Automata Theory(SCS2112)

Slides By

Dr Damitha D Karunaratna

Transducers

Definition : A transducer is defined by the tuple $T = (Q, \sum, \Gamma, \delta, q_0, \omega)$ where

- Q : is a finite set of **internal states**
- $\sum$: is a finite set of symbols called the **input alphabet**
- Γ : is the **output alphabet**
- $\delta : Q \times \sum \rightarrow Q$ is a function called the **transition function**
- $q_0 \in Q$ is the **initial state**
- ω : output function

$\omega: Q \times \Sigma \to \Gamma$ Mealy Model

output depends on input and state

$\omega: Q \to \Gamma$ Moore Model

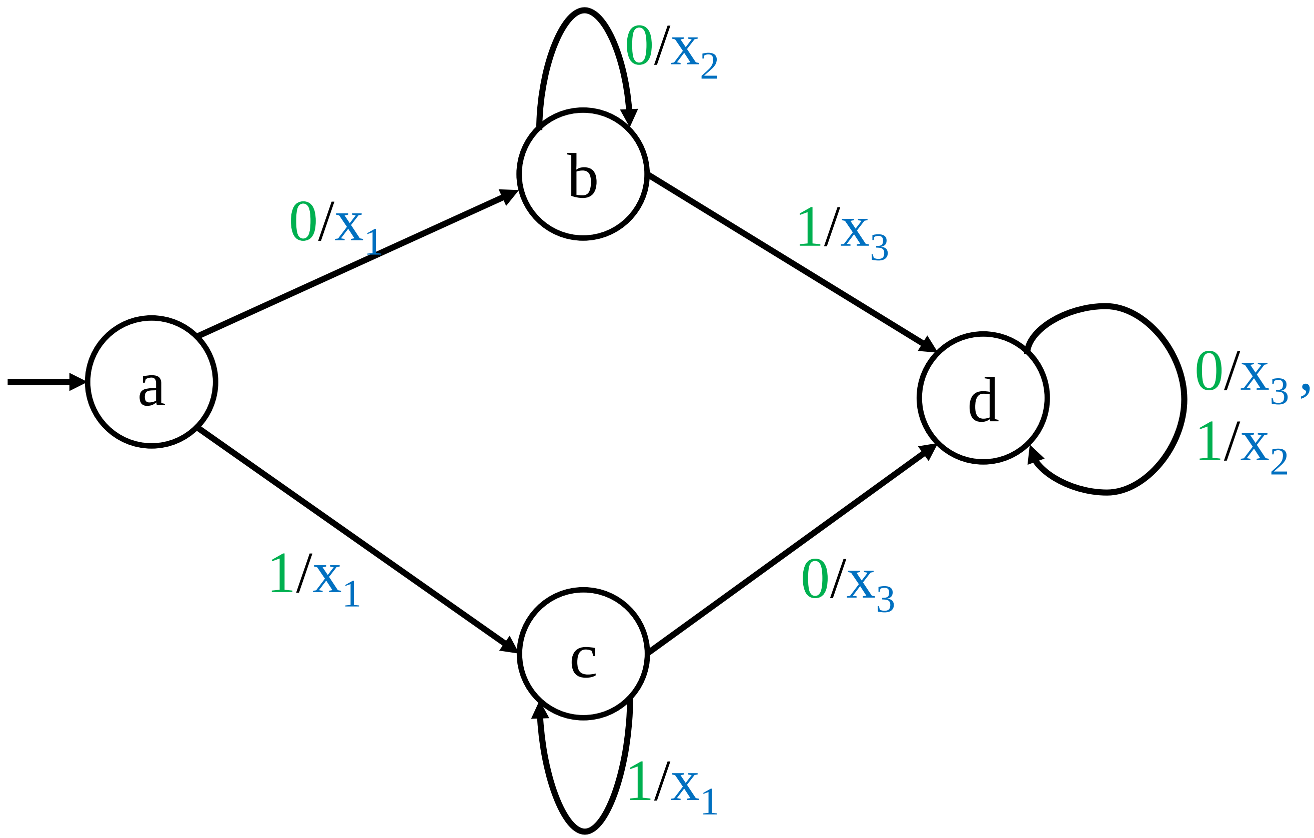
output depends only on the state

- Mealy Machine
  - Is an FSM whose output depends on the present state as well as the present input.
  - It can be described by a 6 tuple $(Q, \Sigma, \Gamma, \delta, \omega, q_0)$ where −
    - Q : is a finite set of **internal states**
    - Σ : is a finite set of symbols called the **input alphabet**
    - Γ : is the output alphabet
    - $\delta : Q \times \Sigma \rightarrow Q$ is a function called the **input transition function**
    - $\omega : Q \times \Sigma \rightarrow \Gamma$ is a **output transition function**
    - $q_0 \in Q$ is the **initial state**

- The state table of a Mealy Machine is shown below

Present State	Next State			
	input = 0		input = 1	
	State	Output	State	Output
→a	b	x_1	c	x_1
b	b	x_2	d	x_3
c	d	x_3	c	x_1
d	d	x_3	d	x_2

