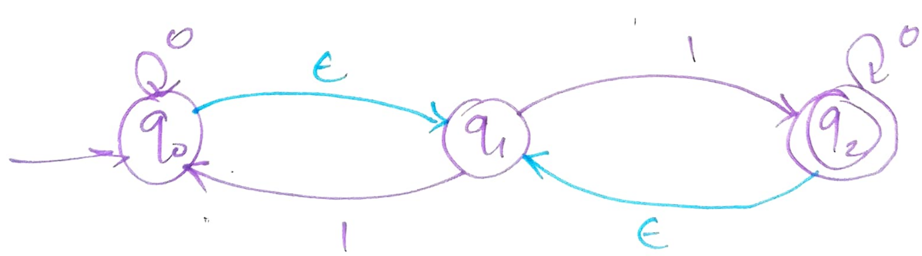
* ε-Closure of q :- (or) Extended transition funⁿ of ε-NFA

Let 'q' is any state in ε-NFA then the set of all the states which are at '0' ~~dista~~ (zero) distance from the state 'q' is called as ε-Closure(q)

Note: Every state is at zero (0) distance from itself

⇒ ε-Closure of ~~a~~ state q_i is the set of states including q_i where q_i can reach by any number of ε-moves of the given non-deterministic finite automata (NFA)

⇒ ε-Closure of state q_i includes q_i.

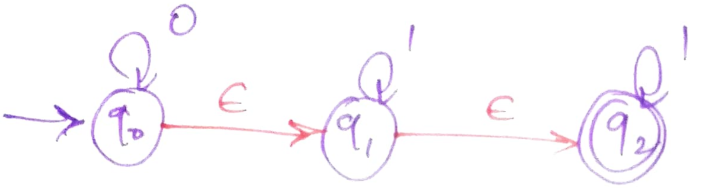


ε-Closure(q_0) = {q_0, q_1}

ε-Closure (q_1) = {q_1}

ε-Closure(q_2) = {q_2, q_1}

(2)

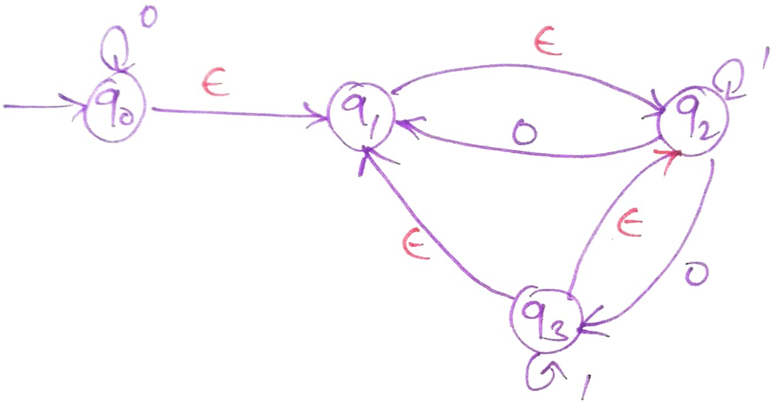


ϵ-Closure $(q_0) = \{q_0, q_1, q_2\}$

ϵ-Closure $(q_1) = \{q_1, q_2\}$

ϵ-Closure $(q_2) = \{q_2\}$

(3)



