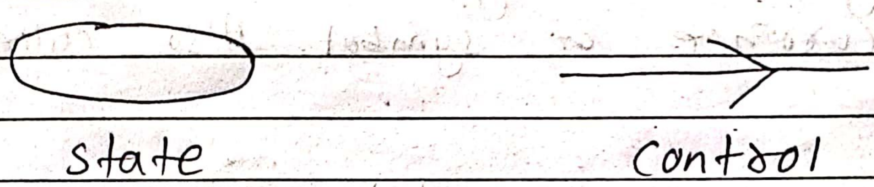
Unit 1 (Basic) + Unit 2

Date ____ Page ____

Abstract machine is a mathematical model that has state and control; state moves one state to another state in response to external inputs and control may be deterministic or non-deterministic.



Basic Terms :-

1. Symbol

→ Basic building block of TOC. It can be represented by pictorial form. eg:-
$\{a,b\}$ $\{0, ----9\}$ $\{$क --- ज्ञ$\}$

2. Alphabet (Σ)

→ Finite set of non-empty symbols. eg:-
$\Sigma = \{a,b\}$ or $\Sigma = \{0,1\}$

3. String (ω)

→ Finite sequence of symbol over given alphabet
Here, $\Sigma = \{a,b\}$
$\omega =$ aba, aaa

4. length of String ($|w|$)
→ Number of symbol in the String. eg:-
$w = abab$
$|w| = 4$

5. Zero length of String (ϵ)
→ Zero occurrence of symbol. Also called null String.

6. Power of Alphabet (Σ^k)
→ Set of all Strings exactly k length
eg:-
if $\Sigma = \{a, b\}$
Then,
$\Sigma^0 = \epsilon$
$\Sigma^1 = \{a, b\}$
$\Sigma^2 = \{aa, bb, ab, ba\}$
$\Sigma^3 = \{aaa, aab, aba, baa, abb, bab, bba, bbb\}$

7. Kleen closure (Σ^*)
→ Set of all possible String from the given alphabet
eg.
$\Sigma = \{a, b\}$

$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup --- \cup \Sigma^\alpha$

8. Positive kleen Closure (Σ^+)
→ set of all possible string except zero string.
eg:-
$$\Sigma = \{a, b\}$$

$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup --- \cup \Sigma^\infty$$

9. Prefix
→ Obtained by removing zero or more trailing symbol
eg:-
$w = abcd$
Prefix are;
$\{abcd, abc, ab, a, \epsilon\}$

10. suffix
→ obtained by removing zero or more leading symbol. eg:-
$w = abcd$
Suffix are;
$\{abcd, bcd, cd, d, \epsilon\}$

11. Substring
→ Obtained by removing prefix or suffix.
conditions for substring:-
→ order should be maintained
→ should be Consecutive

CamScanner

12. Language
→ Set of string, or subset of kleen Closure i.e
$(L \subset \Sigma^*)$

It's types are:- ① Finite language
② Infinite Language

Examples:-

Infinite
→ set of all strings ending with a, over the given alphabet $\Sigma = \{a, b\}$

$L = \{a, aa, aaa, ba, ----\}$

Finite:-
→ set of all strings exactly two length over the given alphabet $\Sigma = \{a, b\}$

$L = \{ab, ba, aa, bb\}$

13. Concatenation
→ Represented by $(\cdot)$
eg
$w_1 = aa$
$w_2 = bb$

$w_1 \cdot w_2 = aabb$
$w_2 \cdot w_1 = bbaa$
$w_1 \cdot w_2 \neq w_2 \cdot w_1$

Date
Page

14. Reverse of string. (w^R)
eg:-
$w = abb$ $w^R = bba$

Finite automata

→ Finite automata is a mathematical abstract machine that has a state and control; State moves one state to another state in response of external input and control may be deterministic or non-deterministic

It's types are:-

① FA with output
→ Mealy machine
→ Moore machine.

② FA without output
→ Deterministic Finite Automata (DFA)
→ Non-deterministic Finite Automata (NFA)
→ ϵ - NFA

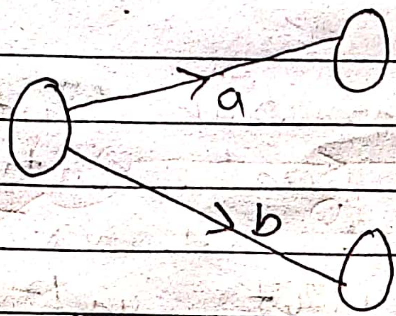
- Deterministic :- each input has exactly one transition.

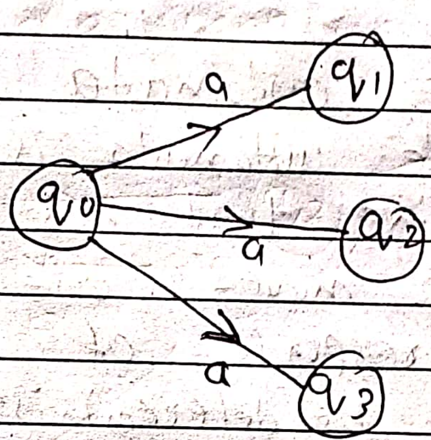
- Non Deterministic :- each input has zero or more transition.

eg:-

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

Deterministic | Non-Deterministic





Deterministic Finite Automata (DFA)

DFA is defined by 5-tuples:-

① Q : Finite set of states

② q_0 : Initial state ($q_0 \in Q$)

③ Σ : Finite set of non-empty symbol.

④ F : set of final state ($F \subseteq Q$)

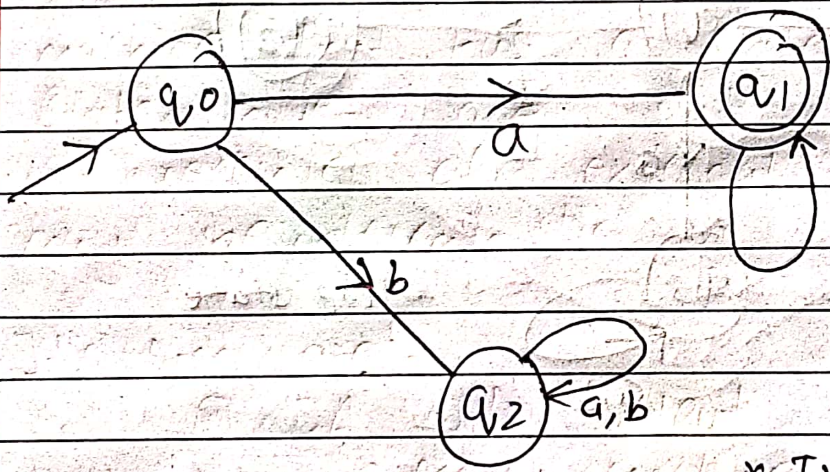
⑤ δ : Transition function ($\delta : Q \times \Sigma \rightarrow Q$)

(Q1) Construct a DFA set of all string starting with a, over $\Sigma = \{a, b\}$

Here,

$L = \{a, aa, aaa, ab, aba, ----\}$

* Take minimum string i.e 'a'



* Final state represented by double circle

* Trap state :- The state that doesn't lead to final state.

So,

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

$q_0 = \{q_0\}$

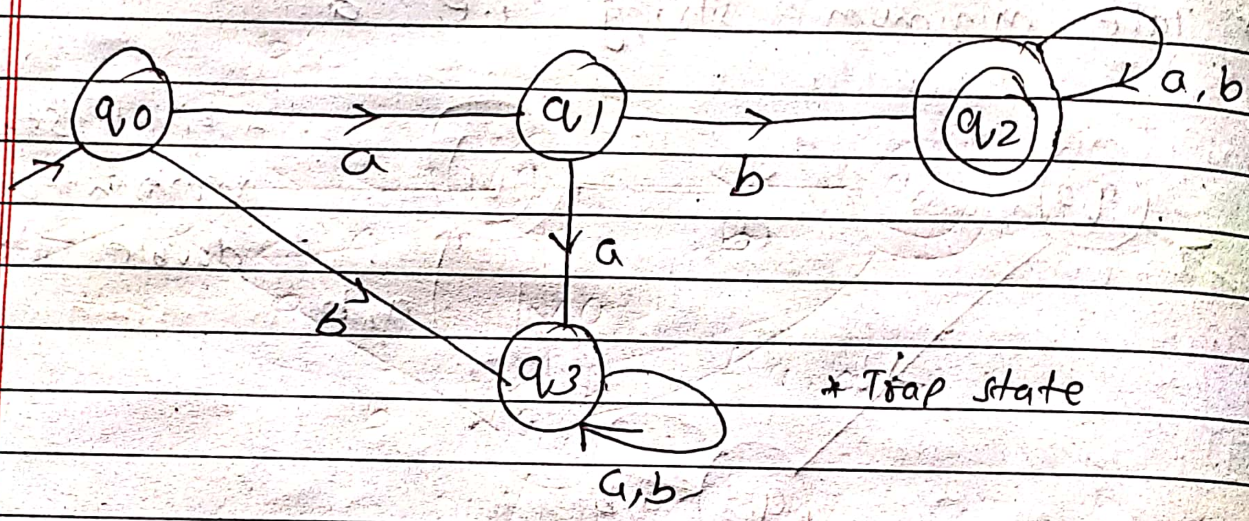
$\Sigma = \{a, b\}$

$F = q_1 \subseteq Q$

$\delta =$
$\delta(q_0, a) \longrightarrow q_1$
$\delta(q_0, b) \longrightarrow q_2$
$\delta(q_1, a) \longrightarrow q_1$
$\delta(q_1, b) \longrightarrow q_1$
$\delta(q_2, a) \longrightarrow q_2$
$\delta(q_2, b) \longrightarrow q_2$

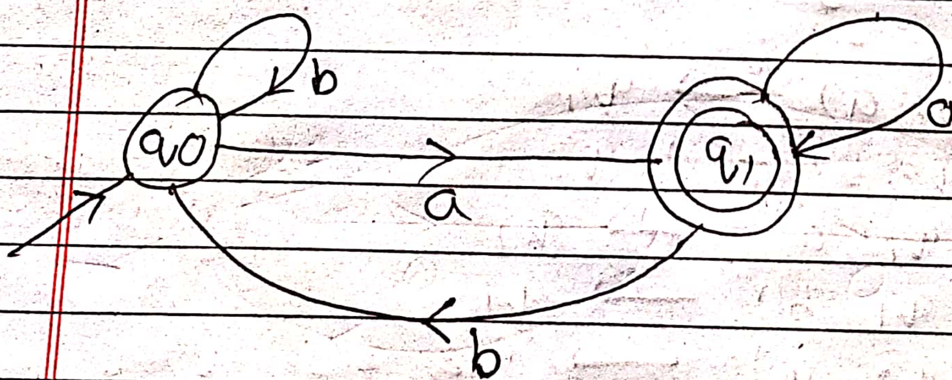
(Q2) Construct a DFA, set of all string starting with ab, over $\Sigma = \{a, b\}$
Here,

$L = \{ab, aba, \cancel{bb}\ abb, abab, abbb, \ldots\}$



(Q3) Construct a DFA, set of all string ending with a, over $\Sigma = \{a, b\}$
Here,

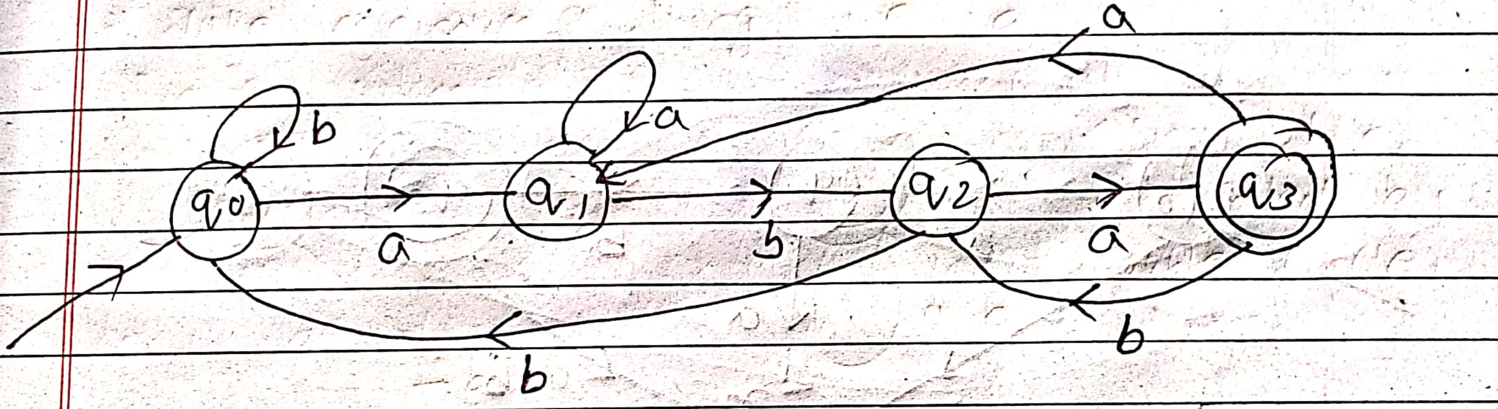
$L = \{a, aa, ba, aaa, aba, baa, bba, \ldots\}$



