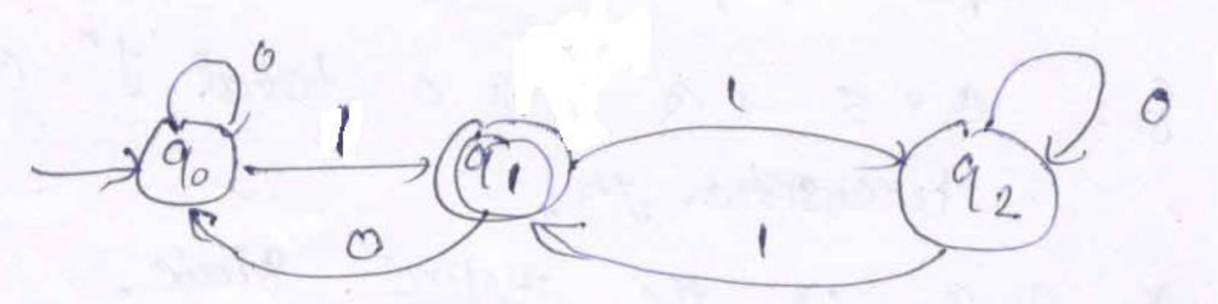
Vertex labeled with q_i: state $q_i \in Q$,

Edge from q_i to q_j labeled with a: transition $\delta(q_i, a) = q_j$.

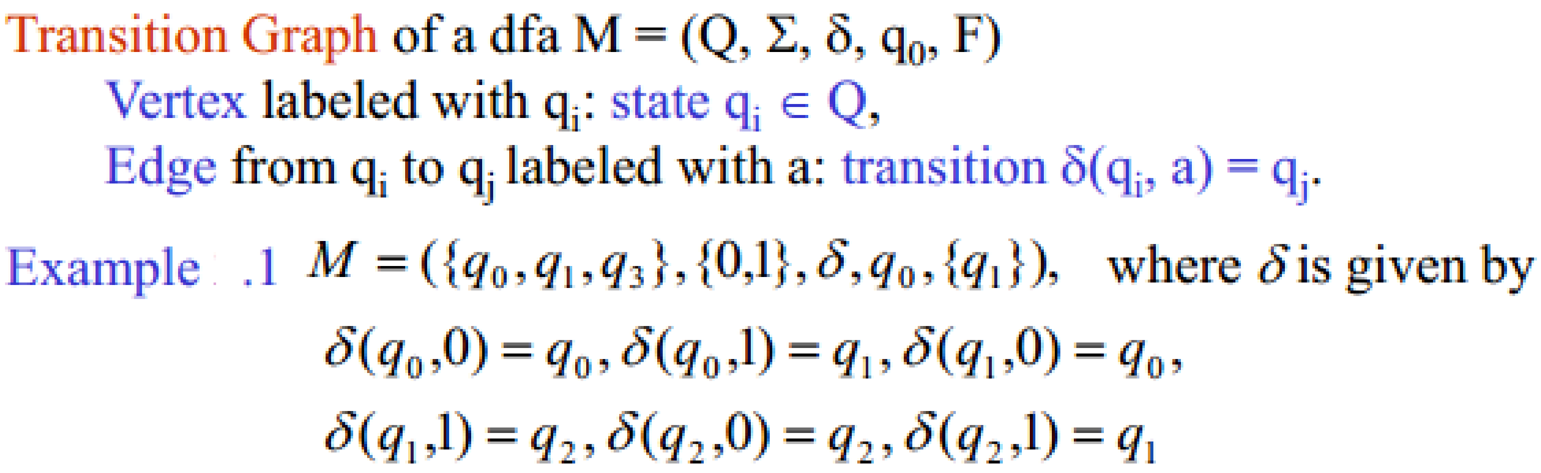
Example .1 $M = (\{q_0, q_1, q_3\}, \{0,1\}, \delta, q_0, \{q_1\})$, where δ is given by

$$\delta(q_0,0) = q_0, \delta(q_0,1) = q_1, \delta(q_1,0) = q_0,$$

$$\delta(q_1,1) = q_2, \delta(q_2,0) = q_2, \delta(q_2,1) = q_1$$



Transition Table



Transition Graph of a dfa M = (Q, Σ, δ, q_0, F)

Vertex labeled with q_i: state $q_i \in Q$,

Edge from q_i to q_j labeled with a: transition $\delta(q_i, a) = q_j$.

