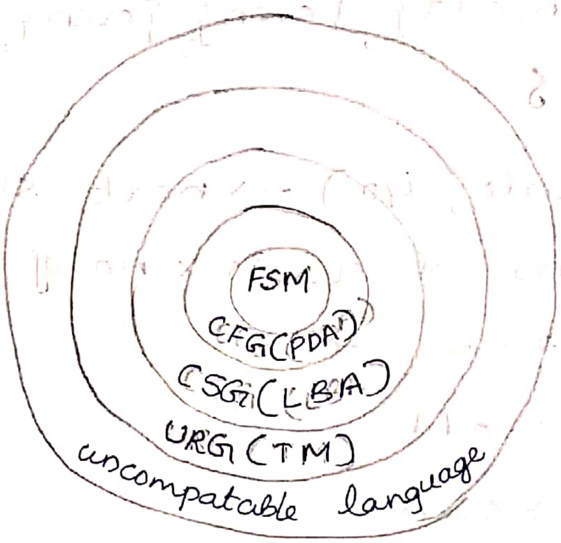
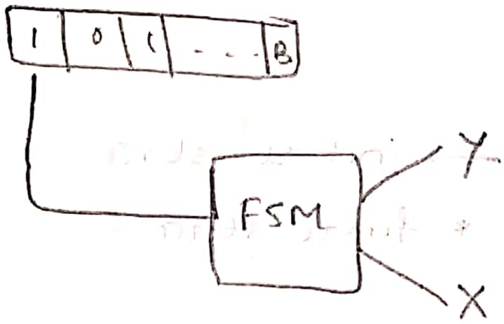
MODULE - 1

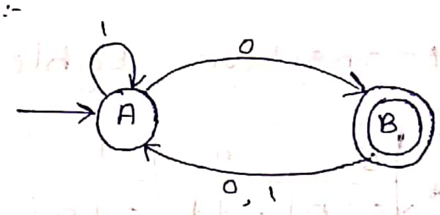
Formal language of machines



Finite state machine (FSM)

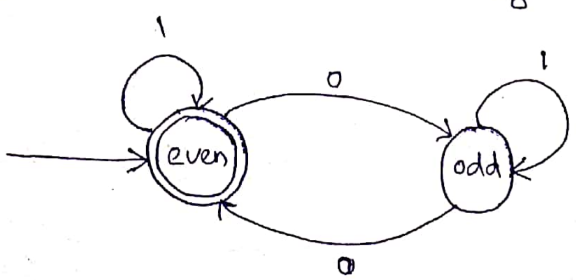


eg:-



Q1. Design a FSM to recognise even number of 0's, over {0, 1}.

transition state diagram:-



FSM or FA = $\{Q, q_0, F, \delta, \Sigma\}$

Q = set of all states ; q_0 = initial state (only 1) [set of Q]

F = final state [can be a set]

δ = transition function ; Σ = inputs (language)

$\therefore$ FSM = $\{\{even, odd\}, \{even\}, \{even\}, \delta, \{0,1\}\}$

Transition fn, δ

δ(current state, i/p) $\rightarrow$ next state.

no. of δs. = no. of states x no. of i/ps

= 2 x 2 = 4

δ(even, 0) $\rightarrow$ odd

δ(even, 1) $\rightarrow$ even

δ(odd, 0) $\rightarrow$ even

δ(odd, 1) $\rightarrow$ odd

transition table

Q \ Σ	0	1
$\rightarrow$ * even	odd	even
odd	even	odd

$\rightarrow$ initial state

* final state

Q. | 1 | 0 | 0 | 1 | 1 | B |

δ(A, 10011)

δ(A, 0011)

δ(B, 011)

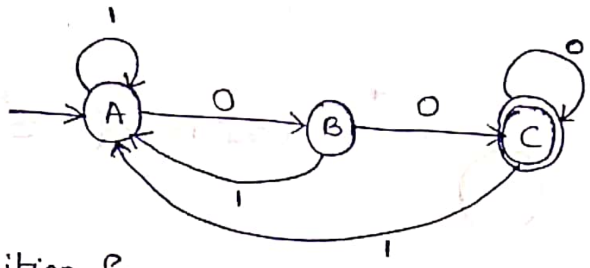
δ(A, 11)

δ(A, 1)

δ(A, #) # - blank

Scanned by CamScanner

Q2. Design a FA that recognizes string that ends with '00' over {0,1}



transition fn, eg:-

$\delta(A, 0010100)$

— $\delta(B, 010100)$

$\delta(C, 10100)$

$\delta(A, 0100)$

$\delta(B, 100)$

$\delta(A, 00)$

$\delta(B, 0)$

$\delta(C, \#)$

$\delta(A, 0010)$

$\rightarrow \delta(B, 010)$

$\delta(C, 10)$

$\delta(A, 0)$

$\delta(B, \#)$

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

$= \{\{A, B, C\}, \{A\}, \{C\}, \delta, \{0,1\}\}$

transition fn :-

$\delta(A, 0) \rightarrow B$

$\delta(A, 1) \rightarrow A$

$\delta(B, 0) \rightarrow C$

$\delta(B, 1) \rightarrow A$

$\delta(C, 0) \rightarrow C$

$\delta(C, 1) \rightarrow A$

$\delta(C, \#) \rightarrow C$

Q3. Design a FA that recognizes string that has atleast two zero's over $\{0,1\}$.