Date ______
Page ______

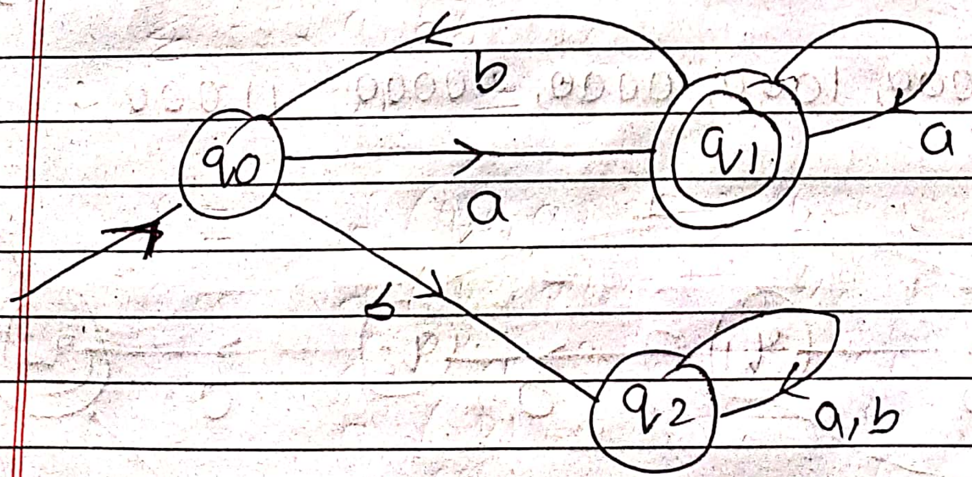
(Q4) Construct a DFA, set of all strings ending with 'aba' over $\Sigma = \{a, b\}$
Here,

$L = \{aba, aaba, baba, aaaba, bbaba \ldots\}$



(Q5) Construct a DFA, set of all strings starting or ending with 'a' over $\Sigma = \{a, b\}$
Here,

$L = \{a, aa, aaa, aba, abba, aba, \ldots\}$

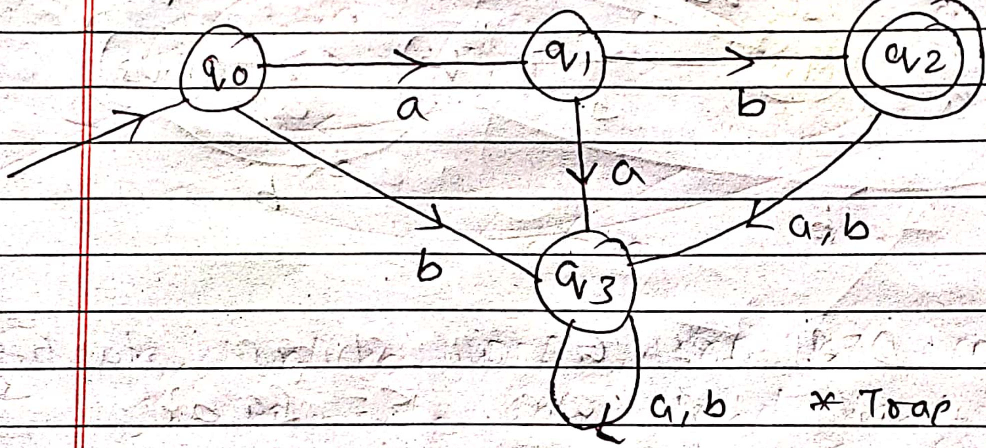


* Trap state

Q6 Construct a DFA, set of all strings starting with 'ab' and ending with even length of string over $\Sigma = \{a, b\}$

Here,

$L = \{ab, abaa, abab, abbb, abaab, \ldots\}$

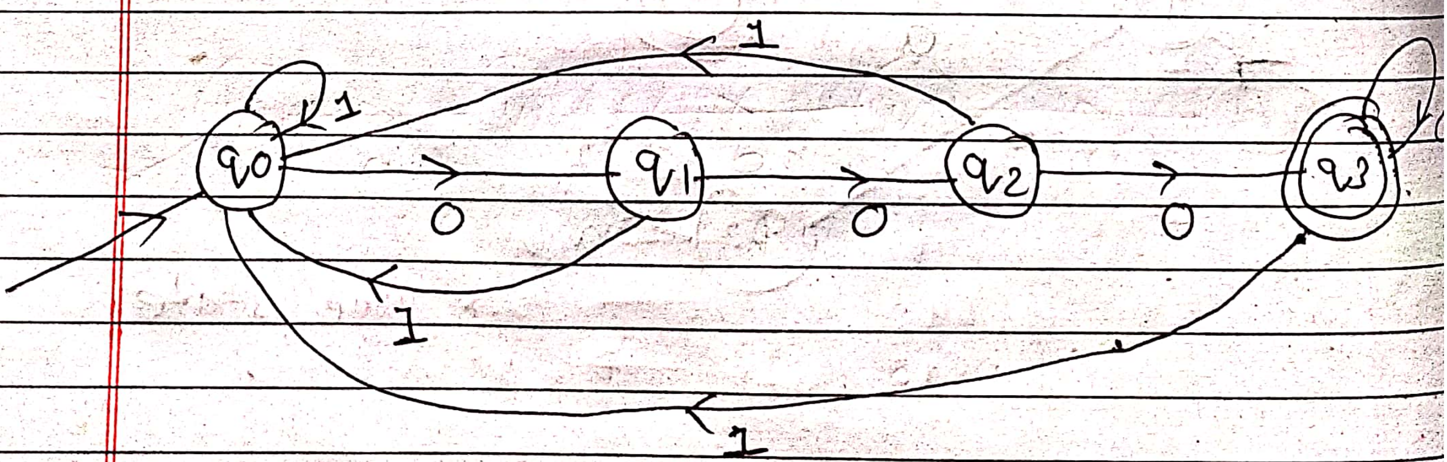


* Trap state

Q7. Construct a DFA, set of all strings ending with three consecutive zero over $\Sigma = \{0, 1\}$

Here,

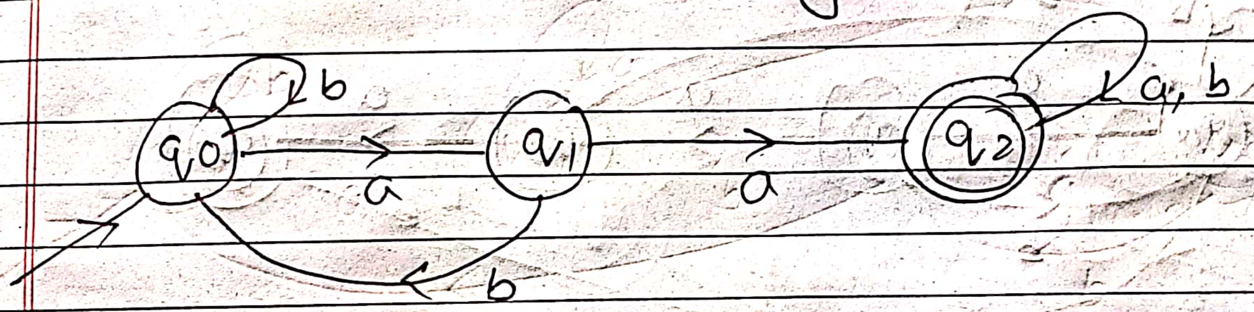
$L = \{000, 1000, 0000, 00000, 11000, \ldots\}$



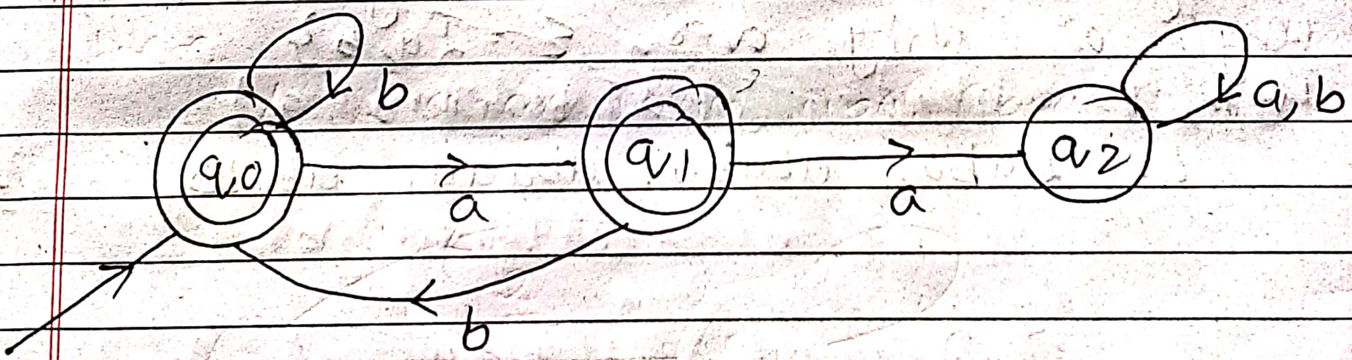
Date ______
Page ______

(Q8) Construct a DFA, set of all strings not containing 'aa' as a substring over $\Sigma = \{a, b\}$
Here
$L = \{a, b, ab, aba, bba, abba, \ldots\}$

* First construct containing.



* Now complement the states.

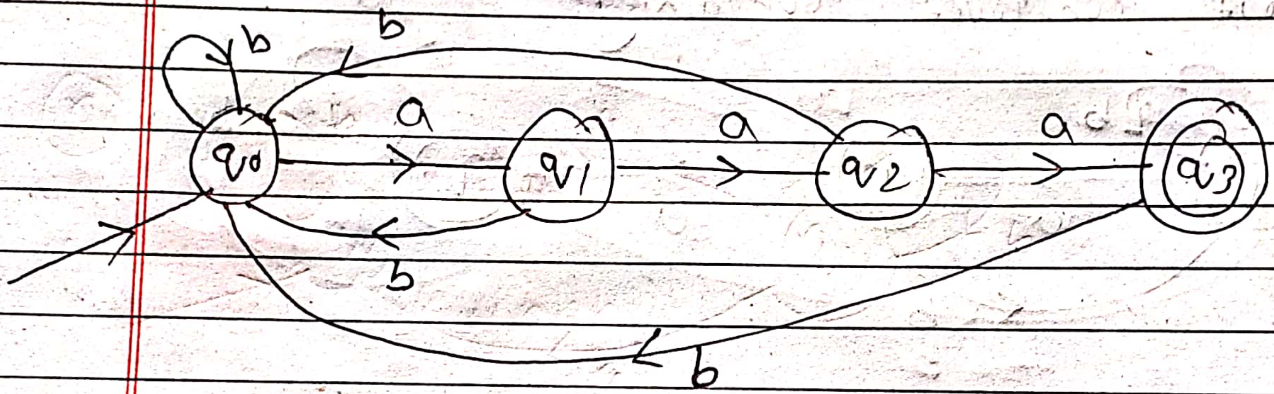


* This is final figure.

Date ____
Page ____

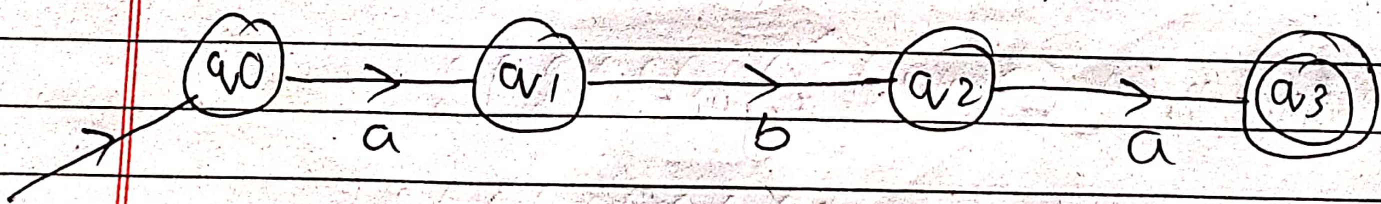
(Q9) Construct a DFA, set of all strings, containing three consecutive 'a' over $\Sigma = \{a, b\}$
Here,

$L = \{aaa, baaa, aaab, aaaa, baaaa \ldots\}$



(Q10) Construct a DFA, set of all string starting with aba and ending with odd length of string over $\Sigma = \{a, b\}$
Here,

$L = \{aba, ababa, abaaa, ababb, \ldots\}$



How the DFA process the string.