- The state diagram of the above Mealy Machine is –

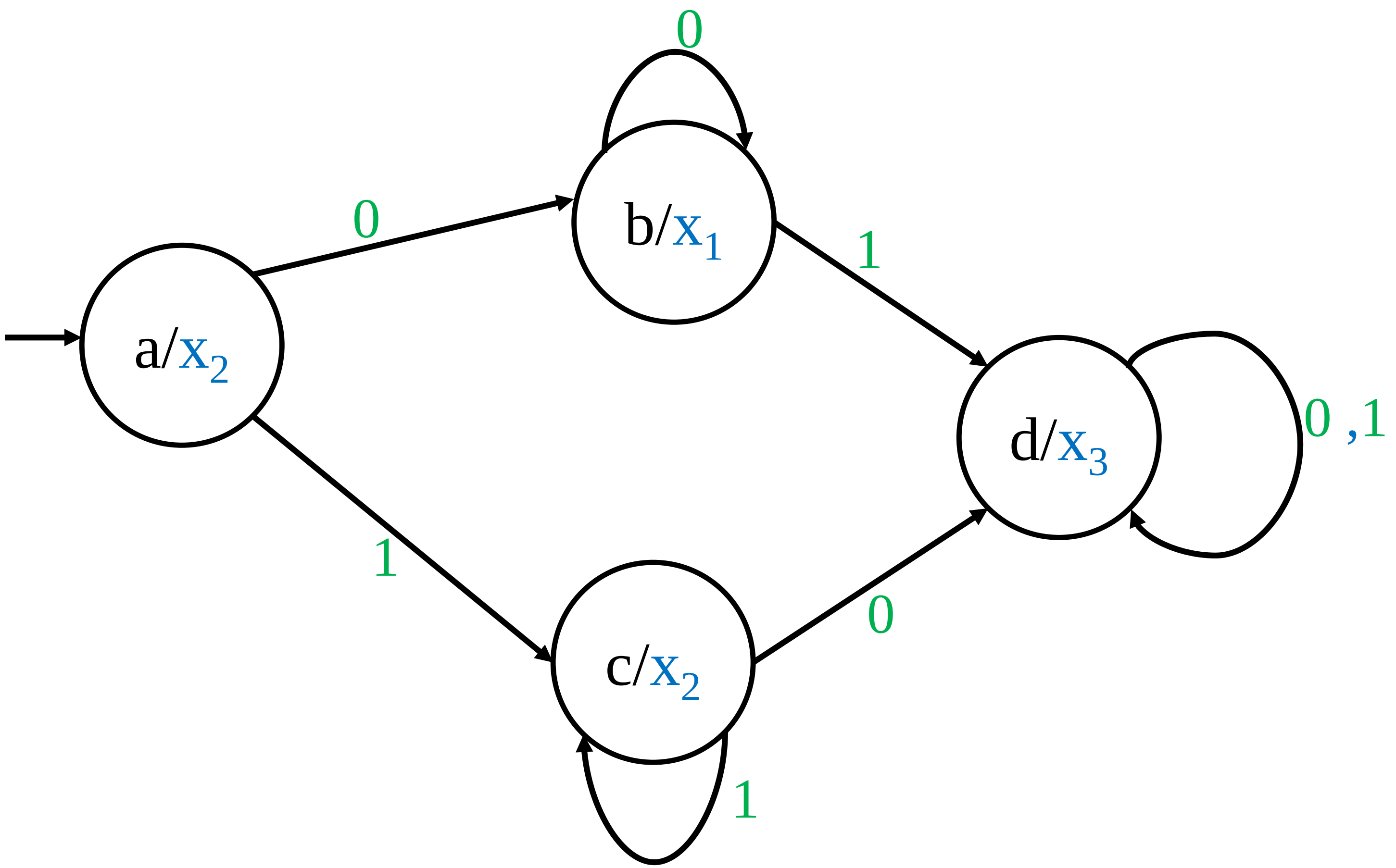


- Moore Machine
  - Is an FSM whose output depends only n the present state.
  - It can be described by a 6 tuple $(Q, \sum, \Gamma, \delta, \omega, q_0)$ where −
    - Q : is a finite set of **internal states**
    - $\sum$: is a finite set of symbols called the **input alphabet**
    - Γ : is the output alphabet
    - $\delta : Q \times \sum \rightarrow Q$ is a function called the **input transition function**
    - $\omega : Q \rightarrow \Gamma$ is a **output transition function**
    - $q_0 \in Q$ is the **initial state**

- The state table of a Moore Machine is shown below

Present State	Next State		Output
	input = 0	input = 1	
→a	b	c	x_2
b	b	d	x_1
c	c	d	x_2
d	d	d	x_3

- The state diagram of the above Moore Machine is –



- Conversion of Moore Machine to Mealy Machine

Input − Moore Machine

Output − Mealy Machine

Step 1 − Take a blank Mealy Machine transition table format.

Step 2 − Copy all the Moore Machine transition states into this table format.

Step 3 − Check the present states and their corresponding outputs in the Moore Machine state table; if for a state Q_i output is m, copy it into the output columns of the Mealy Machine state table wherever Q_i appears in the next state.

- Conversion of Mealy Machine to Moore Machine

Input − Mealy Machine

Output − Moore Machine

Step 1 − Calculate the number of different outputs for each state (Q_i) that are available in the state table of the Mealy machine.

Step 2 − If all the outputs of Qi are same, copy state Q_i. If it has n distinct outputs, break Q_i into n states as Q_{in} where **n** = 0, 1, 2.......

Step 3 − Create an empty Moore machine with new generated state. For Moore machine, Output will be associated to each state irrespective of inputs.

Non-deterministic Finite Automata(NFA)

Definition: A non-deterministic finite automata **(NFA)** is defined by the tuple M = $(Q, \sum, \delta, q_0, F)$ where $Q, \sum, \delta, q_0, F$ are defined as for dfa, but δ is defined as below