<u>ϵ-Closure</u>

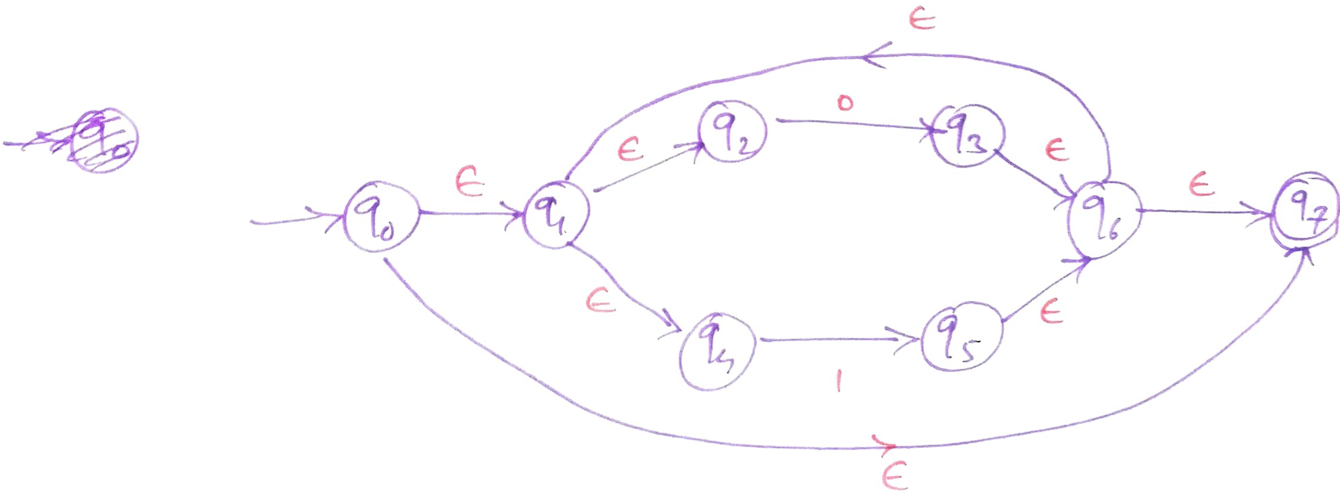
ϵ-Closure $\{q_0\} = \{q_0, q_1, q_2\}$

ϵ-Closure $(q_1) = \{q_1, q_2\}$

ϵ-Closure $(q_2) = \{q_2\}$

ϵ-Closure $(q_3) = \{q_3, q_1, q_2\}$

(4)



Note:-

1) ϵ-Closure of q is a nonempty finite subset of Q

2) ϵ-Closure$(\phi) = \phi$

3) ϵ-Closure$(q_0 \cup q_1 \cup \cdots \cup q_n) = \bigcup_{i=0}^{n} \epsilon$-Closure$(q_i)$

---

* Conversion of ϵ-NFA to NFA:-

1) No change in initial state.

2) No change in ~~total~~ total no. of states.

3) May be change in the final states.

Algorithm:

Let $M = \{Q, \Sigma, \delta, q_0, F\} \Rightarrow \epsilon$-NFA

$M' = \{Q', \Sigma, \delta', q_0', F'\} \Rightarrow$ NFA.

① Initial State:

No change in initial state

$q_0' = q_0$

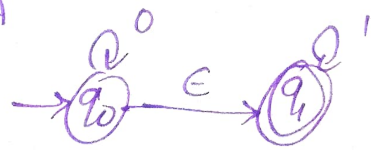
② Construction of δ':-

$\delta'(q, x) = \epsilon\text{-Closure}(\delta(\epsilon\text{-Closure}(q), x))$

③ Final state:

Every state whose ϵ-Closure contains the final state of ϵ-NFA is a final state in NFA

ex:



$\epsilon(q_0) = \{q_0, q_1\}$

$\epsilon(q_1) = \{q_1\}$

$$\delta'(q_0, 0) = \epsilon\text{-Closure}\{\delta(\epsilon\text{-Closure}(q_0), 0)\}$$

$$= \epsilon\text{-Closure}\{\delta((q_0, q_1), 0)\}$$

$$= \epsilon\text{-Closure}\{\delta(q_0, 0) \cup \delta(q_1, 0)\}$$

$$= \epsilon\text{-Closure}\{\{q_0, q_1\} \cup \phi\}$$

$$= \{q_0^*, q_1^*\}$$

$$\delta'(q_0, 1) = \epsilon\text{-Closure}\{\delta(\epsilon\text{-Closure}(q_0), 1)\}$$