Let,

$w = a_1, a_2, a_3, \text{----} a_n$

$\delta(q_0, a_1) = q_1$
$\delta(q_1, a_2) = q_2$
$\delta(q_2, a_3) = q_3$
$\vdots$
$\delta(q_{n-1}, a_n) = q_n \in$ Final state (F)

If q_n belongs to final state we accept $w = a_1, a_2, a_3, \text{----}, a_n$

Extended transition function of DFA

→ It is represented by $\hat{\delta}$. It has two argument (state and string) eg:-

$\hat{\delta}(q_0, w) = P \in F$

Proof by mathematical Induction :-

<u>Basic</u>

$w = \epsilon$

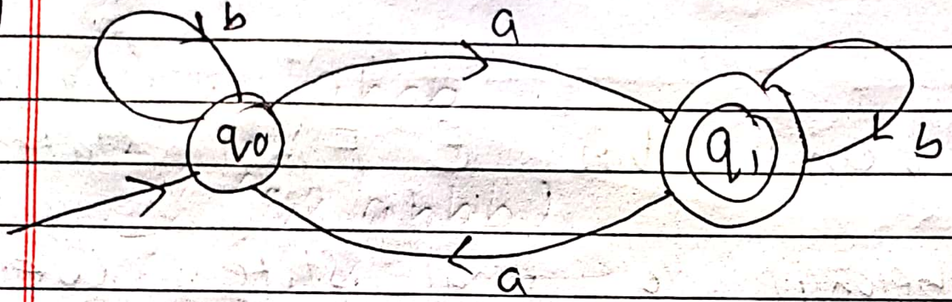
$\hat{\delta}(q_0, \epsilon) = q_0$

<u>Induction</u>

$w = xa$ ($\because$ where x is a substring without last symbol $a \neq \epsilon$)

$$\hat{\delta}(q_0, w) = \delta(\hat{\delta}(q_0, x), a)$$
$$= \delta(q, a)$$
$$= p \in F$$

9



1 $w = aaba$. verify !!

$$\hat{\delta}(q_0, aaba) = \delta(\hat{\delta}(q_0, aab), a)$$
$$= \delta(\delta(\hat{\delta}(q_0, aa), b), a)$$
$$= \delta(\delta(\delta(\hat{\delta}(q_0, a), a), b), a)$$
$$= \delta(\delta(\delta(\delta(\hat{\delta}(q_0, \epsilon), a), a), b), a)$$
$$= \delta(\delta(\delta(\delta(q_0, a), a), b), a)$$
$$= \delta(\delta(\delta(q_1, a), b), a)$$
$$= \delta(\delta(q_0, b), a)$$

$= \delta(q_0, a)$

$= q_1 \in F$

As final state is seen, we accept $w = aaba$

(2) $w = abba$ .. verify !!

$\hat{\delta}(q_0, abba) = \delta(\hat{\delta}(q_0, abb), a)$

$= \delta(\delta(\hat{\delta}(q_0, ab), b), a)$

$= \delta(\delta(\delta(\hat{\delta}(q_0, a), b), b), a)$

$= \delta(\delta(\delta(\delta(\hat{\delta}(q_0, \epsilon), a), b), b), a)$

$= \delta(\delta(\delta(\delta(q_0, a), b), b), a)$

$= \delta(\delta(\delta(q_1, b), b), a)$

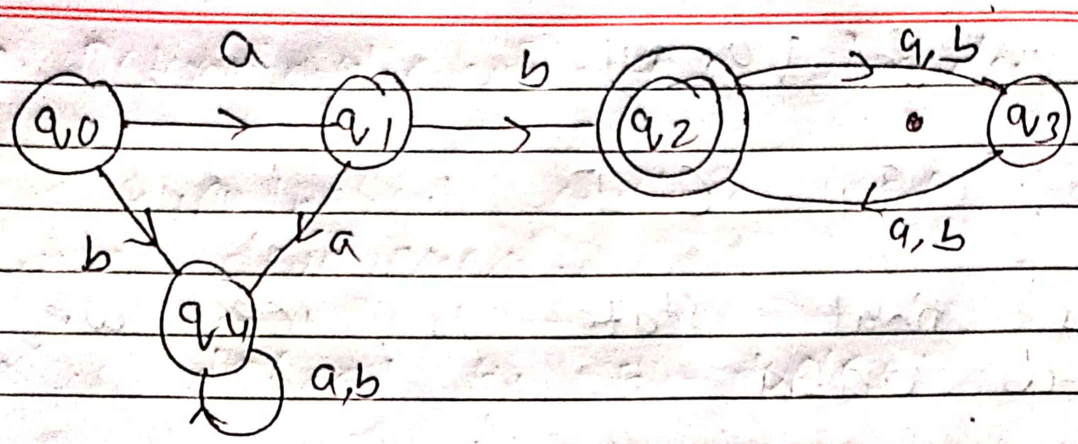
$= \delta(\delta(q_1, b), a)$

$= \delta(q_1, a)$

$= q_0 \notin F$

so, the given string is rejected.

Q



(1) $w = abbb$

$= \hat{\delta}(q_0, abbb)$

$= \delta(\hat{\delta}(q_0, abb), b)$

$= \delta(\delta(\hat{\delta}(q_0, ab), b), b)$

$= \delta(\delta(\delta(\hat{\delta}(q_0, a), b), b), b)$

$= \delta(\delta(\delta(\delta(\hat{\delta}(q_0, \epsilon), a), b), b), b)$

$= \delta(\delta(\delta(\delta(q_0, a), b), b), b)$

$= \delta(\delta(\delta(q_1, b), b), b)$

$= \delta(\delta(q_2, b), b)$

$= \delta(q_3, b)$

$= q_2 \in F$ Accept $w = abbb$

(2) $w = ababa$

$\hat{\delta}(q_0, ababa)$

$= \delta(\hat{\delta}(q_0, abab), a)$

$= \delta(\delta(\hat{\delta}(q_0, aba), b), a)$

$= \delta(\delta(\delta(\hat{\delta}(q_0, ab), a), b), a)$

$= \delta(\delta(\delta(\delta(\hat{\delta}(q_0, a), b), a), b), a)$

$= \delta(\delta(\delta(\delta(\delta(\hat{\delta}(q_0, \epsilon), a), b), a), b), a)$

$= \delta(\delta(\delta(\delta(\delta(q_0, a), b), a), b), a)$

$= \delta(\delta(\delta(\delta(q_1, b), a), b), a)$

$= \delta(\delta(\delta(q_2, a), b), a)$

$= \delta(\delta(q_3, b), a)$

$= \delta(q_2, a)$

$= q_3 \notin F$ Reject $w = ababa$ //

Non-Deterministic Finite Automata (NFA)

→ It is hypothetical machine used for software development.

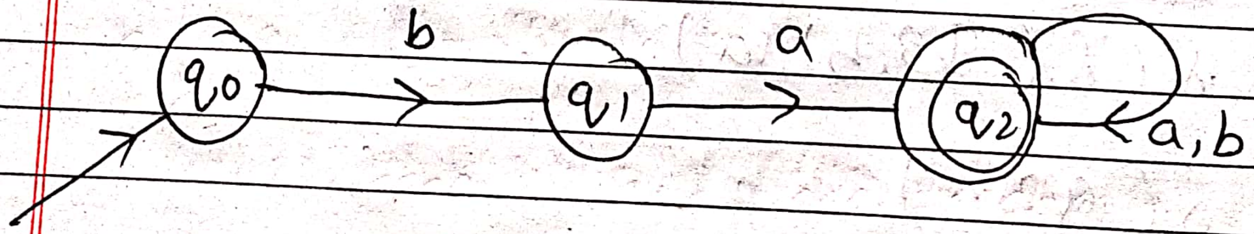
NFA is defined by 5-tuples :-

1. Q
2. q_0
3. F
4. Σ
5. $\delta : Q \times \Sigma \longrightarrow 2^Q$

Q1. Construct a NFA set of all string starting with ba over the alphabet $\Sigma = \{a, b\}$

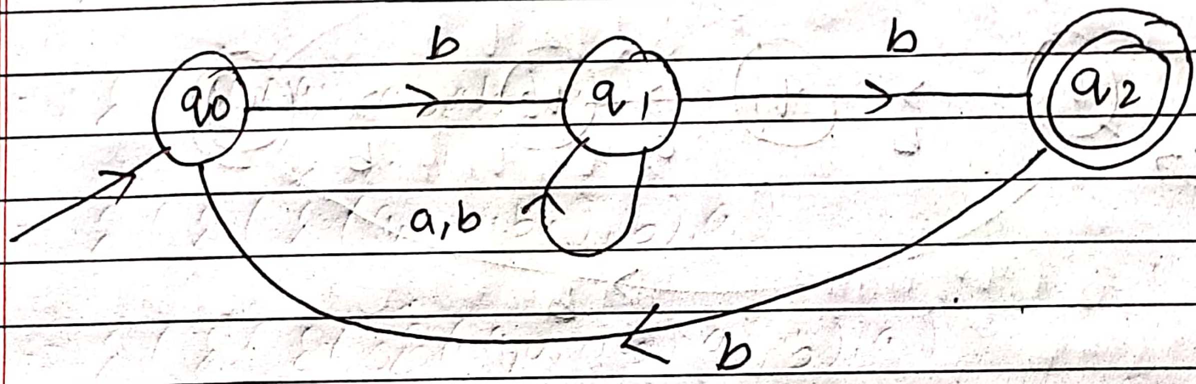
Here,

$L = \{ba, baa, baaa, bab, babb, \ldots\}$



Date
Page

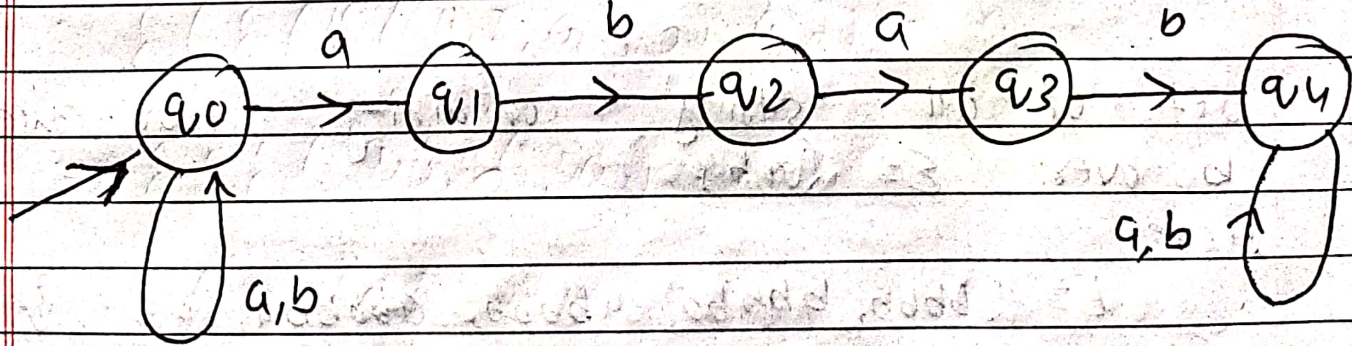
(Q2) Construct a NFA set of all string starting and ending with b over $\Sigma = \{a, b\}$



(Q3) Construct a NFA set of all string containing abab as a substring over $\Sigma = \{a, b\}$