$$\delta : Q \times (\sum \cup \{\in\}) \rightarrow 2^Q$$

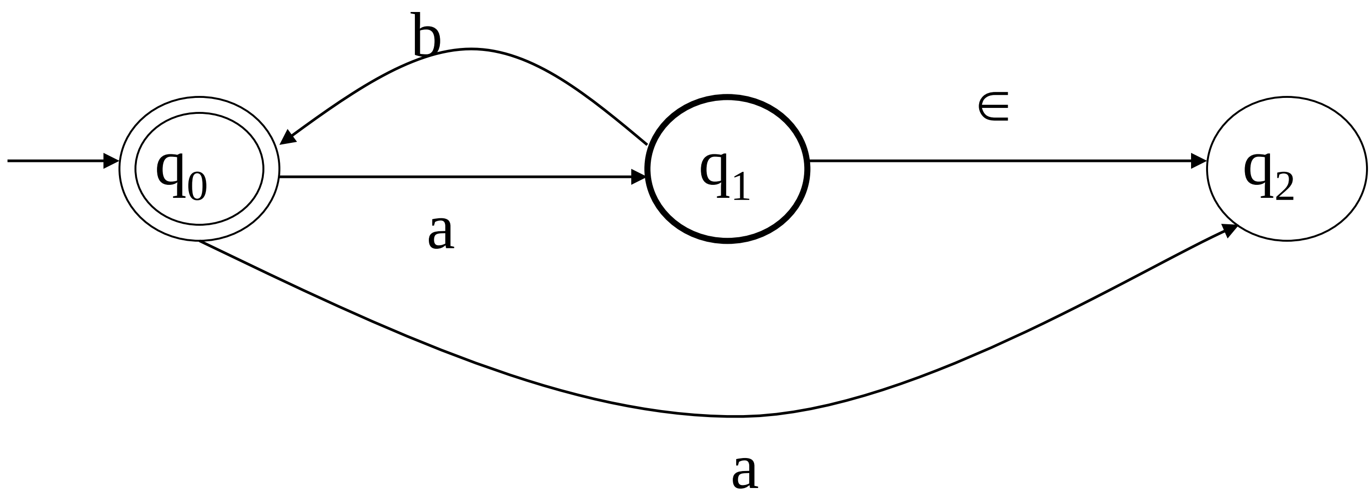
Where 2^Q is the power set of Q

δ Is a relation (not a function)

Differences between DFA and NFA

- The range of δ is a set.
  - There may be multiple transitions defined on the same state on the same input.
- $\in$ is allowed as the second argument of δ
  - -NFA can make a transition without consuming an input symbol.
  - Input mechanism is stationary on some moves
- $\{\} \in 2^Q$ – transitions may not have been defined for some inputs.

Example of a nfa



- Several edges with the same label originate from one vertex.