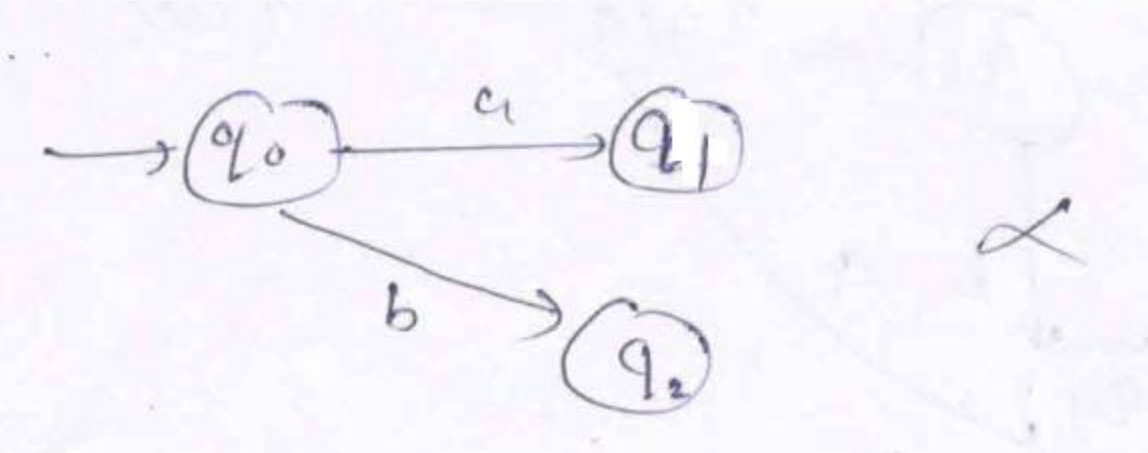
Example .1 $M = (\{q_0, q_1, q_3\}, \{0,1\}, \delta, q_0, \{q_1\})$, where δ is given by

$$\delta(q_0,0) = q_0, \delta(q_0,1) = q_1, \delta(q_1,0) = q_0,$$

$$\delta(q_1,1) = q_2, \delta(q_2,0) = q_2, \delta(q_2,1) = q_1$$

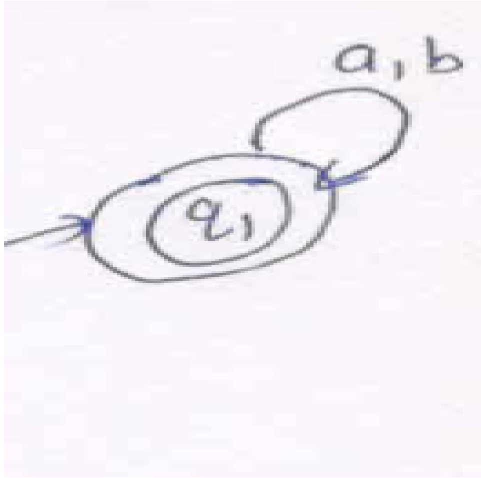
	0	1
→ q0	q0	q1
*q1	q0	q2
q2	q2	q1

- Note: Machine should be complete



- The processed symbol is remembers by changing the state.