Here

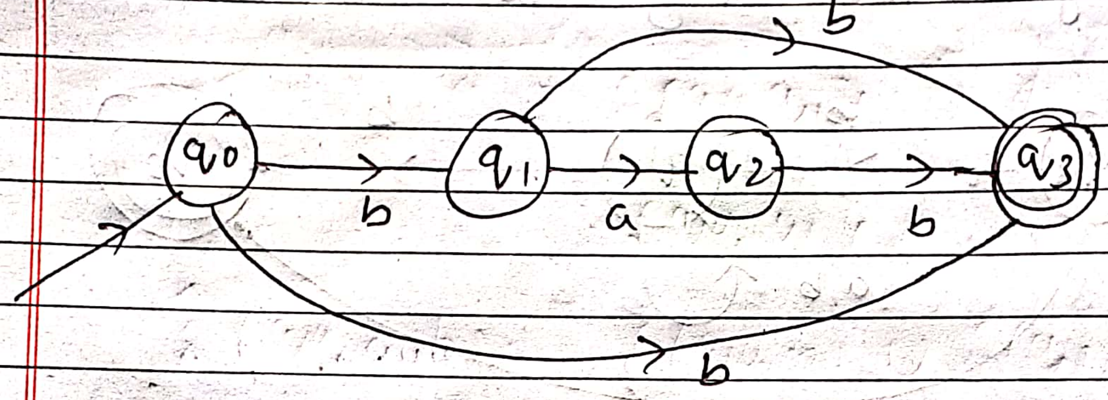
$L = \{abab, aabab, ababa, \ldots\}$



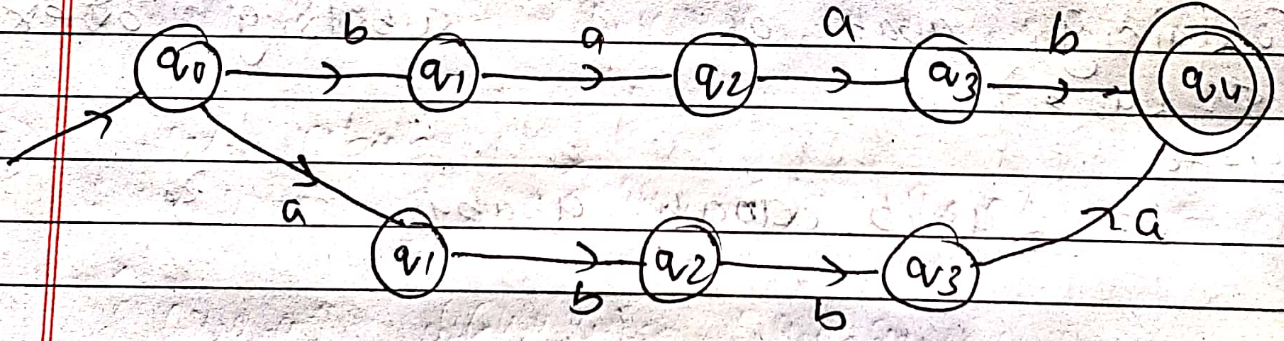
CamScanner

Date
Page

(Q4) $L = \{b, bb, bab\}$



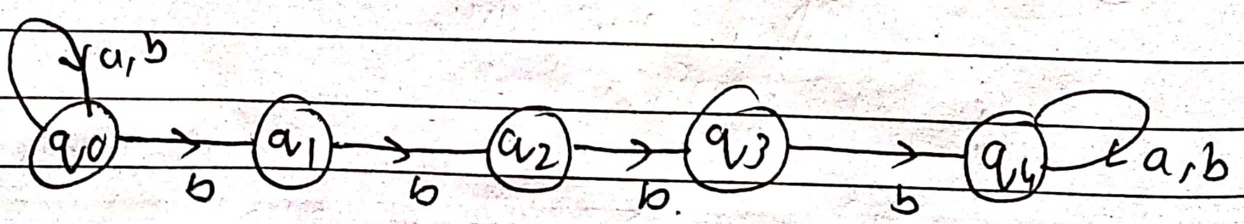
(Q5) $L = \{baab, abba\}$



(Q6) set of all string containing four consecut b over $\Sigma = \{a, b\}$

Here,

$L = \{bbbb, bbbba, abbbb, abbbba, \ldots\}$



Extended transition function of NFA

$$\hat{\delta}(q_0, w) \cap F \neq \phi$$

<u>Basic</u> : $\hat{\delta}(q_0, \epsilon) = \{q_0\}$

<u>Inductive</u> : $w = xa$, where x is a substring without last symbol $a \neq \epsilon$

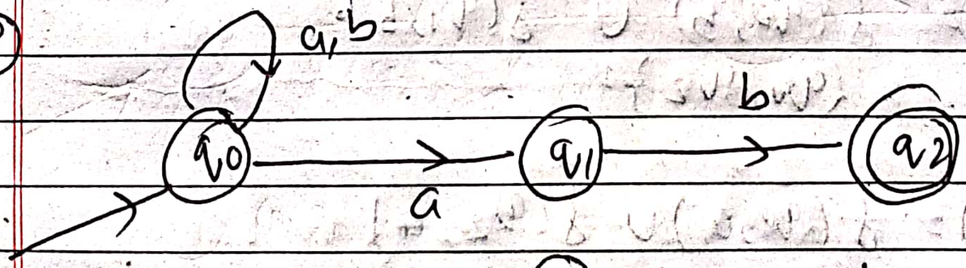
$$\hat{\delta}(q_0, w) = \delta(\hat{\delta}(q_0, x), a)$$

$$\hat{\delta}(q_0, x) = \{r_1, r_2, r_3, \dots r_m\}$$

$$\bigcup_{i=1}^{m} \delta_N(r_i, a) = \{P_1, P_2, \dots P_n\} \cap F \neq \phi$$

*meaning atleast one final state

Q.



(1) $w = aab$

solution

$$\hat{\delta}(q_0, \epsilon) = \{q_0\}$$

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

$\hat{\delta}(q_0, aa) = \delta(q_0, a) \cup \delta(q_1, a)$

$= \delta\{q_0, q_1\} \cup \phi$

$= \{q_0, q_1\}$

$\Rightarrow \hat{\delta}(q_0, aab) = \delta(q_0, b) \cup \delta(q_1, b)$

$= \{q_0, q_2\} \in F$

So Accept//

(2) $w = abab$

$\hat{\delta}(q_0, \epsilon) = \{q_0\}$

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

$\hat{\delta}(q_0, ab) = \delta(q_0, b) \cup \delta(q_1, b)$

$= \{q_0, q_2\}$

$\hat{\delta}(q_0, aba) = \delta(q_0, a) \cup \delta(q_2, a)$

$= \{q_0, q_1\} \cup \phi$

$= \{q_0, q_1\}$

$\hat{\delta}(q_0, abab) = \delta(q_0, b) \cup \delta(q_1, b)$

$= \{q_0, q_2\} \in F$

So Accepted//

Conversion of NFA into DFA (sub-set construction algorithm)

Rule :-

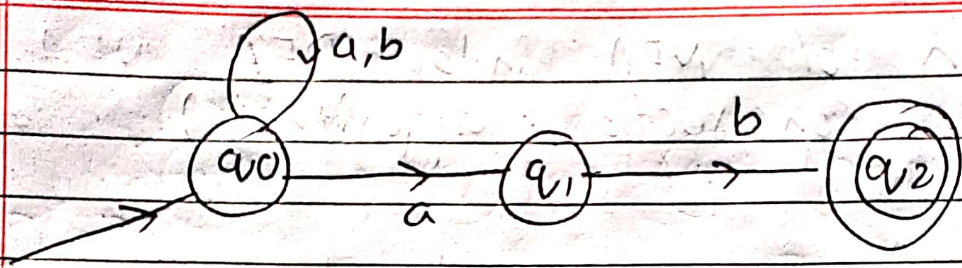
NFA	DFA
① If q_0 is initial state	$\{q_0\}$ is also the initial state
② If total state is 'n'	The total state is '2^n'
③ If final state set is $\{q_2\}$	If any set has q_2 then it is final state
④ $\{F\}$	$\{F\}$

Steps :-

① Making subset table

② Converting to DFA

9

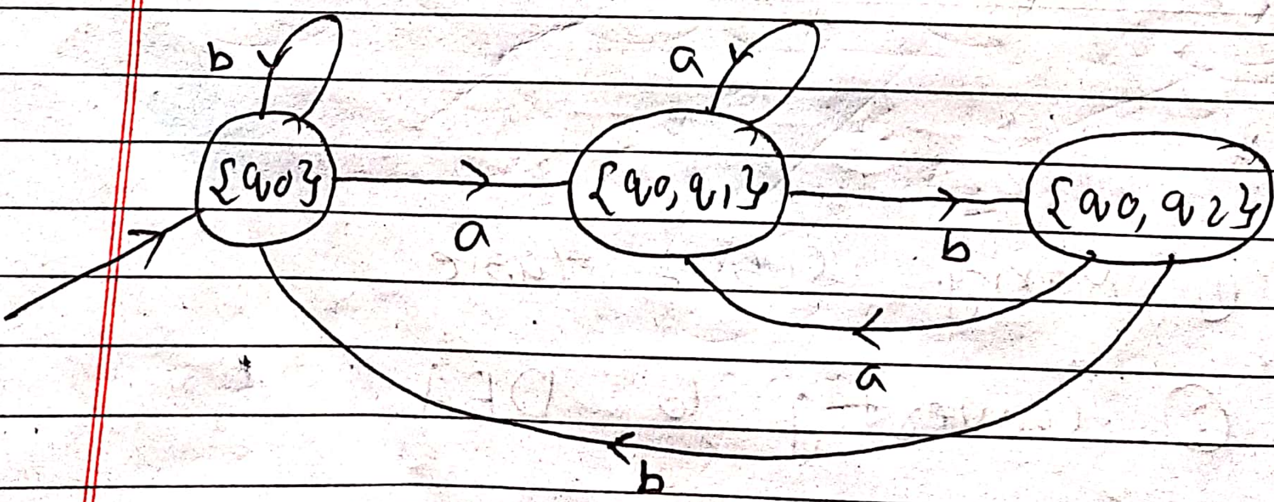


Solution :-

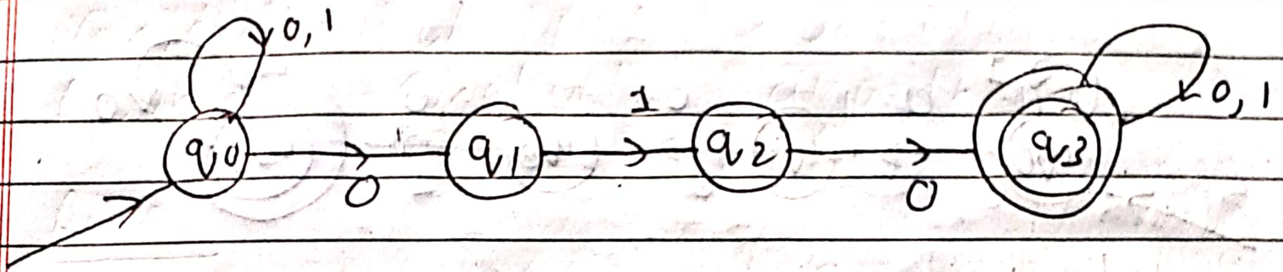
State	Input Symbol a	Input Symbol b
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1\} \cap \phi$ $\{q_0, q_1\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0, \}$

* other are the unreachable state

Now constructing DFA



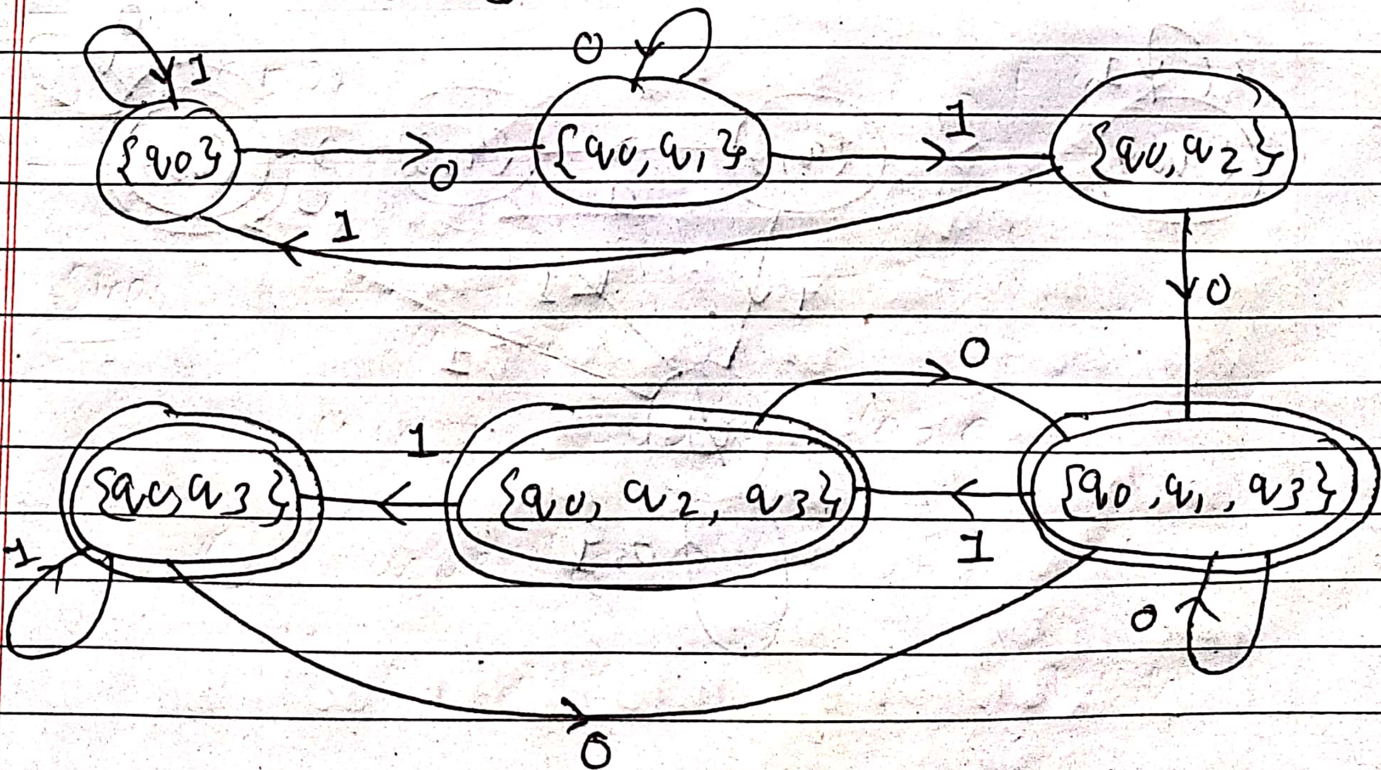
9



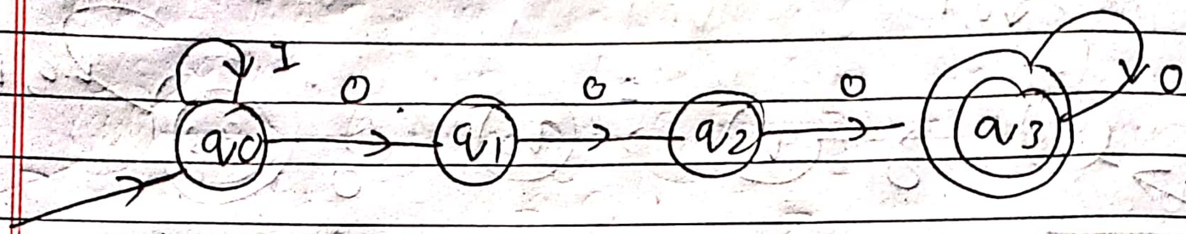
Solution:

States	Input symbol 0	Input symbol 1
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1, q_3\}$	$\{q_0\}$
$\{q_0, q_1, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_2, q_3\}$
$\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$
$\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$

Now constructing DFA :-



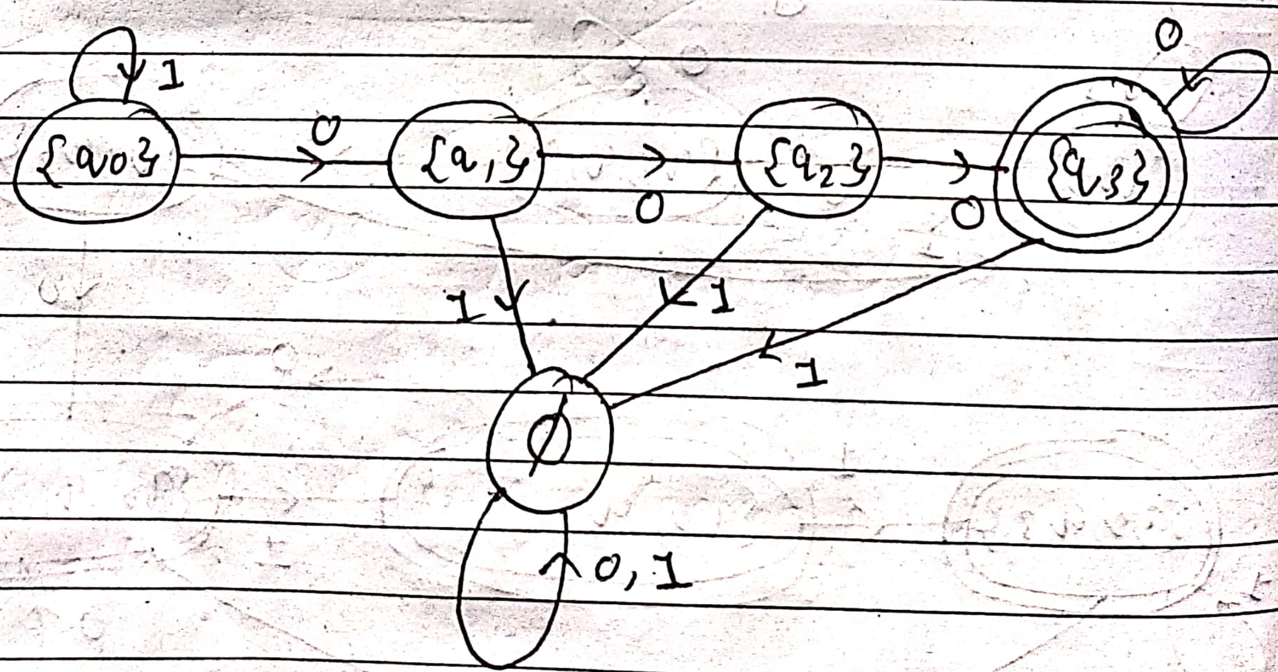
9



solution:-

States	Input symbol 0	1
{q0}	{q1}	{q0}
{q1}	{q2}	∅
{q2}	{q3}	∅
{q3}	{q3}	∅
∅	∅	∅

Constructing DFA :-



Language of NFA

$$L(NFA) = \{ \hat{\delta}(q_0, w) \cap F \neq \phi \}$$

ε-NFA

→ It is an extension of NFA. It allows transition on Epsilon (ε), without consuming input symbol

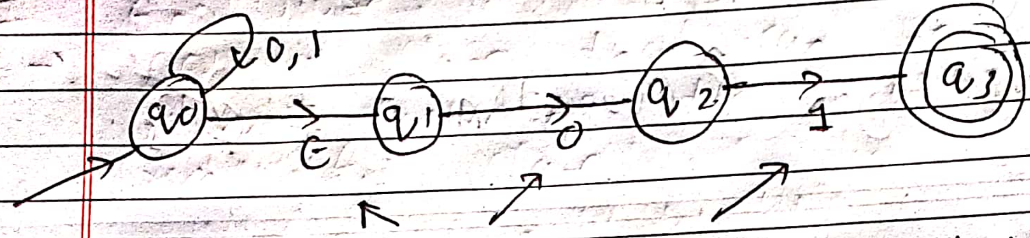
ε-NFA is defined by 5-tuples:-

1. Q:
2. q_0:
3. F:
4. Σ
5. $\delta : Q \times \Sigma \cup \{\epsilon\} \longrightarrow 2^Q$

(Q1) Construct a ε-NFA set of all string ending with 01 over the given $\Sigma = \{0, 1\}$

Here

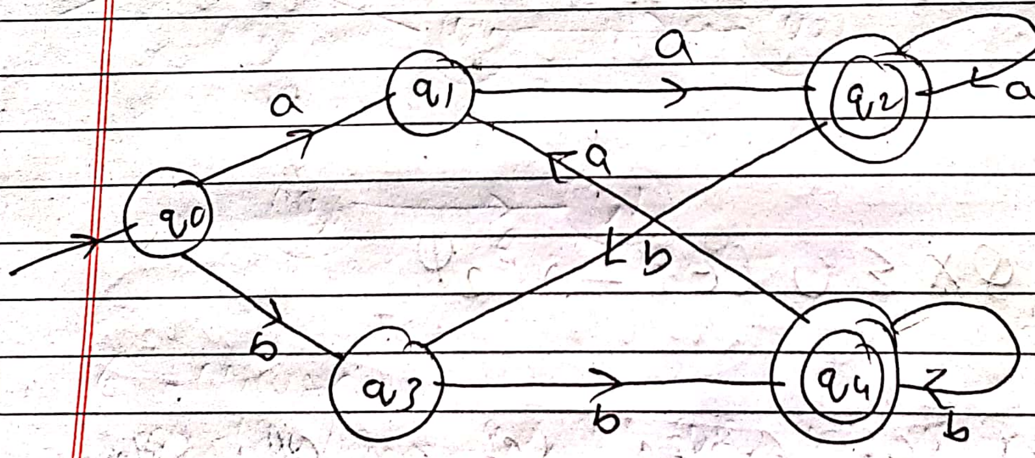
$$L = \{01, 101, 001, 1101, ----\}$$



* ε can be placed anywhere.

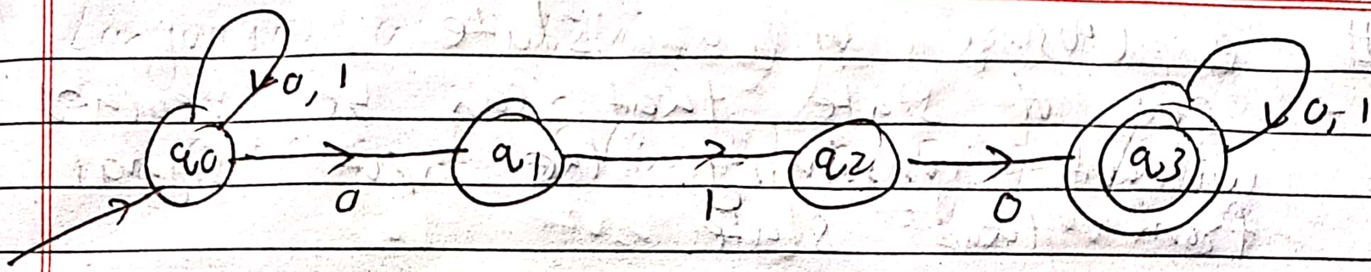
(Q2) Construct ε-NFA Starting or ending with aa or bb over Σ = {a, b}

Here,
L = {aa, bb, aaa, ~~bbb~~, aab, abb, aabb}



(Q3) Construct ε-NFA Containing 010 as a substring

Here
L = {010, 0010, 1010, 0101, 001010}

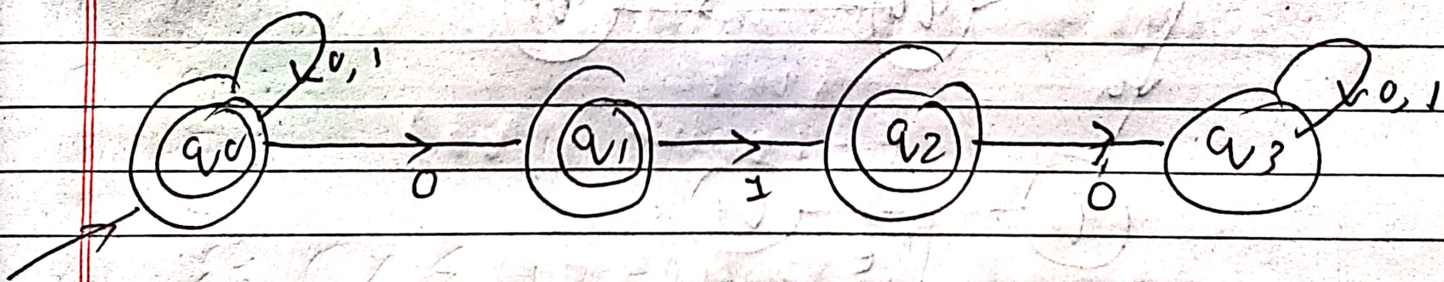


* For doing do not containing we simply complement the state.

(Q4) Construct ϵ NFA donot containing 010 as a ~~str~~ substring.