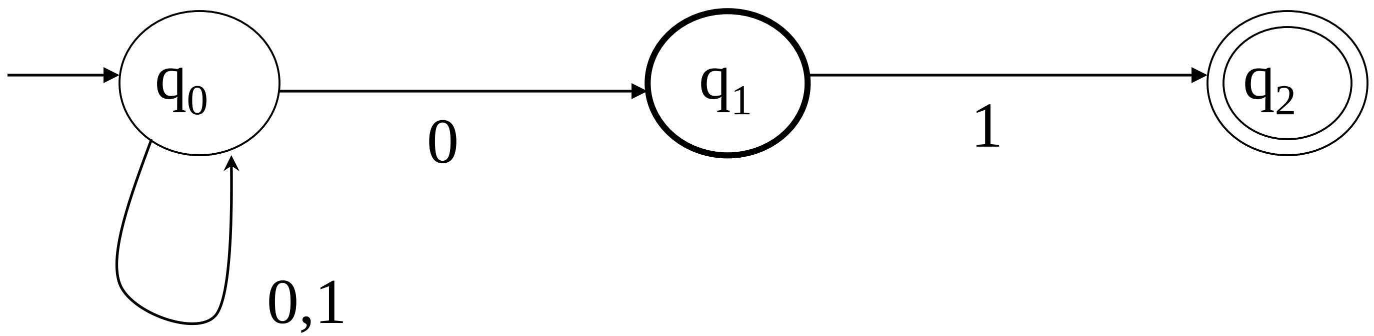
- Has $\in$-transitions

- Some transitions are unspecified eg. $\delta(q_2,a)$

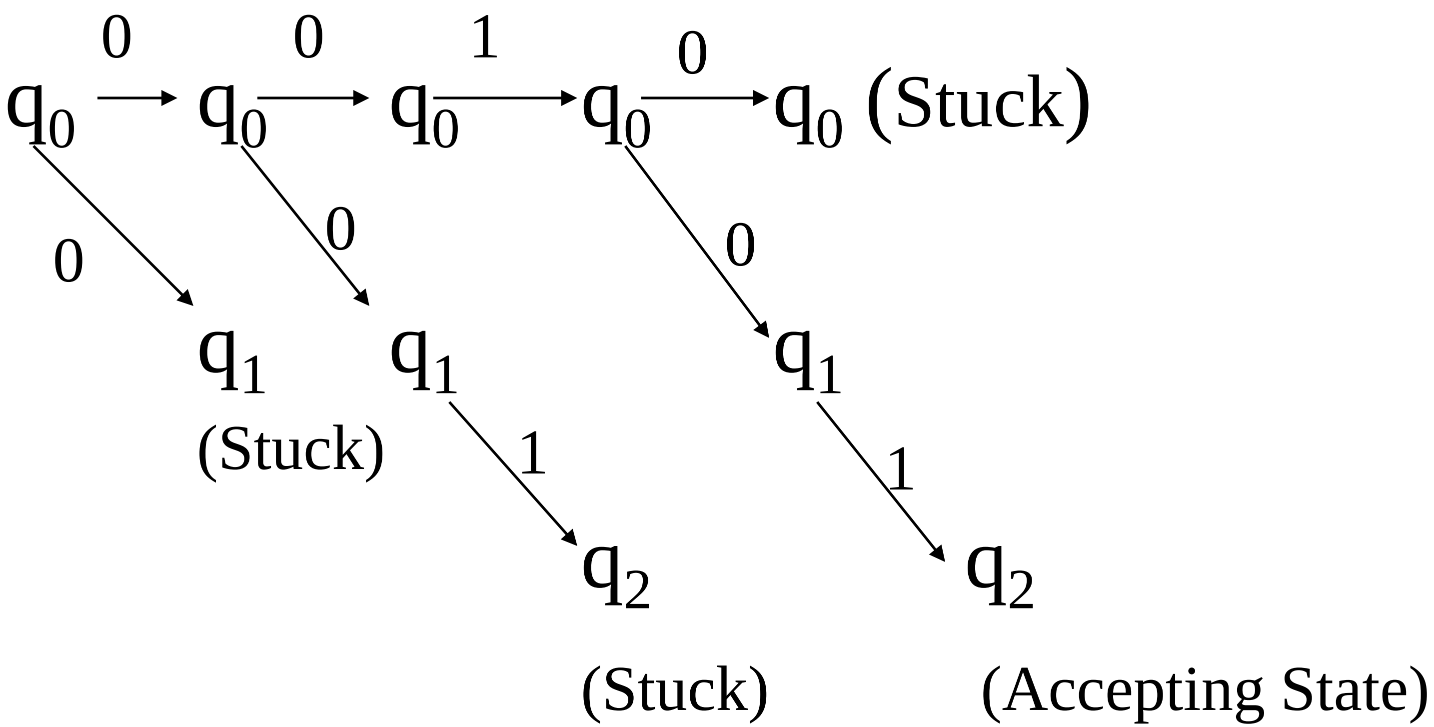
Transition table for the NFA