Number of final states



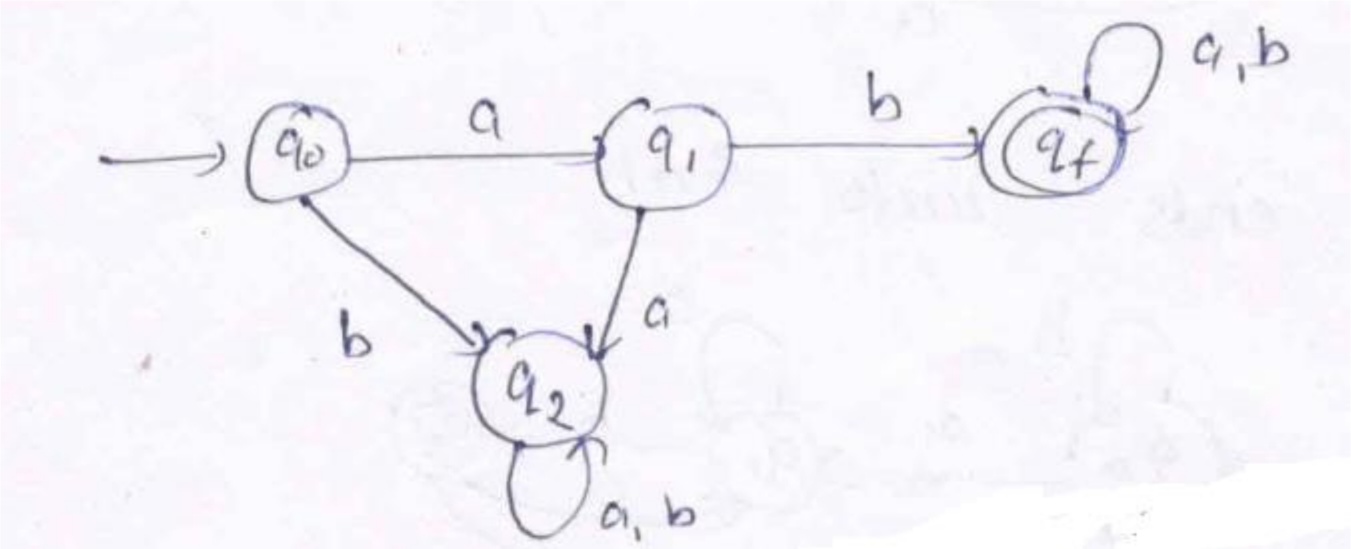
Dead and Unreachable state

Dead State: A rejecting state that is essentially a dead state.

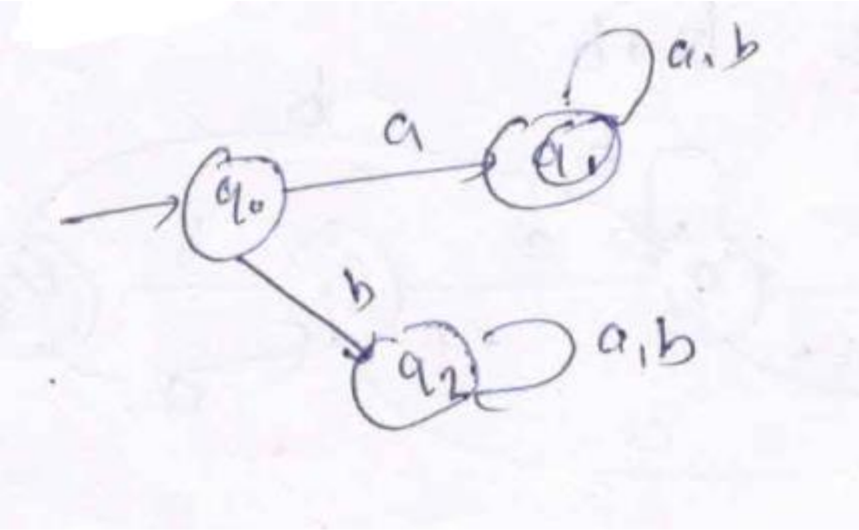
- Once the machine enters a dead state, there is no way for it to reach an accepting state.

Unreachable state: the states that are not reachable from the initial state of the DFA.