Here

$L = \{1, 0, 01, 10, 101, 011, 110, ---\}$



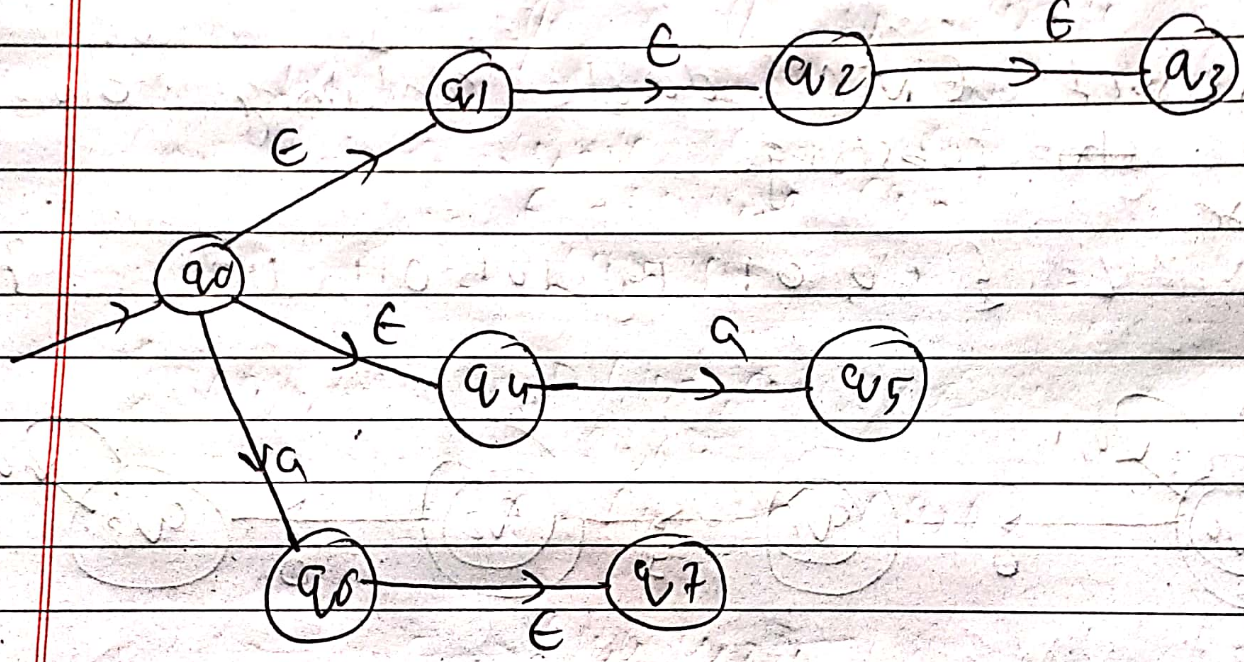
Identity law of concatenation

$$\boxed{a \cdot \epsilon = \epsilon \cdot a = a}$$

ε-closure of a state

→ set of state that can be reached without consuming the input signal from that state.

eg:-



ε-closure

1. (q_0) = {q_0, q_1, q_2, q_3, q_4}
2. (q_1) = {q_1, q_2, q_3}
3. {q_2) = {q_2, q_3}
4. {q_3) = {q_3}
5. (q_4) = {q_4}
6. {q_5} = {q_5}
7. {q_6} = {q_6, q_7}
8. {q_7} = {q_7}

Extended transition function of E-NFA

$$\hat{\delta}(q_0, w) \cap F \neq \phi$$

<u>Basic</u>

$w = \epsilon$

$\hat{\delta}(q_0, \epsilon) = \epsilon\text{-closure}(q_0)$

<u>Induction</u>

$w = xa$ — * x is substring without last symbol

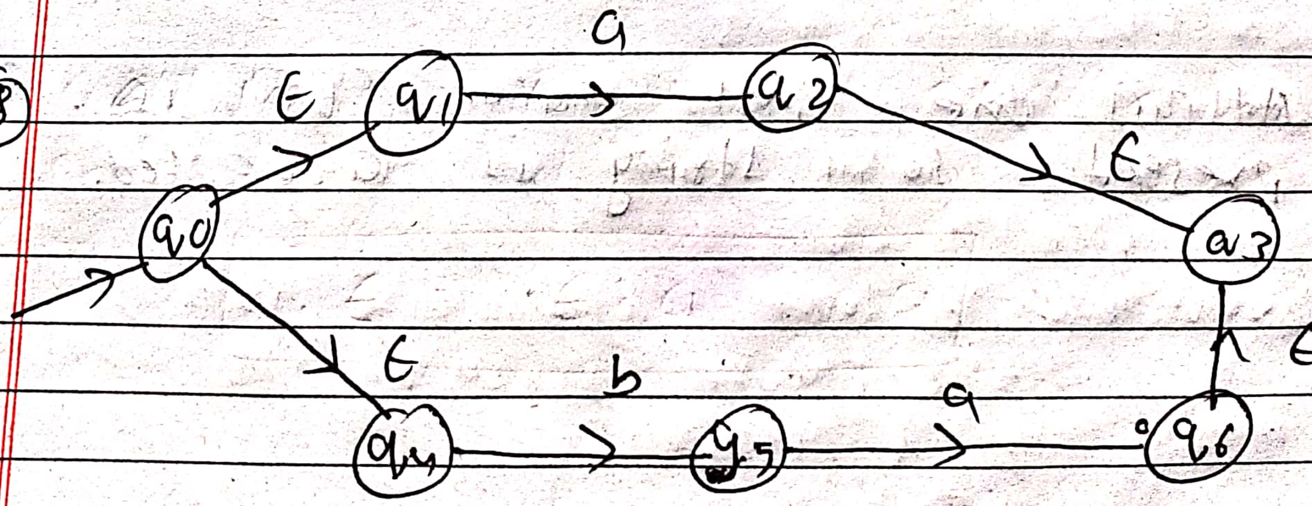
$\hat{\delta}(q_0, w) = \delta(\hat{\delta}(q_0, x), a)$

$\hat{\delta}(q_0, x) = \{r_1, r_2, r_3, \ldots, r_m\}$

$\bigcup_{i=1}^{m} \delta(r_i, a) = \{P_1, P_2, P_3, \ldots, P_n\}$

$= \bigcup_{J=1}^{n} \epsilon\text{-closure}(P_J)$

Q



Input String ($w = ba$)

$\hat{\delta}(q_0, \epsilon) = \epsilon\text{-closure}(q_0)$

$= \{q_0, q_1, q_4\}$

$\hat{\delta}(q_0, b) = \delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_4, b)$

$= \phi \cup \phi \cup \{q_5\}$

$= \{q_5\}$

$= \epsilon\text{-closure}(q_5)$

$= \{q_5\}$

$\hat{\delta}(q_0, ba) = \delta(q_5, a)$

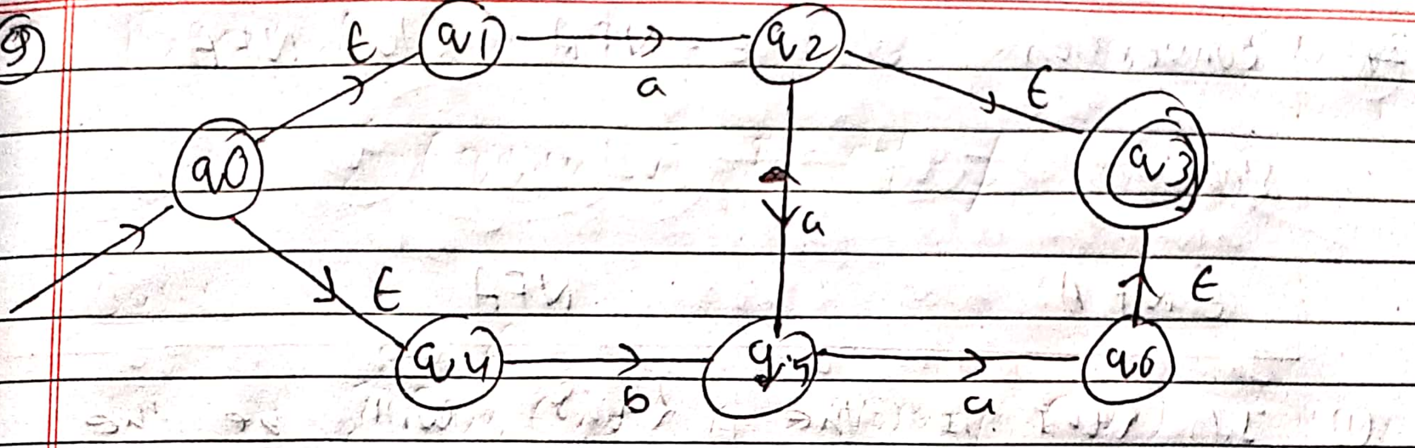
$= \{q_6\}$

$= \epsilon\text{-closure}(q_6)$

$= \{q_6, q_3\}$

Atleast one final state 'q_3' is present so string is accepted.

9



Input string (w= aaa)

$\hat{\delta}(q0, \epsilon) = \epsilon\text{-closure}(q0)$
$= \{q0, q1, q4\}$

$\hat{\delta}(q0, a) = \delta(q0, a) \cup \delta(q1, a) \cup \delta(q4, a)$
$= \phi \cup \{q2\} \cup \phi$
$= \{q2\}$
$= \epsilon\text{-closure}(q2)$
$= \{q2, q3\}$

$\hat{\delta}(q0, aa) = \delta(q3, a) \cup \delta(q3, a)$
$= \{q5\} \cup \phi$
$= \epsilon\text{-closure}(q5)$
$= \{q5\}$

$\hat{\delta}(q0, aaa) = \delta(q5, a)$
$= \{q6\}$
$= \epsilon\text{-closure}(q6)$
$= \{q6, q3\}$

Final state found so accepted //

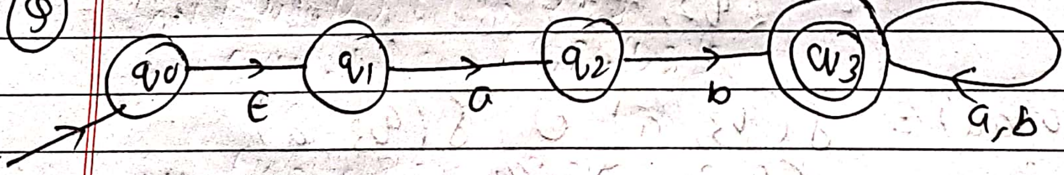
Conversion of ϵ-NFA into NFA

Steps:-

	ENFA	NFA
①	If $\{q_0\}$ is the initial state	$\{q_0\}$ will be the initial state
②	If $\{q_3\}$ is the final state	$\{q_3\}$ will be the final state

$$\boxed{\delta_N(q,a) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q), a))}$$

Q



Initial state of resulting NFA is q_0
($\because$ q_0 is initial state of ϵ-NFA)

$$\delta_N(q_0,a) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q_0), a))$$
$$= \epsilon\text{-closure}(\delta(\{q_0, q_1\}, a))$$
$$= \epsilon\text{-closure}(\{q_2\})$$
$$= \{q_2\}$$

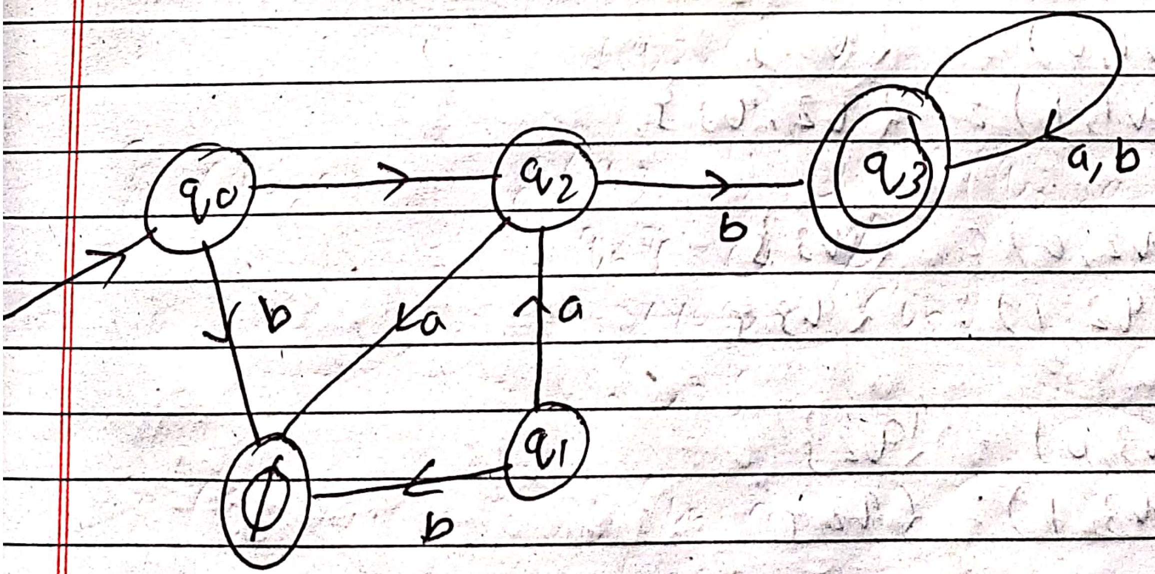
$\delta_N(q_0, b) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q_0), b))$
$= \epsilon\text{-closure}(\delta(\{q_0, q_1\}, b))$
$= \epsilon\text{-closure}(\phi)$
$= \phi$

$\delta_N(q_1, a) = \{q_2\}$
$\delta_N(q_1, b) = \phi$

$\delta_N(q_2, a) = \phi$
$\delta_N(q_2, b) = \{q_3\}$

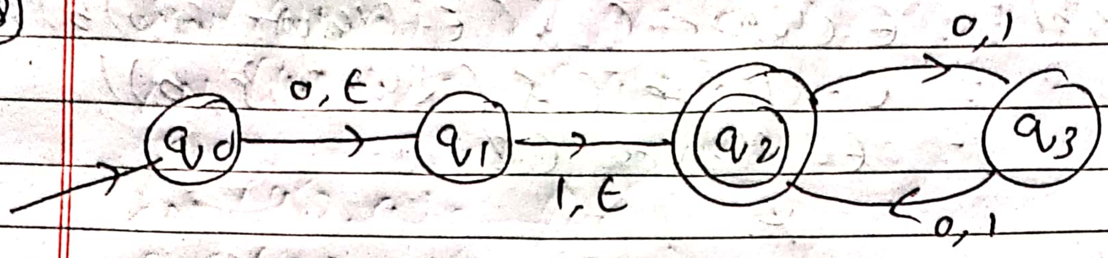
$\delta_N(q_3, a) = \{q_3\}$
$\delta_N(q_3, b) = \{q_3\}$

Drawing the figure.