$$= \epsilon\text{-Closure}\{\delta((q_0, q_1), 1)\}$$

$$= \epsilon\text{-Closure}\{\delta(q_0, 1) \cup \delta(q_1, 1)\}$$

$$= \epsilon\text{-Closure}\{\phi \cup q_1\}$$

$$= \epsilon\text{-Closure}(q_1)$$

$$= q_1$$

$$\delta'(q_1, 0) = \epsilon\text{-Closure}\{\delta(\epsilon\text{-Closure}(q_1), 0\}$$

$$= \epsilon\text{-Closure}\{\delta(q_1, 0)\}$$

$$= \epsilon\text{-Closure}\{\phi\}$$

$$= \phi$$

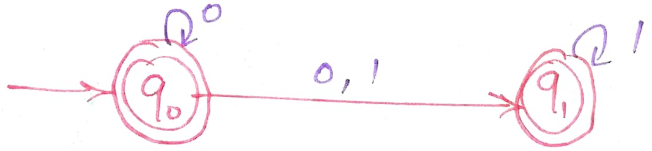
$$\delta'(q_1, 1) = \epsilon\text{-Closure}\{\delta(\epsilon\text{-Closure}(q_1), 1\}$$

$$= \epsilon\text{-Closure}\{\delta(q_1, 1)\}$$

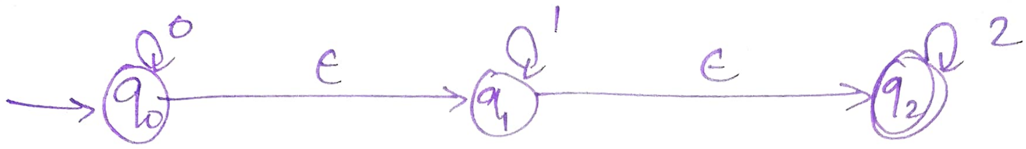
$$= \epsilon\text{-Closure}\{q_1\}$$

$$= q_1$$

NFA



(2) ϵ-NFA to NFA



$\rightarrow$ $\epsilon\text{-}(q_0) = \{q_0, q_1, q_2\}$

$\epsilon(q_1) = \{q_1, q_2\}$

$\epsilon(q_2) = \{q_2\}$

$\delta'(q_0, 0) = \epsilon\text{-Closure}\{\delta(\epsilon(q_0), 0)\}$

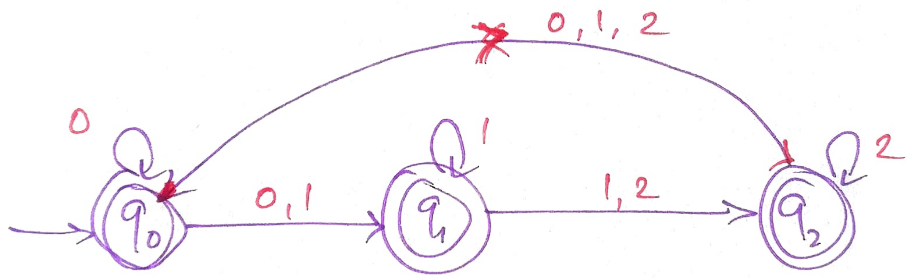
$= \epsilon\text{-Closure}\{\delta(q_0, q_1, q_2), 0)\}$

$= \epsilon\text{-}\{q_0 \cup \phi \cup \phi\}$

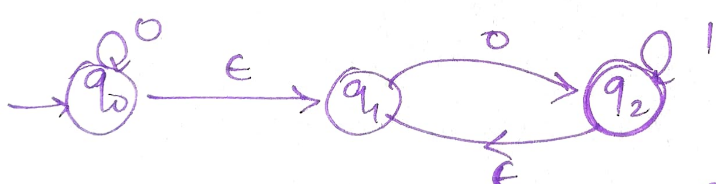
$= \{q_0^*, q_1^*, q_2^*\}$

NFA

δ	0	1	2
$\rightarrow q_0^*$	$\{q_0, q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
q_1^*	$\{\phi$	$\{q_1, q_2\}$	$\{q_2\}$
q_2^*	ϕ	ϕ	q_2