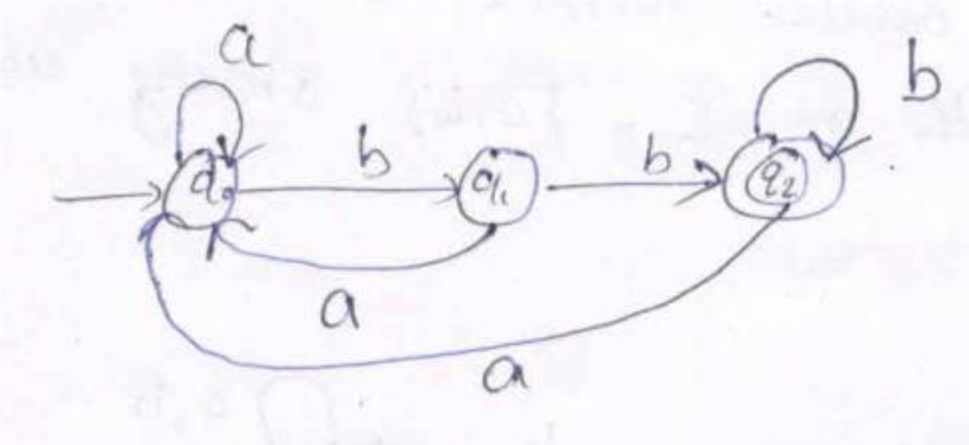
- Draw a DFA which accepts all the strings on $\Sigma=\{a, b\}$ with the prefix ‘ab’.



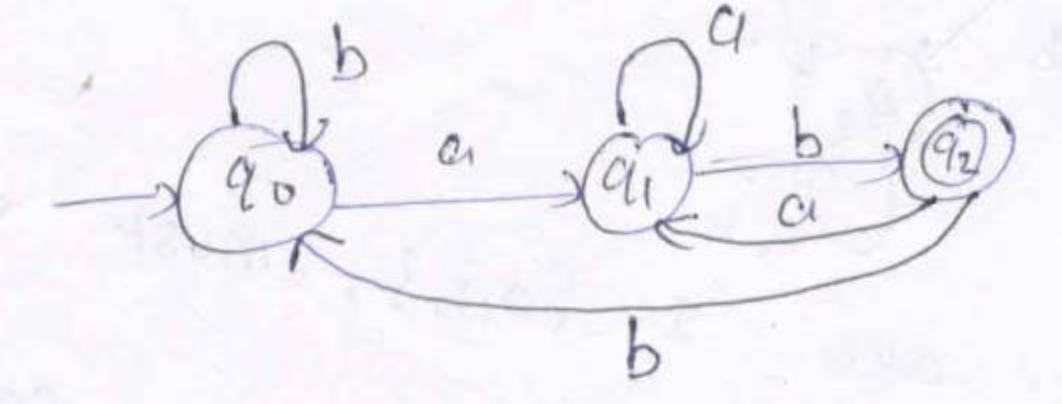
- Draw a DFA which accepts all the strings on $\Sigma = \{a,b\}$ which starts with ‘a’.