$\Sigma = \{0, 1\}$

$\Sigma^* = ?$ $\Sigma^* =$ Union of Σ^0 to Σ^α

$\Sigma^0 = \phi$

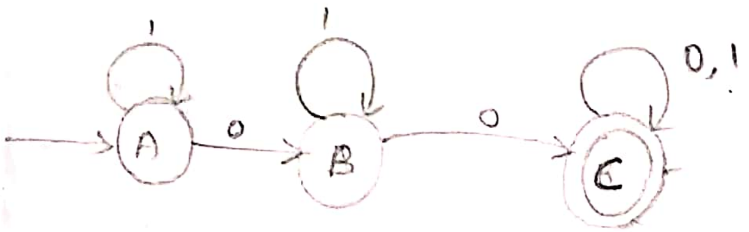
$\Sigma^1 = \phi$

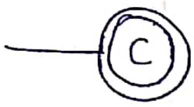
$\Sigma^2 = \{00\}$

$\Sigma^3 = \{001, 010, 100, 000\}$

$\Sigma^4 = \{0000, 0001, 0010, 0011, 0100, 0101, 0110, 1000, 1001, 1010, 1100\}$

without ϕ :-





Finite automata / finite State machine (FA)

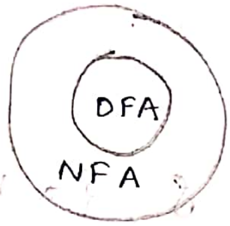
DFA ⊂ NFA

deterministic

- FA without o/p
  - DFA
  - NFA
  - NFA - ϵ
- FA with o/p
  - Moore
  - Mealy

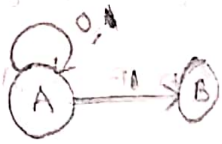
$M = \{Q, q_0, F, \delta, \Sigma\}$

DFA $= \delta : Q \times \Sigma \longrightarrow Q$



NFA $= \delta : Q \times \Sigma \longrightarrow P(Q)$ power set

can be there may be more than 1 value.



$(B, 0) \rightarrow \epsilon$

DFA

Transition fn :-

1. $\delta(q, x) \rightarrow q'$

   $q, q' \in Q$

   $x \in \Sigma$

2. $\delta(q, \epsilon) \longrightarrow q$

3. $\delta(q, xa)$

   $* \ \delta(\delta(q, x), a)$

2 & 3 extended δ

NFA - ϵ

$Q \times (\Sigma \cup \epsilon) \rightarrow P(Q)$

language acceptability of DFA :-

$L(DFA) = \{ w \mid w \in \Sigma^* \ \& \ \delta^*(q_0, w) \in F \}$

Q1. Design a DFA to accept even numbers of a's and b's over $\{a, b\}$.