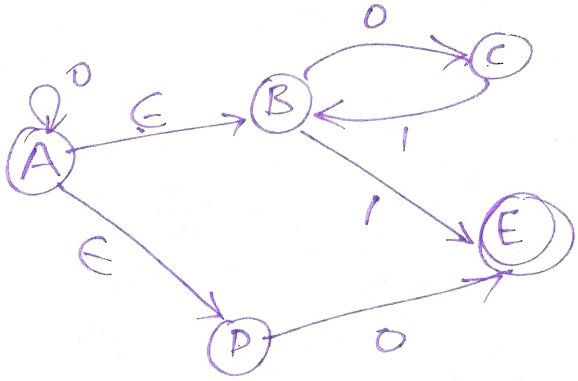


(3)



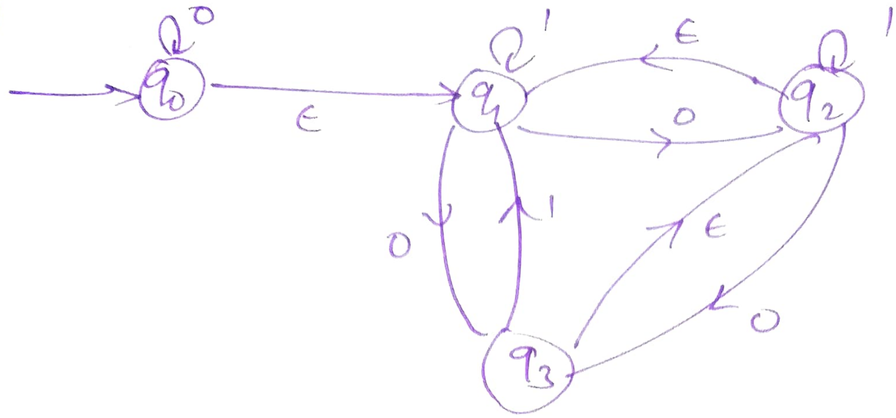
(B)

(4)



ϵ - NFA to DFA

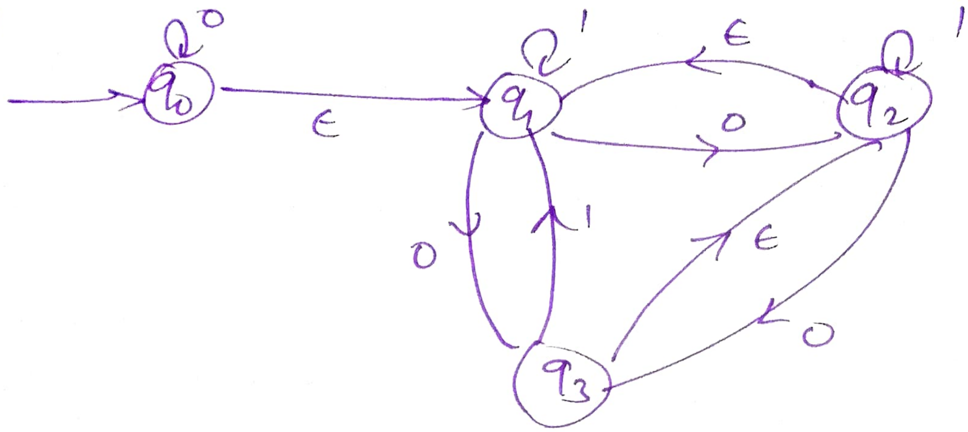
(5)



The process of conversion of ϵ-NFA to NFA is called Thomson construction.