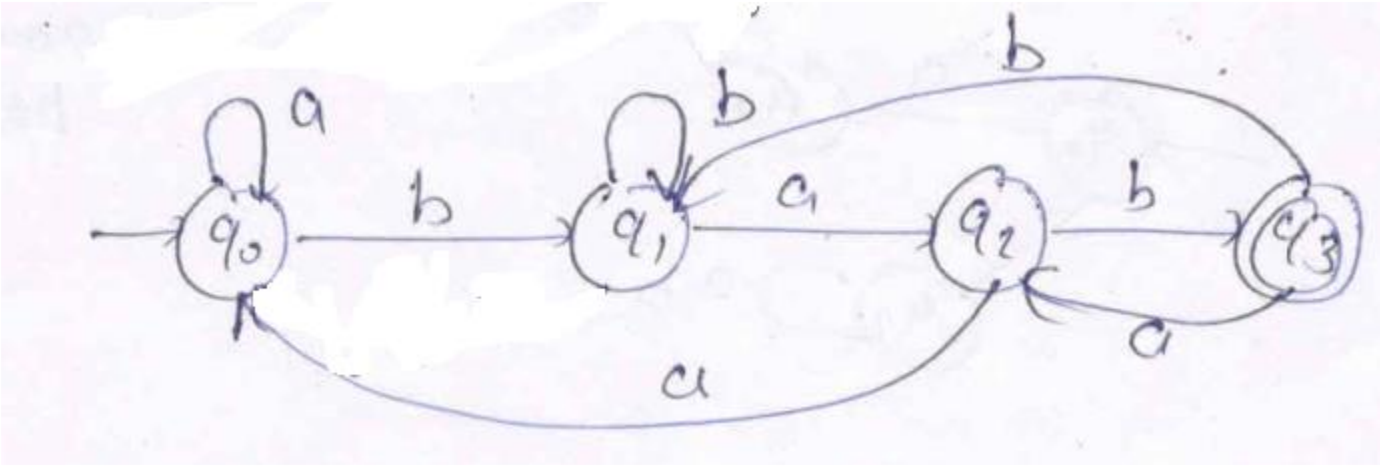
- Draw a DFA which accepts all the strings on $\Sigma=\{a, b\}$ which must end with 'bb'.



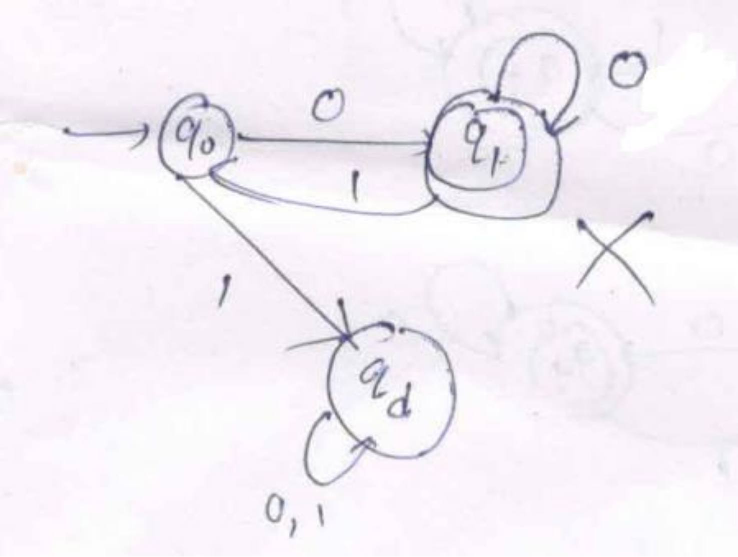
- Draw a DFA which accepts all the strings on $\sum$={a, b} which must end with ‘ab’.



- Draw a DFA which accepts all the strings on Σ={a,b} which must end with 'bab'.