$\Sigma^0 = \phi$

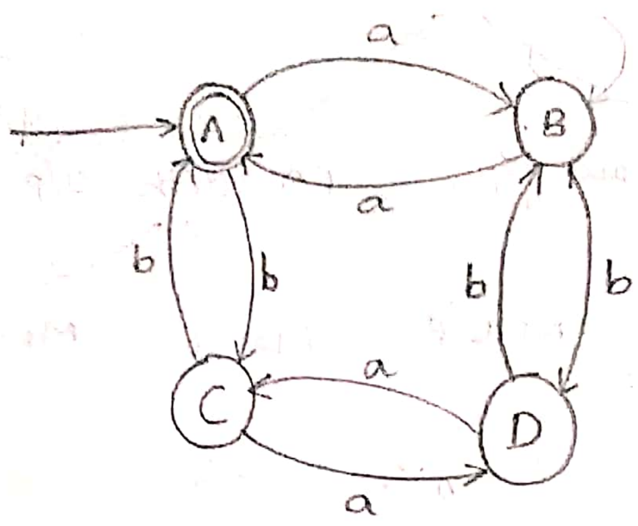
$\Sigma^1 = \phi$

$\Sigma^2 = \{aa, bb\}$

$\Sigma^3 = \{$ ~~aaa~~, ~~aab~~ ϕ

$\Sigma^4 = \{$ aaaa, ~~aabb~~, aabb, abab, abba, bbbb, bbaa, baba, baab $\}$

a bab



~~FA = { {A, B}, {A}~~,

FA = { {A, B, C, D}, {A}, {A}, δ, {a, b} }

$\delta(A, a) \rightarrow B$ $\qquad$ $\delta(C, a) \longrightarrow D$

$\delta(A, b) \longrightarrow C$ $\qquad$ $\delta(C, b) \longrightarrow A$

$\delta(B, a) \longrightarrow A$ $\qquad$ $\delta(D, a) \longrightarrow C$

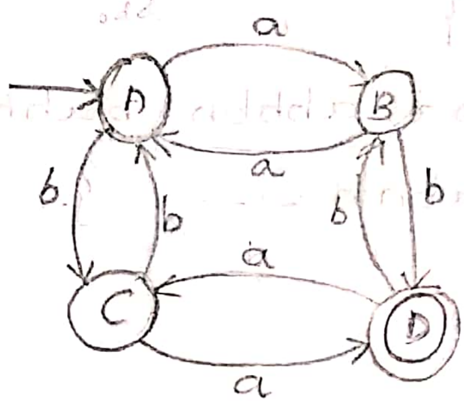
$\delta(B, b) \longrightarrow D$ $\qquad$ $\delta(D, b) \longrightarrow B$

transition table :-

Q \ Σ	a	b
A	B	C
B	A	D
C	D	A
D	C	B

Q2. Design a DFA to accept odd no. of a's & b's over {a,b}

$\Sigma^* = \{$ ab, ba, aaab, aaba, abaa, baaa, bbba, bbab, babb, abbb - - - $\}$



$FA = \{ \{A, B, C, D\}, \{A\}, \{D\}, \delta, \{a, b\} \}$

$\delta(A, a) \rightarrow B$

$\delta(A, b) \rightarrow C$

$\delta(B, a) \rightarrow A$

$\delta(B, b) \rightarrow D$

$\delta(C, a) \rightarrow D$

$\delta(C, b) \rightarrow A$

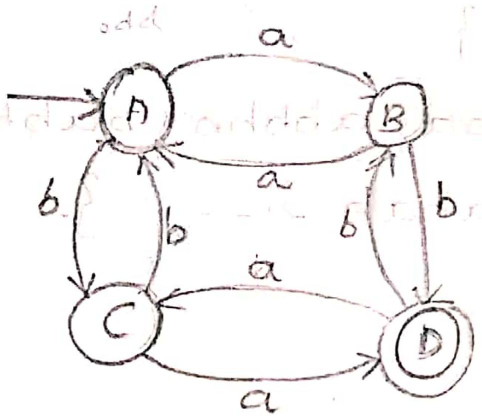
$\delta(D, a) \rightarrow C$

$\delta(D, b) \rightarrow B$

$\delta(D, \#) \rightarrow D$

transition table :-

Q \ Σ	a	b
A	B	C
B	A	D
C	D	A
D	C	B



$FA = \{ \{A, B, C, D\}, \{A\}, \{D\}, \delta, \{a, b\} \}$

$\delta(A, a) \rightarrow B$

$\delta(A, b) \rightarrow C$

$\delta(B, a) \rightarrow A$

$\delta(B, b) \rightarrow D$

$\delta(C, a) \rightarrow D$

$\delta(C, b) \rightarrow A$

$\delta(D, a) \rightarrow C$

$\delta(D, b) \rightarrow B$

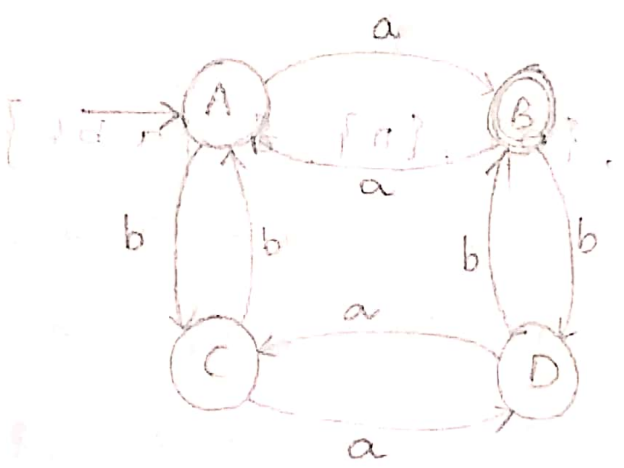
$\delta(D, \#) \rightarrow D$

transition table :-

Q \ Σ	a	b
A	B	C
B	A	D
C	D	A
D	C	B

Q3. Design a DFA to accept odd no. of a's & even no. of b's over {a, b}

$\Sigma = \{a, abb, bab, bba, abbbb, babbb \ldots, aaa, aaabb, aababb \ldots\}$



$FA = \{\{A, B, C, D\}, \{A\}, \{B\}, \delta, \{a, b\}\}$

$\delta(A, a) \to B$ $\delta(C, a) \to D$

$\delta(A, b) \to C$ $\delta(C, b) \to A$

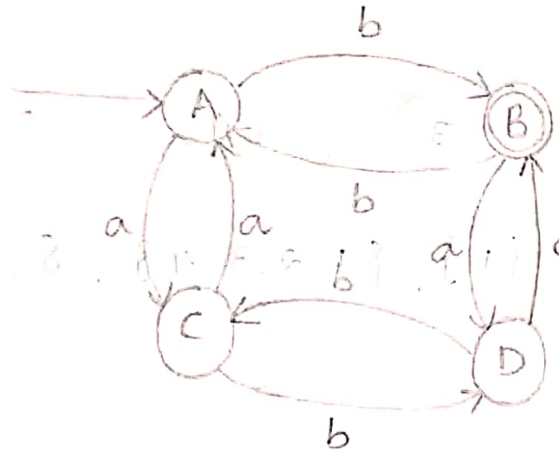
$\delta(B, a) \to A$ $\delta(D, a) \to C$