State	Input Symbols		
	a	b	$\in$
q_0	$\{q_1, q_2\}$	--	--
q_1	--	$\{q_0\}$	$\{q_2\}$
q_2	--	--	---

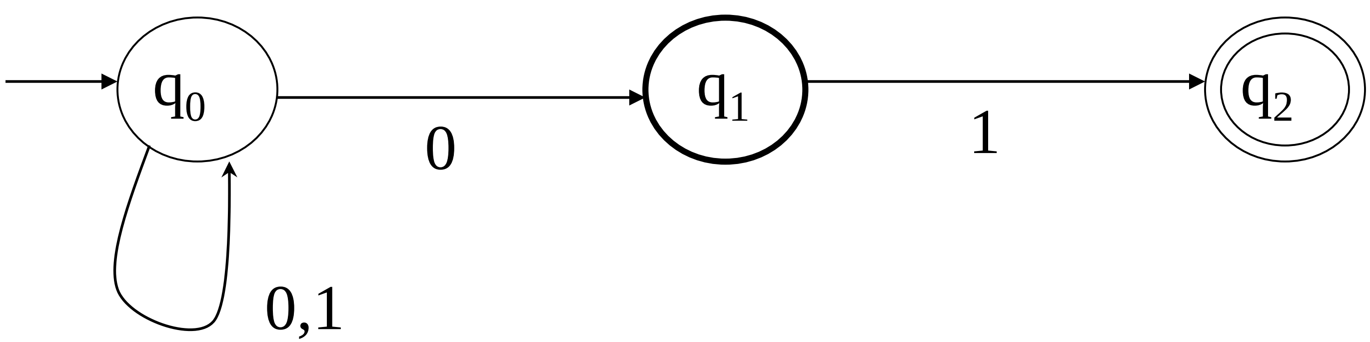
- Example



- Is the string 00101 is accepted by this NFA?