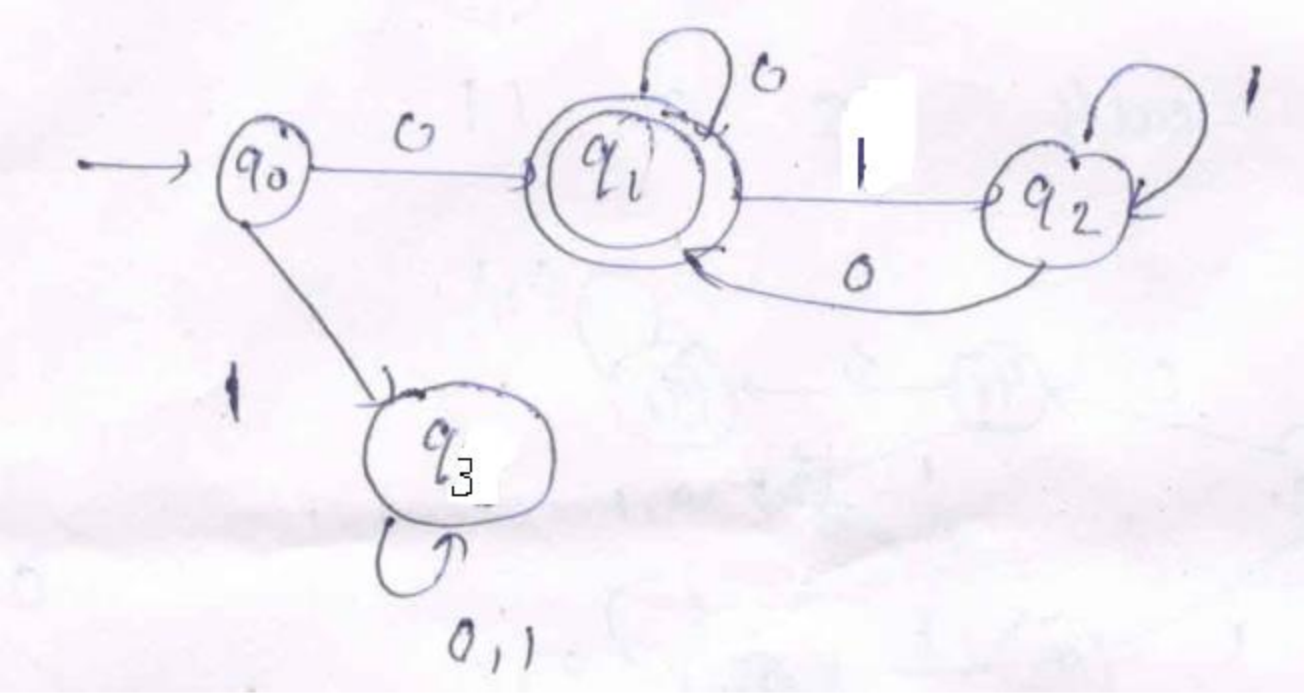
- Draw a DFA which accepts all the strings on $\sum=\{0,1\}$ which must start & end with '0'.