$\delta(B, b) \to D$ $\delta(D, b) \to B$

Q \ Σ	a	b
A	B	C
B	A	D
C	D	A
D	C	B

Q4. even a's & odd b's

$\Sigma^* = \{ b, aab, aba, baa, \cancel{bbb}, baaaa \ldots$
$bbb, bbbaa, bbaab \ldots, bbbbb \ldots$



$FA = \{ \{A, B, C, D\}, \{A\}, \{B\}, \delta, \{a, b\} \}$

$\delta(A, a) \rightarrow C$

$\delta(A, b) \rightarrow B$

$\delta(B, a) \rightarrow D$

$\delta(B, b) \rightarrow A$

$\delta(C, a) \rightarrow A$

$\delta(C, b) \rightarrow D$

$\delta(D, a) \rightarrow B$

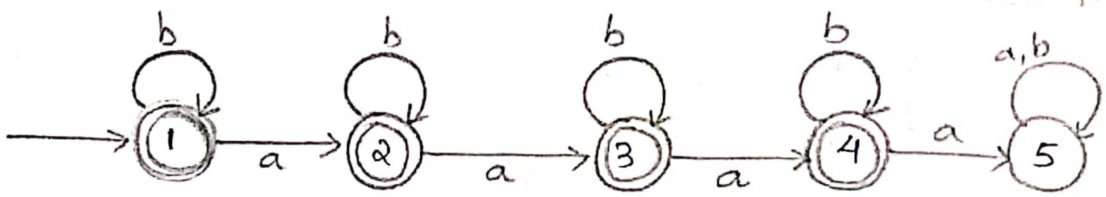
$\delta(D, b) \rightarrow C$

Q \ Σ	a	b
A	C	B
B	D	A
C	A	D
D	B	C

Q5. atmost 3 a's over {a, b}

$\Sigma^* = \{ \phi, a, aa, aaa, b, bb, bbb, \ldots,$

$ab, ba, aab, baa, \ldots, bbabbaa \ldots \}$



$FA = \{\{1, 2, 3, 4, 5\}, \{1\}, \{1, 2, 3, 4\}, \delta, \{a, b\}\}$

$\delta(1, a) \rightarrow 2$

$\delta(1, b) \rightarrow 1$

$\delta(2, a) \rightarrow 3$

$\delta(2, b) \rightarrow 2$

$\delta(3, a) \rightarrow 4$

$\delta(3, b) \rightarrow 3$

$\delta(4, a) \rightarrow 5$

$\delta(4, b) \rightarrow 4$

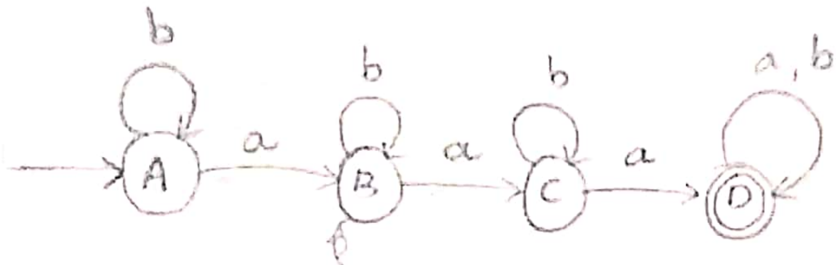
$\delta(5, a) \rightarrow 5$

$\delta(5, b) \rightarrow 5$

Q \ Σ	a	b
1	2	1
2	3	2
3	4	3
4	5	4
5	5	5

Q6. atleast 3 a's over {a, b}

$\Sigma^* = \{aaa, aaab, aaba, aaabb \ldots, aaaa, aaaba, aaaab \ldots, aaaaa \ldots\}$