String to check 00101 ?



Strings Accepted by NFA

– A string is accepted by an NFA if there is **a sequence** of possible moves (path) that will take the machine to a final state at the end of the string.
    - For a given string there may be multiple possible walks from the start state.

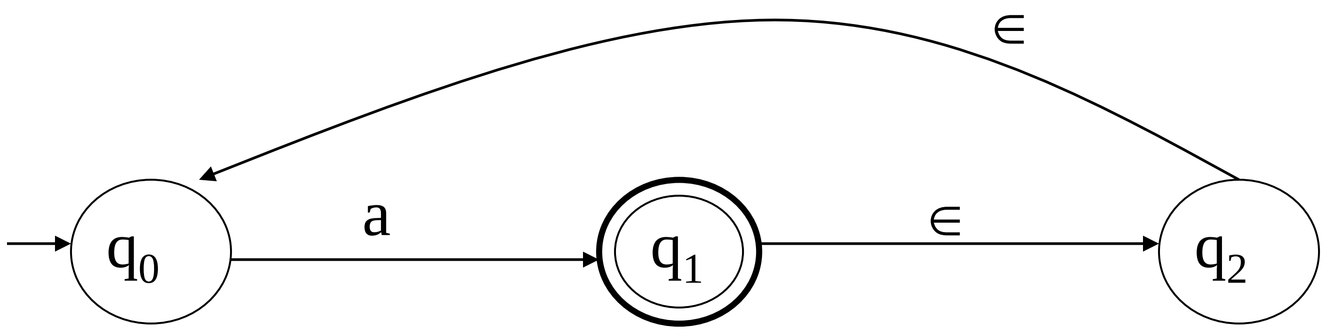
– A string is rejected only if there is no possible sequence of moves by which a final state can be reached.

- Extended Transition function for nfa
  - $\delta^*(q, \in) = \{q\}$
  - Let $w = xa$, where $a \in \sum$ and $\delta^*(q, x) = \{p_1,p_2,\ldots..,p_k\}$ and $\bigcup_{i=1}^{k} \delta(pi,a) = \{ r_1,r_2,\ldots..r_m\}$

    Then $\delta^*(q, w) = \{ r_1,r_2,\ldots..r_m\}$

- Definition : For an NFA, the extended transition function is defined such that $q_j \in \delta^* (q_i, w)$, iff there is a walk in the transition graph from q_i to q_j labeled w.

$\in$-closure



$\delta^*(q_2, \in) = \{q_0\}$

$\delta^*(q_0, a) = \{q_1, q_2, q_0\}$

The set of states that the NFA can reach from a state q (including itself) through $\in$ transitions, is called the $\in$-closure of q.

$\in$-closure$(q_1) = \{q_1, q_2, q_0\}$

Note that $\in$-closure of a state will always include that state itself.