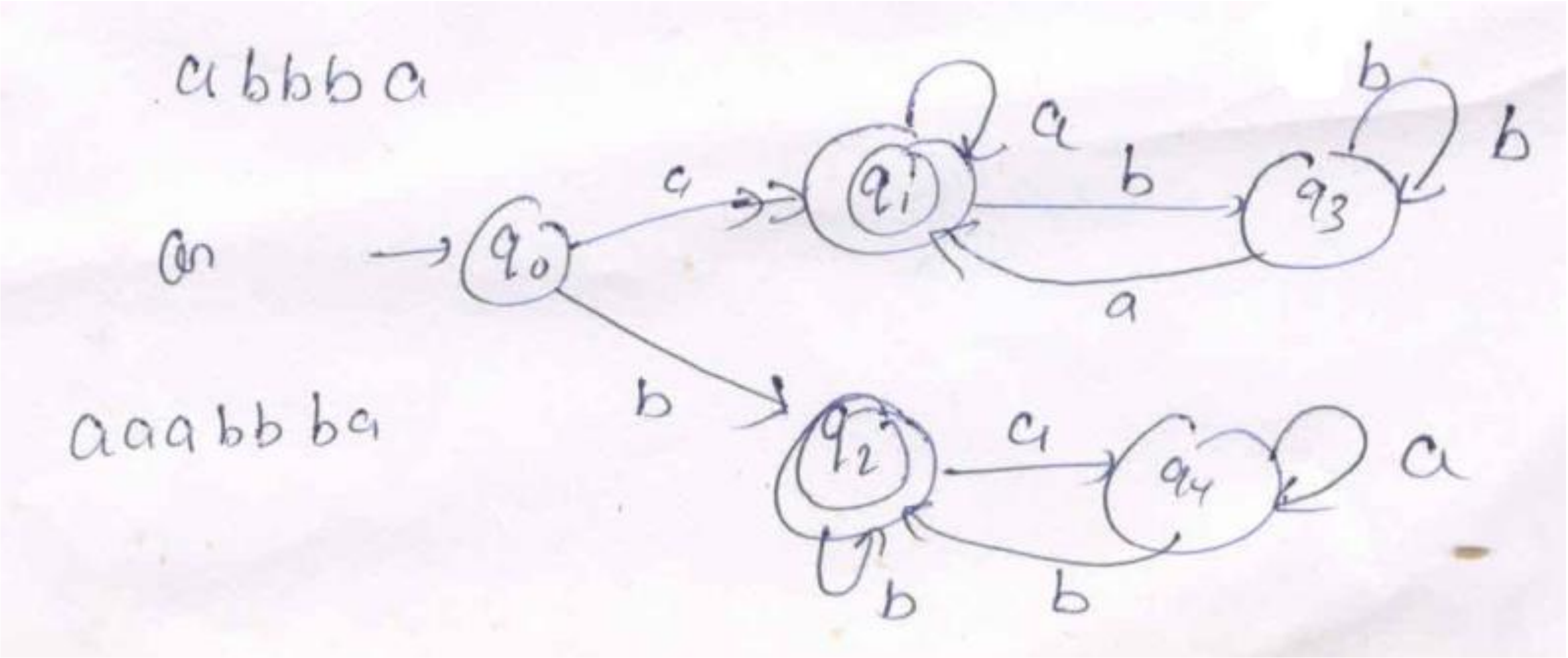
0, 00, 010

01110??

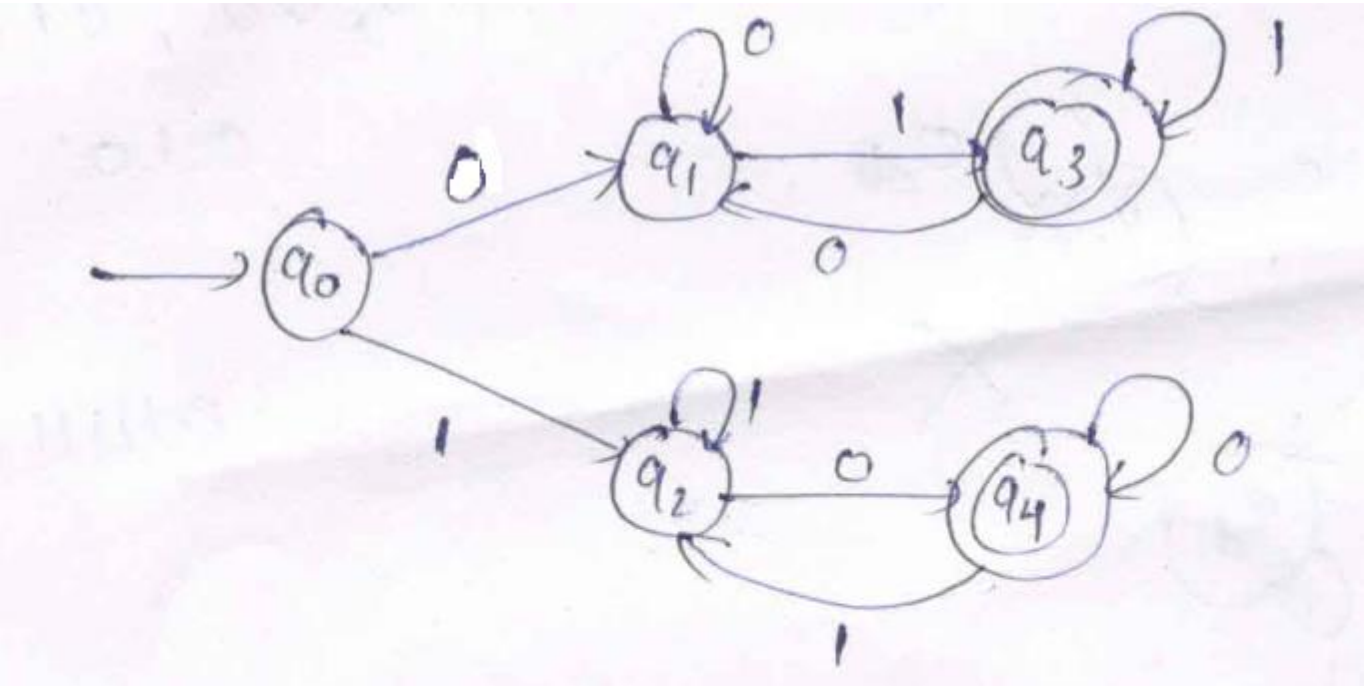
- Draw a DFA which accepts all the strings on $\Sigma=\{0,1\}$ which must start & end with ‘0’.