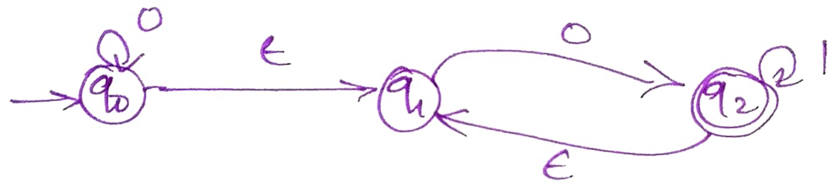
ϵ - NFA to DFA

(5)



The process of conversion of ϵ-NFA to NFA is called Thomson construction.

(3)



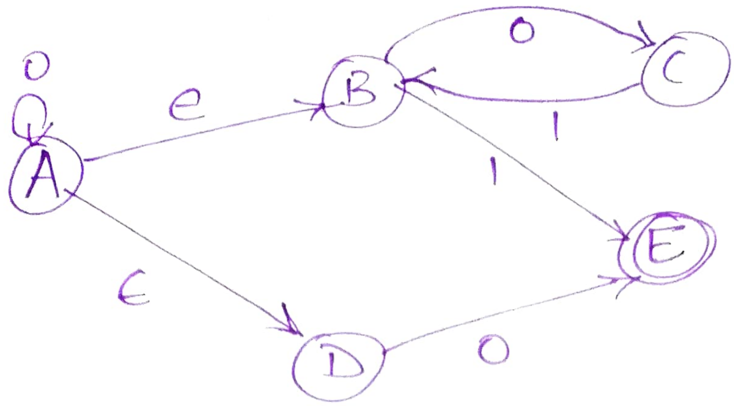
$\epsilon(q_0) = \{q_0, q_1\}$, $\epsilon(q_2) = \{q_2, q_1\}$

$\epsilon(q_1) = \{q_1\}$,

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

δ	0	1
$\rightarrow q_0$	$\{q_0, q_1, q_2\}$	ϕ
q_1	$\{q_2, q_1\}$	ϕ
q_2^*	$\{q_2, q_1\}$	$\{q_2, q_1\}$

4



→ $\epsilon(A) = \{A, B, D\}$

$\epsilon(B) = \{B\}$

$\epsilon(C) = \{C\}$

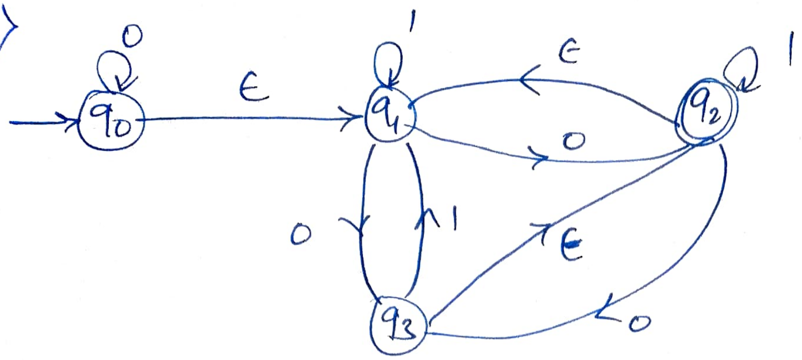
$\epsilon(D) = \{D\}$

$\epsilon(E) = \{E\}$

$$
\begin{aligned}
\delta'(A,0) &= \epsilon\{\delta(\epsilon(A),0)\} \\
&= \epsilon\{\delta(A,B,D),0)\} \\
&= \epsilon\{\delta(A,0) \cup \delta(B,0) \cup \delta(D,0)\} \\
&= \epsilon\{A \cup C \cup E\} \\
&= \{A, B, D, C, E\}
\end{aligned}
$$

δ	0	1
→A	$\{A, B, C, D, E\}$	ϕ
B	C	E
C	ϕ	B
D	E	ϕ
E^*	ϕ	ϕ

5)