FA = $\{\{A, B, C, D\}, \{A\}, \{D\}, \delta', \{a, b\}\}$

$\delta(A, a) \rightarrow B$

$\delta(A, b) \rightarrow A$

$\delta(B, a) \rightarrow C$

$\delta(B, b) \rightarrow B$

$\delta(C, a) \rightarrow D$

$\delta(C, b) \rightarrow C$

$\delta(D, a) \rightarrow D$

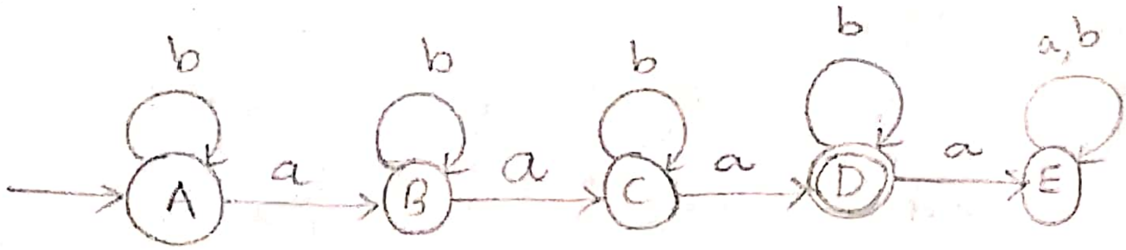
$\delta(D, b) \rightarrow D$

Q \ Σ	a	b
A	B	A
B	C	B
C	D	C
D	D	D

Scanned by CamScanner

Q7. Exactly 3 a's over $\{a, b\}$

$\Sigma = \{aaa, aaab, aaba, aaabb \ldots\}$



$FSM = \{\{A, B, C, D, E\}, \{A\}, \{D\}, \delta, \{a, b\}\}$

$\delta(A, a) \rightarrow B$

$\delta(A, b) \rightarrow A$

$\delta(B, a) \rightarrow C$

$\delta(B, b) \rightarrow B$

$\delta(C, a) \rightarrow D$

$\delta(C, b) \rightarrow C$

$\delta(D, a) \rightarrow E$

$\delta(D, b) \rightarrow D$

$\delta(E, a) \rightarrow E$

$\delta(E, b) \rightarrow E$

Q \ Σ	a	b
A	B	A
B	C	B
C	D	C
D	E	D
E	E	E

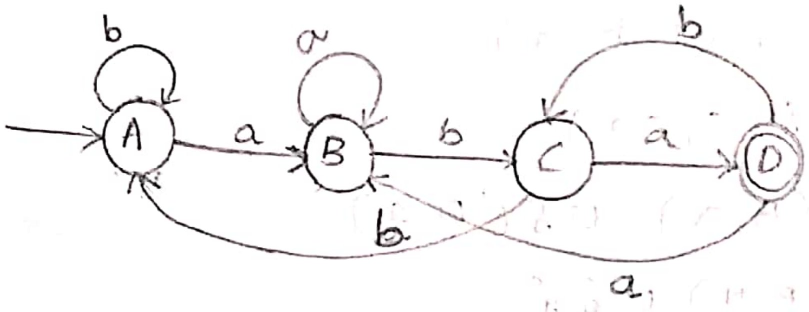
Design of Non-deterministic finite Automata

Q1. Design a NFA to recognize string ending with 'aba' over $\{a, b\}$

aaba

DFA

$\Sigma^* = \{\phi\ aba,\ aaba,\ baba,\ aaaba, \ldots\}$



$\delta(q_0, ababa)$

abaaba

$\vdash \delta(B, baba)$

$\vdash \delta(C, aba)$

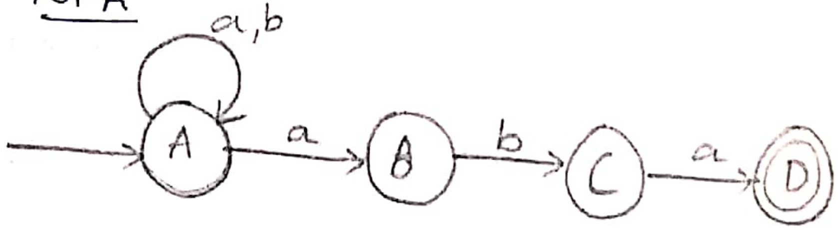
$\vdash \delta(D, ba)$

$\vdash \delta(C, a)$

$\vdash \delta(D, \#)$ $\therefore$ It is acceptable.

$FA = \{\{A, B, C, D\}, \{A\}, \{D\}, \delta, \{a, b\}\}$

NFA



$\delta : Q \times \Sigma \rightarrow P(Q)$

$\delta(A, ababa)$

$\vdash \delta(A, baba) \cup \delta(B, baba)$

$\vdash \delta(A, aba) \cup \delta(C, aba)$

$\vdash \delta(A, ba) \cup \delta(B, ba) \cup \delta(D, ba)$

$\vdash \delta(A,a) \cup \delta(C,a) \cup \cancel{\delta}\ \#$

$\vdash \delta(A,\#) \cup \delta(B,\#) \cup \delta(D,\#)$ $\quad$ $\delta(A,\#), \delta(B,\#)$ does not exist

$\vdash \underline{D}$

$\therefore$ acceptable.