* This is the required NFA

(8)



Initial state of resulting NFA is q_0

$\delta_N(q_0, 0) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q_0), 0))$
$= \epsilon\text{-closure}(\delta(\{q_0, q_1, q_2\}), 0)$
$= \epsilon\text{-closure}(\{q_1, q_3\})$
$= \{q_1, q_2, q_3\}$

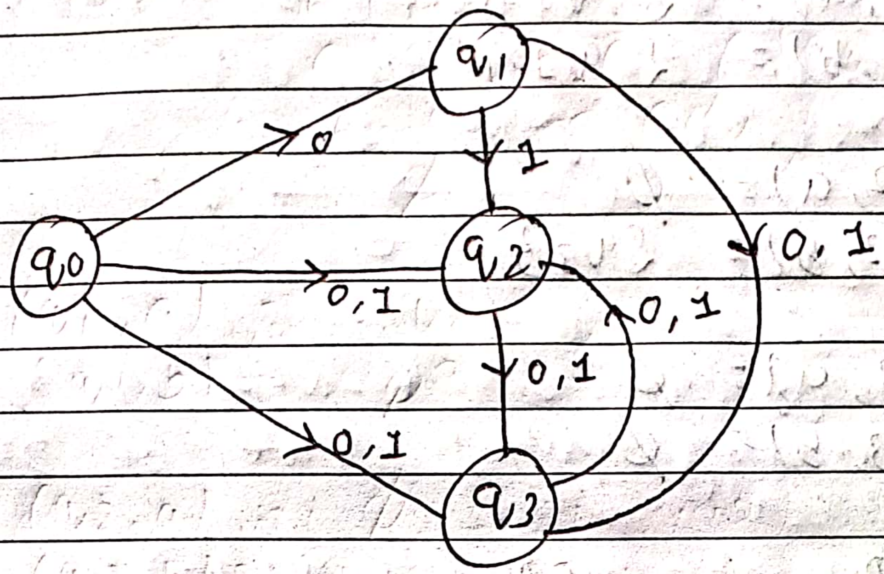
$\delta_N(q_0, 1) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q_0), 1))$
$= \epsilon\text{-closure}(\delta(\{q_0, q_1, q_2\}), 1)$
$= \epsilon\text{-closure}(\{q_2, q_3\})$
$= \{q_2, q_3\}$

$\delta_N(q_1, 0) = \{q_3\}$
$\delta_N(q_1, 1) = \{q_2, q_3\}$

$\delta_N(q_2, 0) = \{q_3\}$
$\delta_N(q_2, 1) = \{q_3\}$

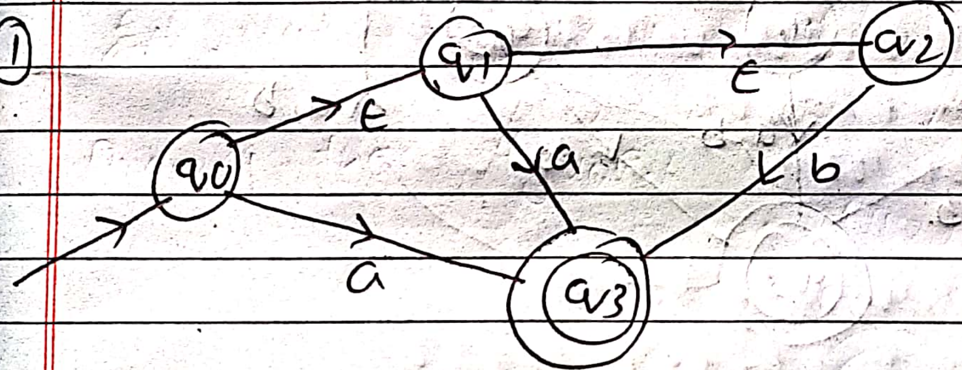
$\delta_N(q_3, 0) = \{q_2\}$
$\delta_N(q_3, 1) = \{q_2\}$

Now, Drawing NFA;



Assignment Questions :-

(1)



Initial State of resulting NFA is q0

$\delta_N(q0, a) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q0), a))$
$= \epsilon\text{-closure}(\delta(\{q0, q1, q2\}), a)$
$= \epsilon\text{-closure}(\{q_3\})$
$= \{q_3\}$

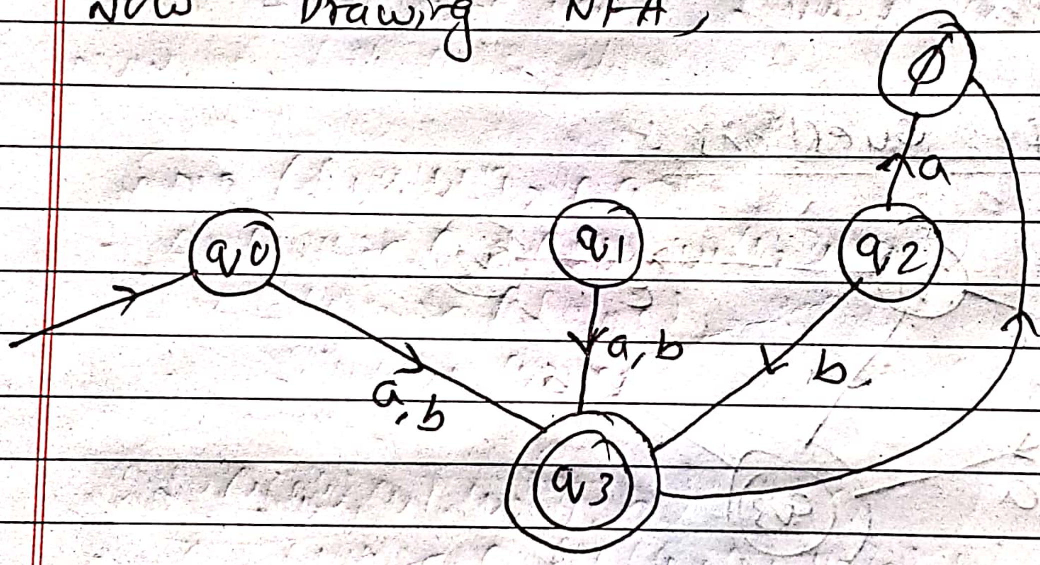
$\delta_N(q0, b) = \{q_3\}$

$\delta_N(q_1, a) = \{q_3\}$
$\delta_N(q_1, b) = \{q_3\}$

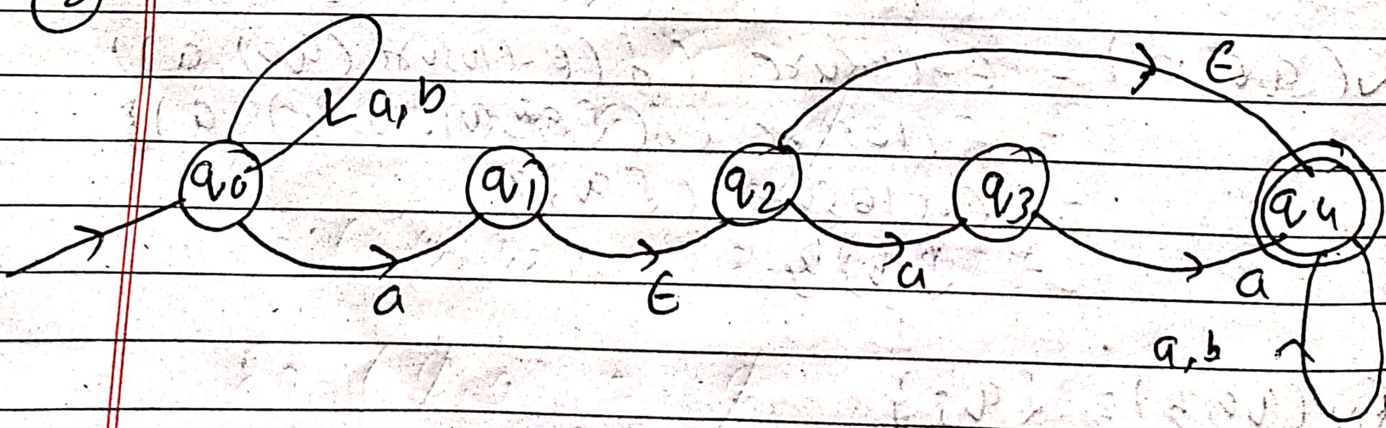
$\delta_N(q_2, a) = \phi$
$\delta_N(q_2, b) = \{q_3\}$

$\delta_N(q_3, a) = \phi$
$\delta_N(q_3, b) = \phi$

Now Drawing NFA;



(2)



Initial state of resulting NFA is q_0

Date
Page

$\delta_N(q_0, a) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(q_0), a))$
$= \epsilon\text{-closure}(\delta(q_0, a))$
$= \epsilon\text{-closure}(\{q_0, q_1\})$
$= \cancel{\{q_1, q_2\}}\ \{q_0, q_1, q_2, q_4\}$

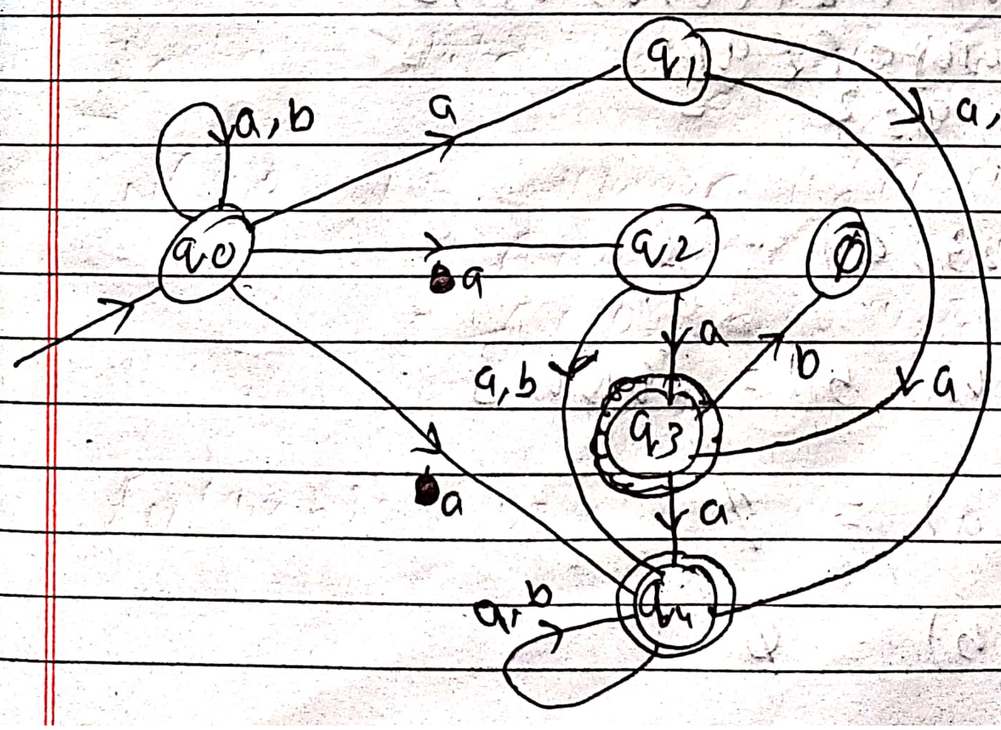
$\delta_N(q_0, b) = \{q_0\}$

$\delta_N(q_1, a) = \cancel{\{q_3\}}\ \{q_3, q_4\}$
$\delta_N(q_1, b) = \{q_4\}$

$\delta_N(q_2, a) = \{q_3, q_4\}$
$\delta_N(q_2, b) = \{q_4\}$

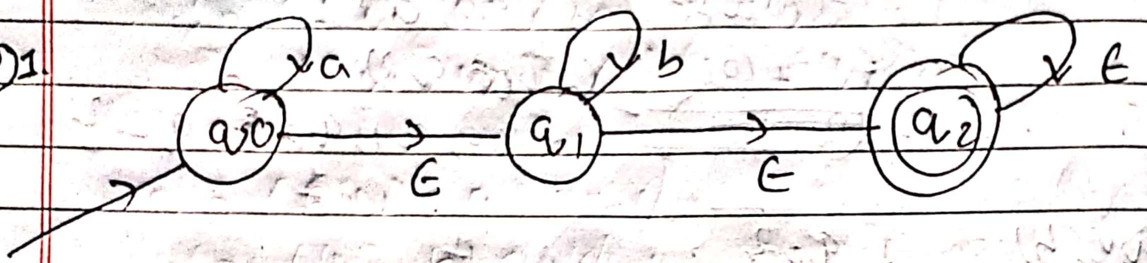
$\delta_N(q_3, a) = \{q_4\}$
$\delta_N(q_3, b) = \emptyset$

$\delta_N(q_4, a) = \{q_4\}$
$\delta_N(q_4, b) = \{q_4\}$



Conversion of ε-NFA to DFA

Q1.