$\epsilon(q_0) = \{q_0, q_1\}$

$\epsilon(q_1) = \{q_1\}$

$\epsilon(q_2) = \{q_2, q_1\}$

$\epsilon(q_3) = \{q_3, q_2\}$

→

δ	0	1
$\rightarrow q_0$	$\{q_0, q_1, q_2, q_3\}$	$\{q_1\}$
q_1	$\{q_1, q_2, q_3\}$	$\{q_1\}$
q_2^*	$\{q_1, q_2, q_3\}$	$\{q_1, q_2\}$
q_3	$\{q_2, q_3\}$	$\{q_1, q_2\}$

* Conversion of ϵ-NFA to DFA :-

* May be change in initial State
* May be Change in total no. of States
* May be Change in final states.

→ It is also based on subset construction.

→ ϵ-Closure is calculated for every state in the subset.

Algorithm :-

Let $M = \{Q, \Sigma, \delta, q_0, F\} \Rightarrow \epsilon$-NFA

$M' = \{Q', \Sigma, \delta', q_0', F'\} \Rightarrow$ DFA

① Initial State :-

$$q_0' = \epsilon\text{-Closure}(q_0)$$

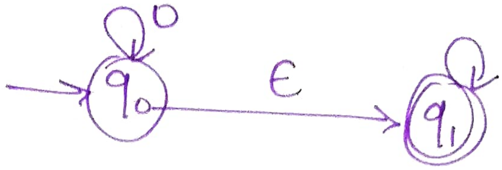
② Construction of δ' :-

$$\delta'(q, x) = \epsilon\text{-Closure}\{\delta(q, x)\}$$

Start the construction of δ' with the initial state & continue for every new state and stop the construction whenever no new state appears.

1

Ex:-



→ $\epsilon(q_0) = \{q_0, q_1\} \Longrightarrow$ ① Initial State

$\epsilon(q_1) = \{q_1\}$

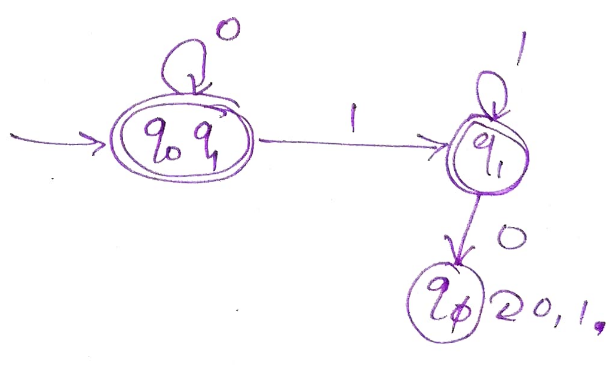
- $\delta'(\{q_0, q_1\}, 0) = \epsilon\text{-closure}\{\delta(q_0, q_1), 0\}$
$= \epsilon\text{-Closure}\{\delta(q_0, 0) \cup \delta(q_1, 0)\}$
$= \epsilon\text{-Closure}\{q_0 \cup \phi\}$
$= \{q_0, q_1\}$

$\delta'(\{q_0, q_1\}, 1) = \epsilon\text{-Closure}\{\delta(q_0, q_1), 1\}$
$= \epsilon\text{-Closure}\{\delta(q_0, 1) \cup \delta(q_1, 1)\}$
$= \epsilon\text{-Closure}\{\phi \cup q_1\}$
$= \{q_1\}$ //