$\longrightarrow \delta(A, abaa)$

$\vdash \delta(A, baa) \cup \delta(B, baa)$

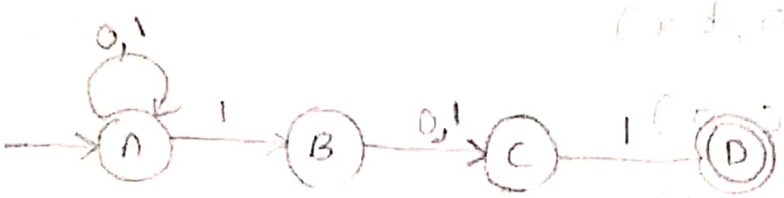
$\vdash \delta(A, aa) \cup \delta(C, aa)$

$\vdash \delta(A,a) \cup \delta(B,a) \cup \delta(D,a)$

$\vdash \delta(A,\#) \cup \delta(B,\#) \cup \cancel{\delta\delta}\ \#$

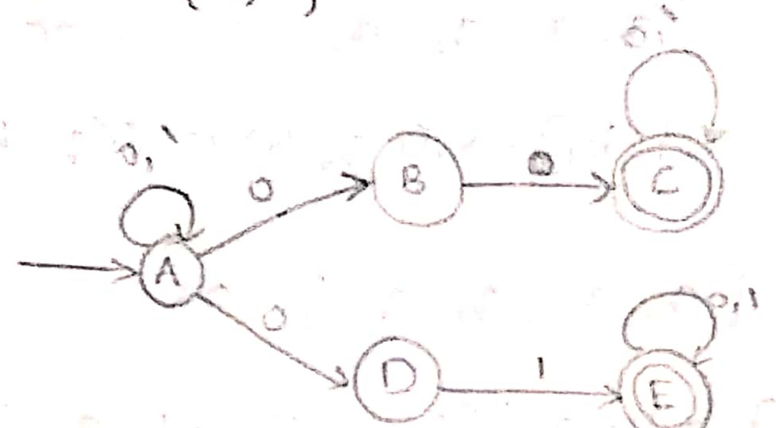
$\vdash \phi$ $\quad\quad$ $\therefore$ It is not acceptable.

Q2. Design NFA to recognize string ending with 101 or 111 over {0,1}



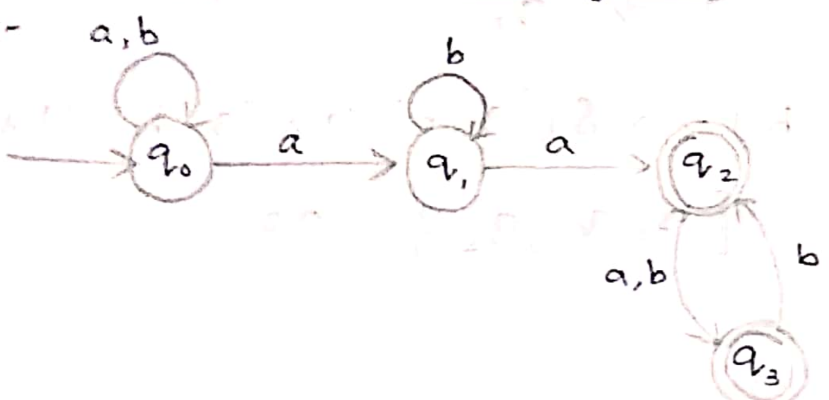
Q3.

Q3. Design NFA to recognize any string having 00 or 01 over {0,1}



Conversion / Equivalence of NFA & DFA

NFA:-



$\Sigma = \{a, b\}$

1. $\delta(q_0, a) \rightarrow \{q_0, q_1\}$ NS (new state)

$\delta(q_0, b) \rightarrow \{q_0\}$ OS (old state)

2. $\delta(\{q_0, q_1\}, a) \rightarrow \delta(q_0, a) \cup \delta(q_1, a)$

$= \{q_0, q_1, q_2\}$ NS

$\delta(\{q_0, q_1\}, b) \rightarrow \delta(q_0, b) \cup \delta(q_1, b)$

$= \{q_0, q_1\}$ OS

3. $\delta(\{q_0, q_1, q_2\}, a) \rightarrow \delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_2, a)$

$= \{q_0, q_1, q_2, q_3\}$ NS

$\delta(\{q_0, q_1, q_2\}, b) \rightarrow \delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_2, b)$

$= \{q_0, q_1, q_3\}$ NS

4. $\delta(\{q_0, q_1, q_2, q_3\}, a) \rightarrow \delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_2, a)$
$\cup \delta(q_3, a)$

$= \{q_0, q_1, q_2, q_3, \phi\}$ OS

$\delta(\{q_0, q_1, q_2, q_3\}, b) \rightarrow \delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_2, b)$
$\cup \delta(q_3, b)$

$= \{q_0, q_1, q_3, q_2\}$ OS

5. $\delta(\{q_0, q_1, q_3\}, a) \rightarrow \delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_3, a)$