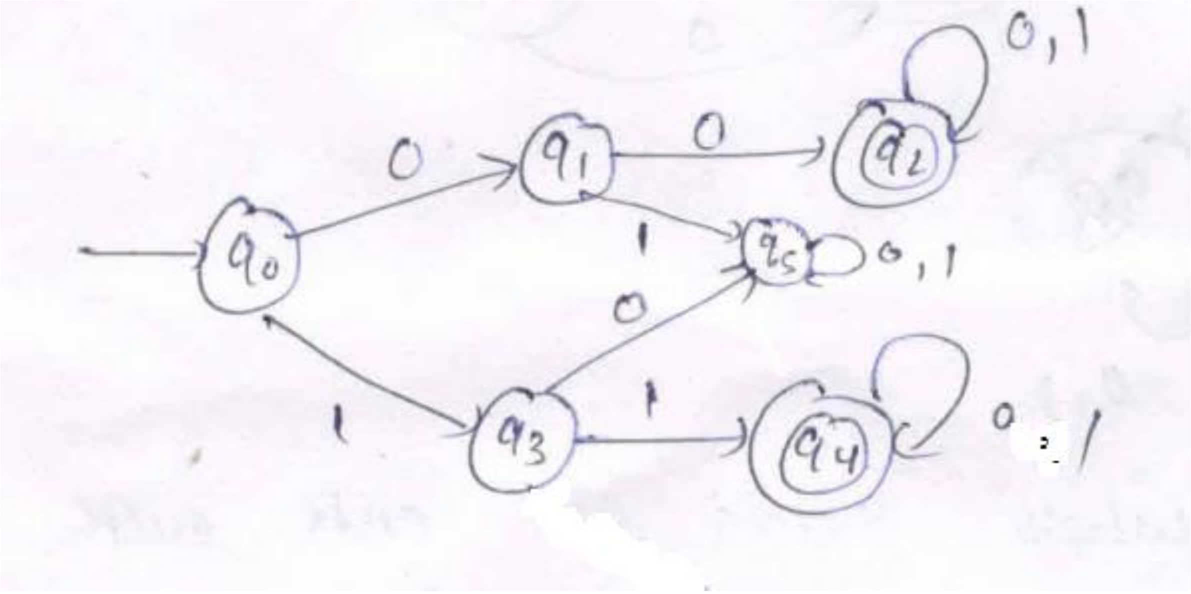
- Draw a DFA which accepts all the strings on $\Sigma=\{a,b\}$ which starts and ends with same symbol.



- Draw a DFA which accepts all the strings on $\Sigma=\{a,b\}$ which starts and ends with different symbol.



- Draw a DFA which accepts all the strings on $\Sigma=\{0,1\}$ which starts with ‘00’ or ‘11’.



Practice Problems

1. Draw a DFA which accepts all the strings on $\sum=\{0,1\}$ which never ends with ‘100’.

2. Draw a DFA which accepts all the strings on $\sum=\{0,1\}$ which contains sub-string ‘00’.

Suggested readings

1. An introduction to FORMAL LANGUAGES and AUTOMATA by PETER LINZ.
2. Introduction to Automata Theory, Languages, And Computation by JOHN E. HOPCROFT, RAJEEV MOTWANI, JEFFREY D. ULLMAN
3. Theory of computer science: automata, languages and computation **by** K.L.P MISHRA