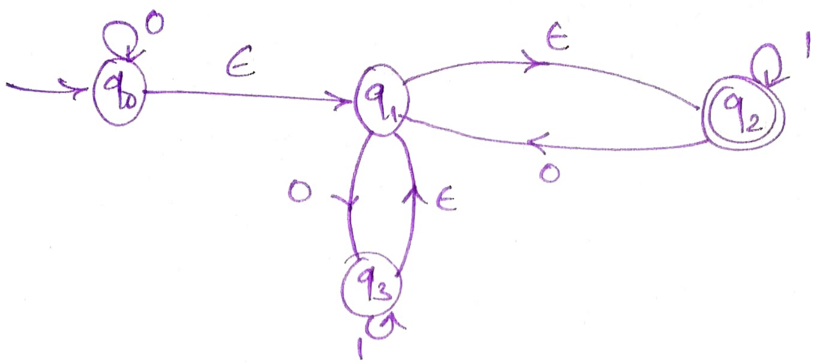
- $\delta'(\{q_1\}, 0) = \epsilon\text{-Closure}\{\delta(q_1, 0)\}$
$= \epsilon\text{-Closure}(\phi)$
$= \phi$

$\delta'(q_1, 1) = \epsilon\text{-Closure}(\delta(q_1, 1))$
$= \epsilon\text{-Closure}(q_1) = q_1.$

δ'	0	1
$\rightarrow \{q_0, q_1\}^*$	$\{q_0, q_1\}$	q_1
q_1^*	ϕ.	q_1
ϕ	ϕ	ϕ



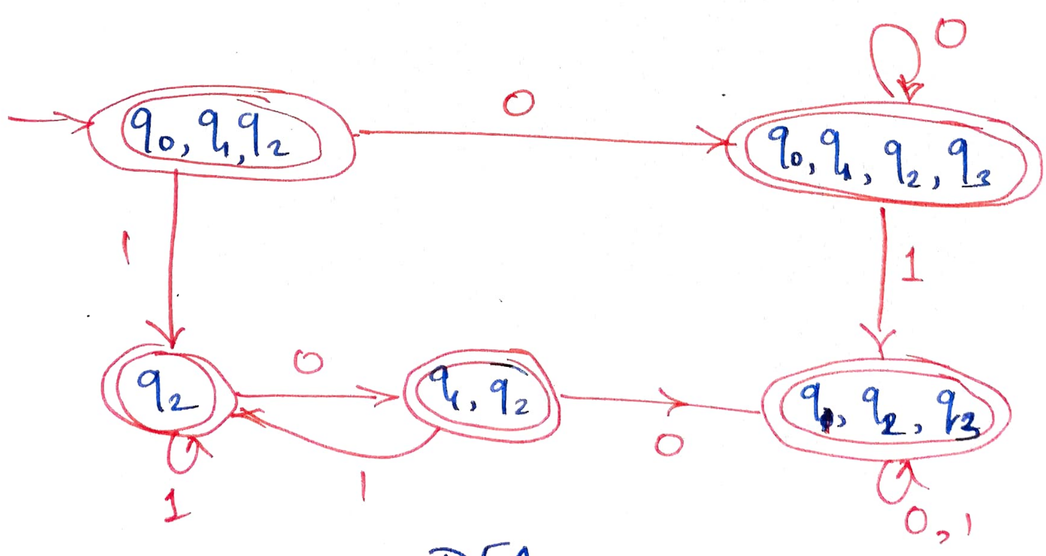
②



→ $\epsilon(q_0) = \{q_0, q_1, q_2\}$; $\epsilon(q_2) = \{q_2\}$

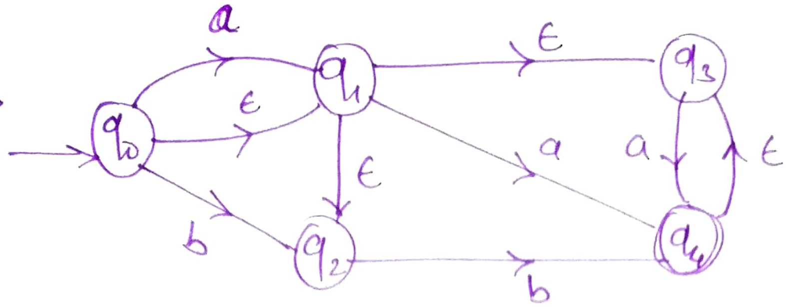
$\epsilon(q_1) = \{q_1, q_2\}$; $\epsilon(q_3) = \{q_3, q_1, q_2\}$