$= \{q_0, q_1, q_2, \phi\}$ OS

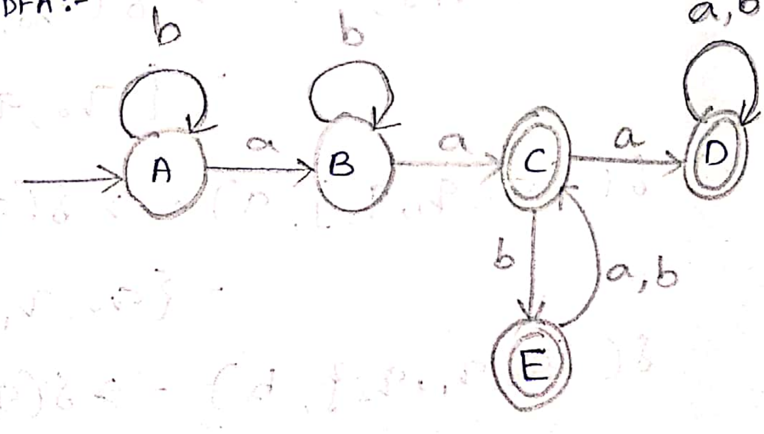
$\delta(\{q_0, q_1, q_3\}, b) \rightarrow \delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_3, b)$

$= \{q_0, q_1, q_2\}$ OS

		a	b
1.	$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
2.	$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$
3. *	$\{q_0, q_1, \underline{q_2}\}$ FS	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_3\}$
4. *	$\{q_0, q_1, \underline{q_2, q_3}\}$ FS	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$
5. *	$\{q_0, q_1, \underline{q_3}\}$ FS final-state	$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$

	δ	a	b
→	A	B	A
	B	C	B
*	C	D	E
*	D	D	D
*	E	C	C

DFA :-



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

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

q_0, Σ are same All others (Q, F, δ) will change.

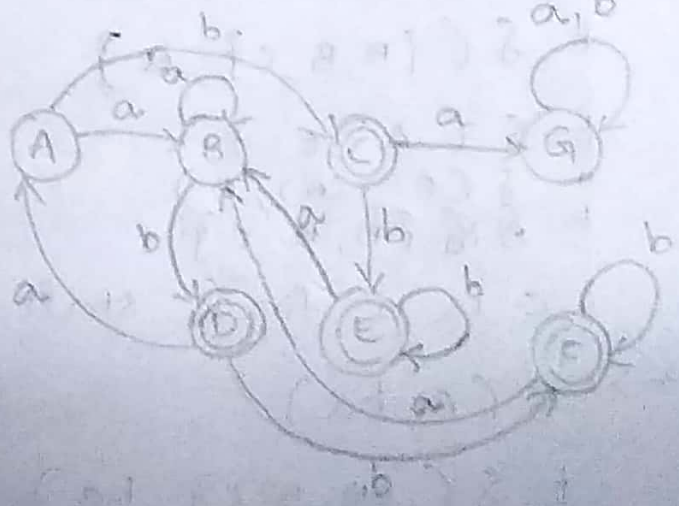
Q2. NFA $\longrightarrow$ DFA

NFA :-

	a	b
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_2\}$
q_1	$\{q_0\}$	$\{q_1\}$
$* q_2$	ϕ	$\{q_0, q_2\}$

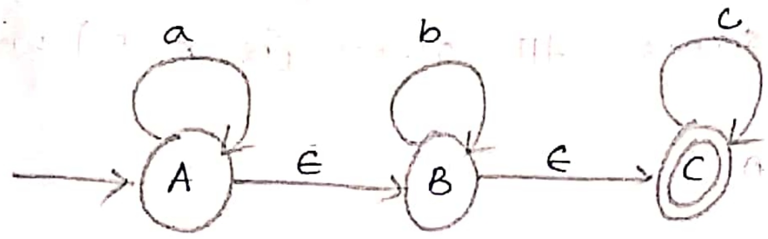
δ		a	b
$\rightarrow$	A $\{q_0\}$	$\{q_0, q_1\}$	$\{q_2\}$
	B $\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_2, q_1\}$
*	C $\{q_2\}$	$\{\phi, q_0\}$	$\{q_0, q_2\}$
	D $\{q_1, q_2\}$	$\{q_0\}$	$\{q_1, q_0, q_2\}$
	E $\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_2, q_0\}$
	F $\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$
	G	G	G

δ	a	b
$\rightarrow$ A	B	C
B	B	D
* C	G	E
* D	A	F
* E	B	E
* F	B	F
G	G	G



NFA - ϵ

$Q \times (\Sigma \cup \epsilon) \longrightarrow P(Q)$



It will accept any no of a followed by any no of b followed by any no of c.

eg:- a, aa, b, abc ...

ϵ- closure $(A) = \{A, B, C\}$

$\epsilon - c\ (B) = \{B, C\}$

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

$\rightarrow$ $\delta(A, aa)$

$\vdash \delta(\epsilon\text{-}c(A), aa)$

$\vdash \delta(\{A, B, C\}, aa\}$

$\vdash \delta((\delta(A, a) \cup \delta(B, a) \cup \delta(C, a)), a\}$

$\vdash \delta(\{A\}, a\}$

$\vdash \delta(\epsilon\text{-}c(A), a)$

$\vdash \delta(\{A, B, C\}, a\}$

$\vdash \delta(A, \epsilon)$

$\vdash \delta(\epsilon\text{-}c(A), \epsilon)$

$\vdash \{A, B, C\}$

C is a final state. $\therefore$ It is accepted.