Languages Accepted by NFA

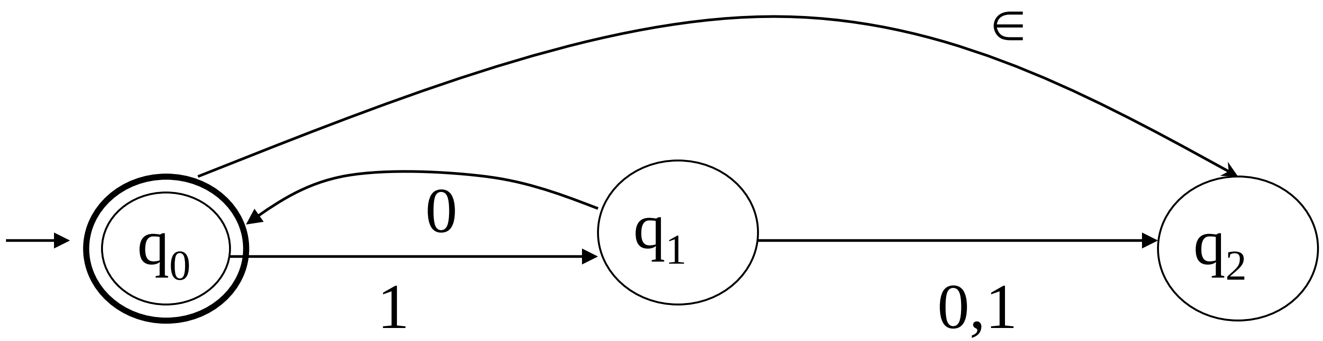
Definition : The language L accepted by an NFA: $M = (Q, \Sigma, \delta, q_0, F)$ is defined as below:

$$L(M) = \{w \mid w \in \Sigma^*, \delta^* (q_0, w) \cap F \neq \phi\}$$

– There is a walk labeled w from the initial vertex of the transition graph to some final vertex.

There may be several walks for a string w that are ended up in different final states

Dead Configurations



What will happen when the string to be read is 110?

- $\delta^*(q_0, 11) = q_2$
- $\delta^*(q_2, 0)$ is undefined

In such a situation the automaton is said to be in a dead configuration.

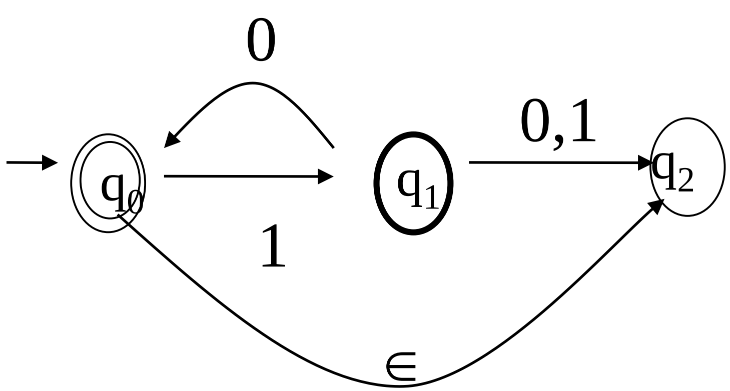
- Automaton stopped without any further action

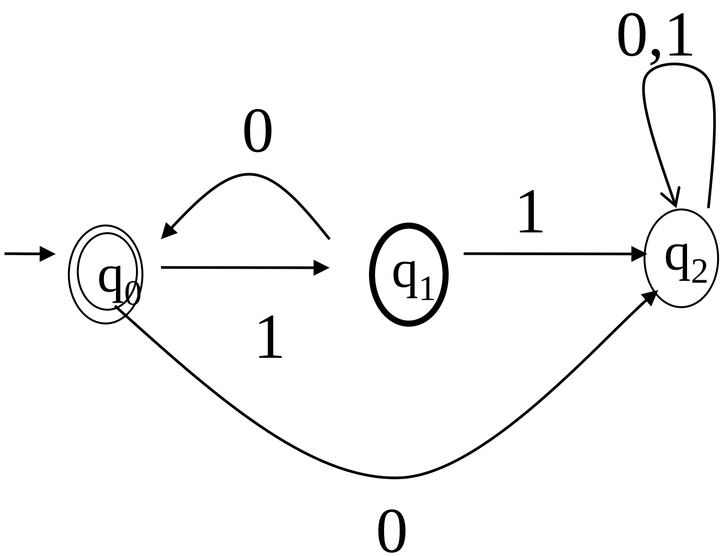
- For every NFA with an arbitrary number of final states there is an equivalent NFA with only one final state.
- Why Non-determinism?

Non-determinism is an effective mechanism for describing some complicated languages concisely.

- Definition: Two finite accepters M1 and M2 are said to be equivalent if they both accept the same language

$$L(M1) = L(M2)$$





Both finite accepters are equivalent and accept the language $L = \{(10)^n | n \geq 0\}$