δ'	0	1
$\rightarrow \{q_0, q_1, q_2\}^*$	$\{q_0, q_1, q_2, q_3\}$ //	q_2 //
$\{q_0, q_1, q_2, q_3\}^*$	$\{q_0, q_1, q_2, q_3\}$	$\{q_1, q_2, q_3\}$ //
$\{q_1, q_2, q_3\}^*$	$\{q_1, q_2, q_3\}$	$\{q_1, q_2, q_3\}$
$\{q_2\}^*$	$\{q_1, q_2\}$ //	$\{q_2\}$
$\{q_1, q_2\}^*$	$\{q_1, q_2, q_3\}$	$\{q_2\}$



DFA

3>



ε-NFA to DFA

→

* FA as Output Devices :-

Moore Machine ($\lambda : Q \to \Delta$)

Mealy Machine ($\lambda : Q \times \Sigma \to \Delta$)

→ Both Moore & Mealy machines are special case of DFA.

→ Moore & Mealy machines are output producers rather than ~~acc~~ language acceptors, so no need to define final states i.e. $F = \phi$

→ Moore & Mealy machines are used to implement small count that doesn't requires extra memory.

→ No concept of dead states but there may be equal states.

→ Machine generates an output on every input and output alphabet is denoted by capital delta (Δ)

→ It uses output function (λ)

* Two Behaviours of Machine:

① State Transition Function (STF) :- δ

② Output Function (λ)

$\delta : \Sigma \times Q \to Q$ {For Mealy & Moore Machine}

$\lambda : \Sigma \times Q \to \Delta$ {Mealy Machine}

$\lambda : Q \to \Delta$ {Moore Machine}

1) **Mealy Machine**:- Output is associated with transition. is called as Mealy Machine.

- Whenever this machine enters any state on a particular input it generates output.

It is represented with 6-tuple

$$\boxed{M = \{Q, \Sigma, \Delta, \delta, \lambda, q_0\}}$$

Where, Q - Set of all finite states

Σ - set of all finite i/p symbols

Δ - set of all finite o/p symbols

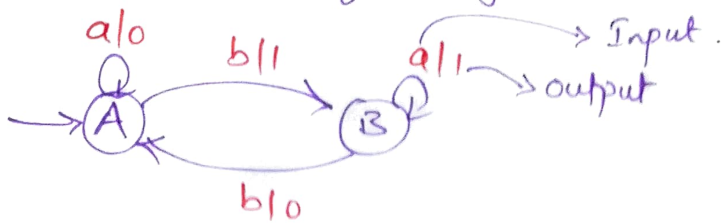
δ - $Q \times \Sigma \rightarrow Q$ [δ is mapping function $Q \times \Sigma$ to Q]

λ - is mapping function which maps $Q * \Sigma$ to Δ ($Q \times \Sigma \rightarrow \Delta$)

q_0 - Initial State.

* Representation of Mealy Machine :-

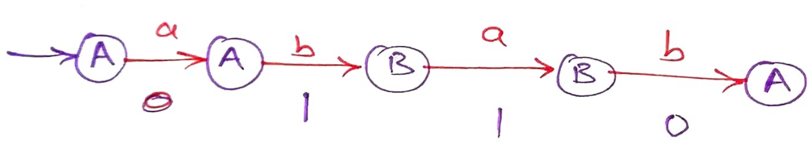
Ex :-



δ	a	λ	b	λ
$\to$ A	A	0	B	1
B	B	1	A	0

δ	a	b
$\to$ A	A/0	B/1
B	B/1	A/0

W = abab



$\therefore \ \lambda(abab) = 0110$

Note :-

1) output depends on state + I/p symbol (transition)

2) Length of the input = length of the output

3) Can not respond for empty string ϵ

$(\lambda(\epsilon) = \epsilon)$