$\rightarrow$ $\delta(A, ba)$

$\vdash \delta(\{A, B, C\}, ba)$

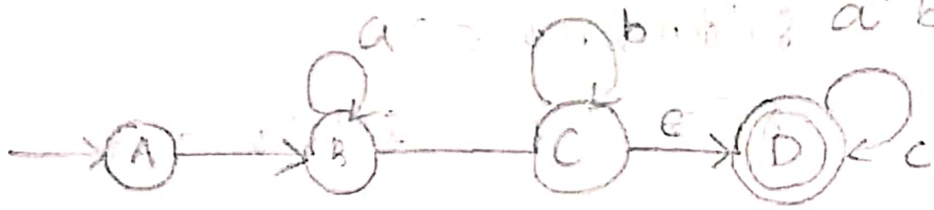
$\vdash \delta(B, a)$

$\vdash \delta(\{B, C\}, a)$

$\vdash \delta \in$ $\longrightarrow$ $\therefore$ not accepted.

Q2. NFA-ϵ to accept a string ~~contain~~ that starts with atleast 1 a.

$a^n b^m c^p$, $n > 0$, $m \geq 0$, $p \geq 0$

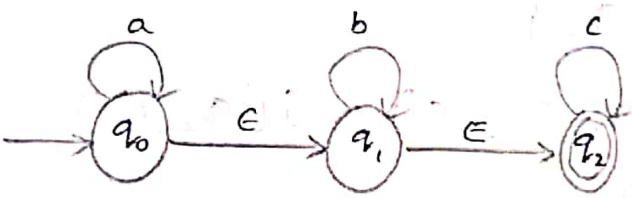


or

NFA-ϵ $\longrightarrow$ NFA

$$NFA = \{Q, q_0, F, \delta, \Sigma\}$$

$$NFA\text{-}\epsilon = \{Q, q_0, \underbrace{F', \delta', \Sigma'}_{\text{change}}\}$$



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

$$\{q_0, q_1, q_2\} \cap \overset{\text{final state}}{F_{(q_2)}} = q_2$$

$$\therefore NFA(F) = \{\underset{\text{initial state}}{q_0}, q_2\}$$

$q_0 \rightarrow$ if the [illegible] state will be there if [illegible]

$$\delta(q_0, a) = \delta(\epsilon\text{-}c(q_0), a)$$

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

$$= \delta(q_0 \cap \phi \cap \phi\}, \epsilon) = \delta(q_0, \epsilon)$$

Scanned by CamScanner

$\vdash \delta(\epsilon\text{-}c(q_0), \epsilon)$

$\vdash \delta(\{q_0, q_1, q_2\}, \cancel{\epsilon})$

$\vdash \cancel{\delta}\{q_0, q_1, q_2\}$

$\delta(q_0, b) = \delta(\{q_0, q_1, q_2\}, b)$

$= \delta(\phi \cap q_1 \cap \phi, \epsilon)$

$= \delta(q_1, \epsilon) = \cancel{\delta\phi}\{q_1, q_2\}$

- $\cancel{\delta(q_1, a) = \delta(\{q_1, q_2\}, a)}$

$= \cancel{\delta(\phi \cap}$

$\delta(q_0, c) = \delta(\{q_0, q_1, q_2\}, \cancel{b}c\}$

$= \delta(\phi \cap \phi \cap q_2, \epsilon) = \delta(q_2, \epsilon)$

$= \{q_2\}$

- $\delta(q_1, a) = \delta(\{q_1, q_2\}, a) = \delta(\phi, \epsilon) = \phi$

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

$= \{q_1, q_2\}$

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

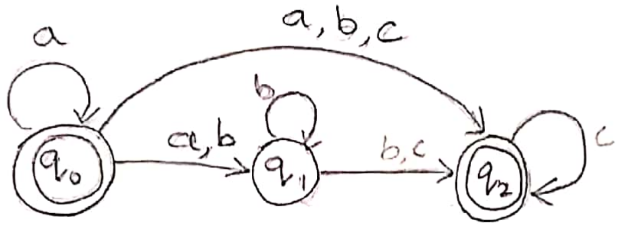
$= \{q_2\}$

- $\delta(q_2, a) = \delta(\{q_2\}, a) = \delta(\phi, \epsilon) = \phi$

$\delta(q_2, b) = \delta(\{q_2\}, b] = \delta(\phi, \epsilon) = \phi$

$\delta(q_2, \cancel{b}c) = \delta(\{q_2\}, c\} = \delta(q_2, \epsilon) = \{q_2\}$

Q \ Σ	a	b	c
$\rightarrow {}^{*}q_0$	$\{q_0, q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
q_1	ϕ	$\{q_1, q_2\}$	$\{q_2\}$
${}^{*}q_2$	ϕ	ϕ	$\{q_2\}$



2. Design NFA-ε to recognise any substring of abac