* Initial state of ε-NFA is q_0. So the resulting initial state of DFA will be

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

Now,

$$\delta_D(\{q_0, q_1, q_2\}, a)$$
$$= \epsilon\text{-closure}(\delta(\{q_0, q_1, q_2\}, a))$$
$$= \epsilon\text{-closure}(\{q_0\})$$
$$= \{q_0, q_1, q_2\}$$

$$\delta_D(\{q_0, q_1, q_2\}, b)$$
$$= \epsilon\text{-closure}(\delta(\{q_0, q_1, q_2\}), b)$$
$$= \epsilon\text{-closure}(\{q_1\})$$
$$= \{q_1, q_2\}$$

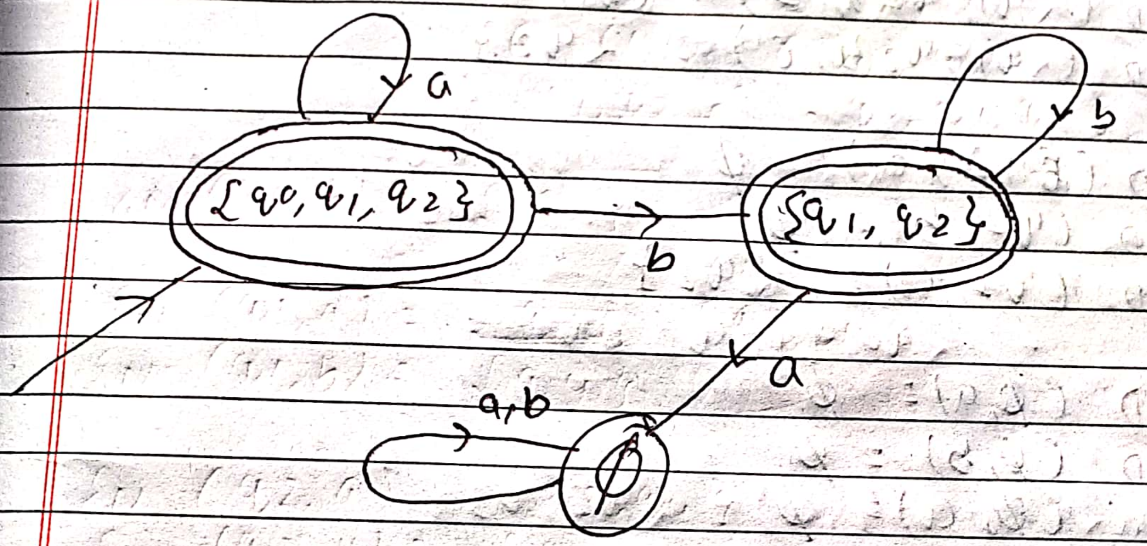
$$\delta_D(\{q_1, q_2\}, a) = \phi$$

$$\delta_D(\{q_1, q_2\}, b) = \{q_1, q_2\}$$

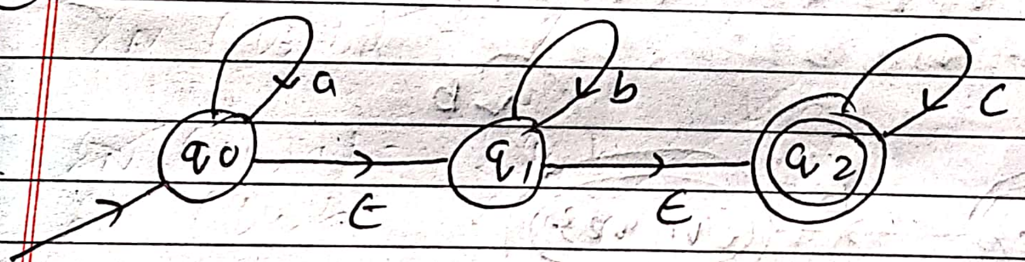
$$\delta_D(\phi, a) = \phi$$

$$\delta_D(\phi, b) = \phi$$

Constructing DFA -



(Q2)



* Initial state of ϵ-NFA is q0. So the resulting initial state of DFA will be ϵ-closure (q0)

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

Now,

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

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

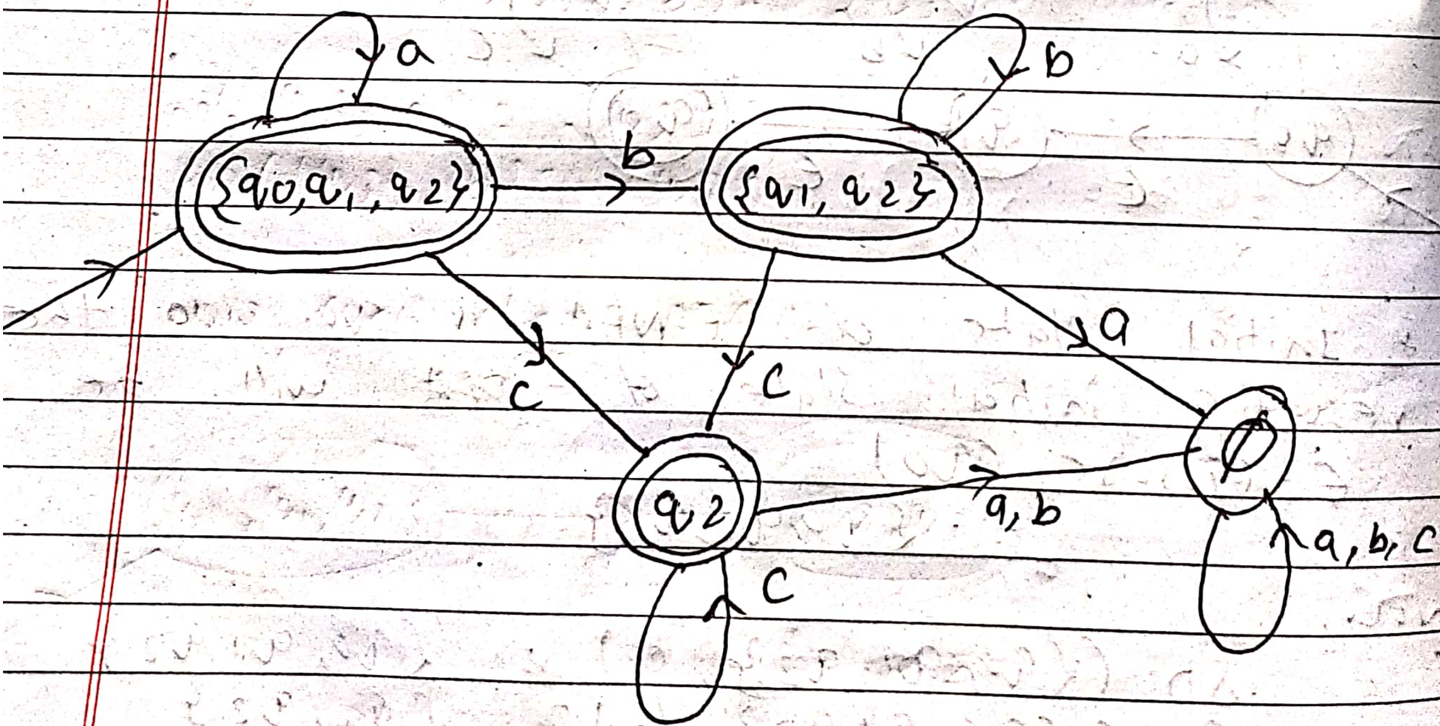
$\delta_D(\{q_0, q_1, q_2\}, c) = \{q_2\}$

$\delta_D(\{q_1, q_2\}, a) = \phi$
$\delta_D(\{q_1, q_2\}, b) = \{q_1, q_2\}$
$\delta_D(\{q_1, q_2\}, c) = \{q_2\}$

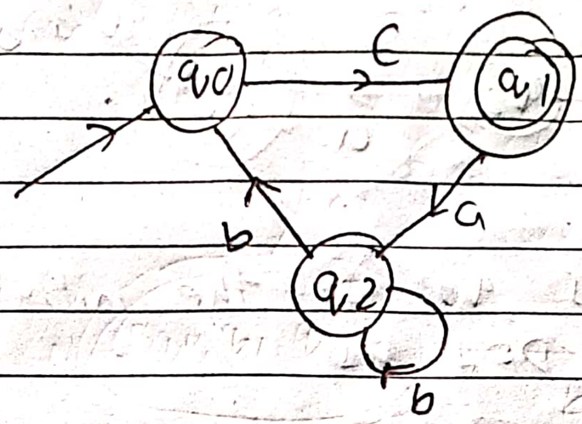
$\delta_D(\{q_2\}, a) = \phi$
$\delta_D(q_2, b) = \phi$
$\delta_D(q_2, c) = \{q_2\}$

$\delta_D(\phi, a) = \phi$
$\delta_D(\phi, b) = \phi$
$\delta_C(\phi, c) = \phi$

Now constructing DFA



Q3



Here q0 is initial state so initial state of resulting DFA is ϵ-closure (q0)

$= \{q0, q1\}$

Now,

$\delta_D(\{q0, q1\}, a) = \{q2\}$

$\delta_D(\{q0, q1\}, b) = \emptyset$

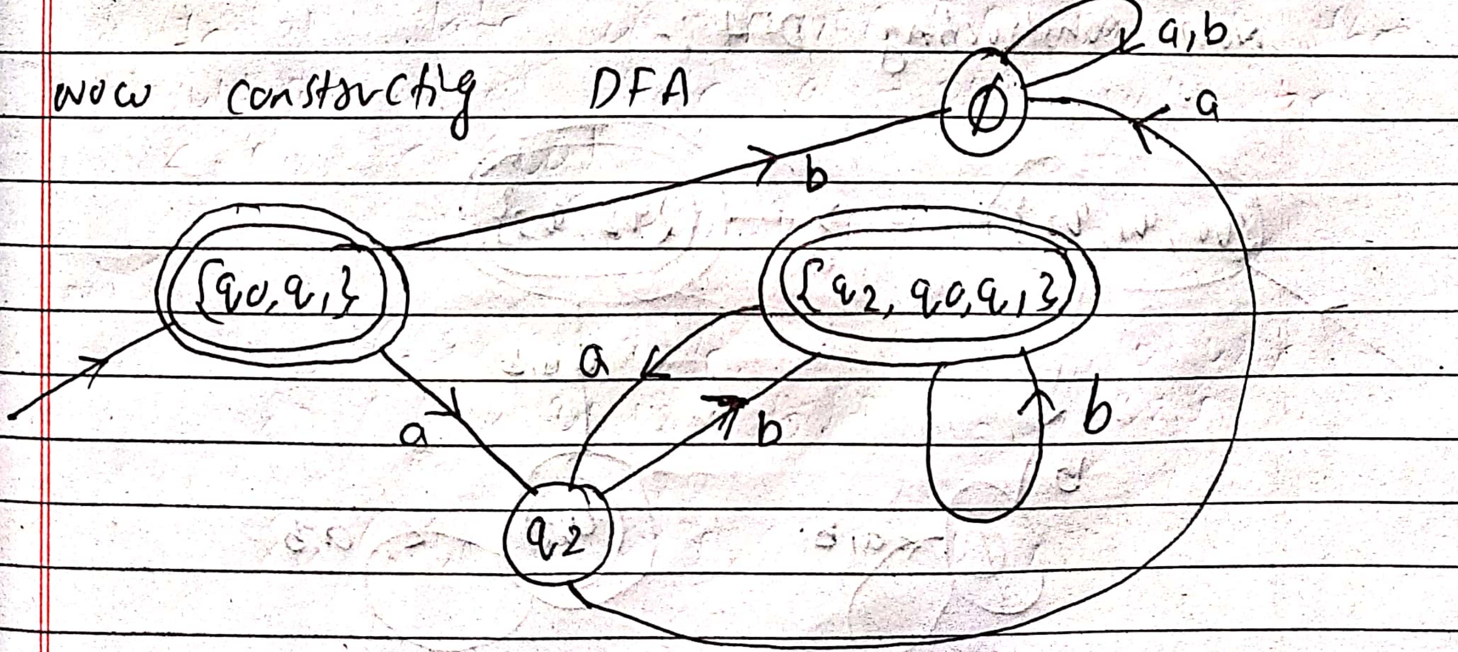
$\delta_D(\{q2\}, a) = \emptyset$

$\delta_D(q2, b) = \{q2, q0, q1\}$

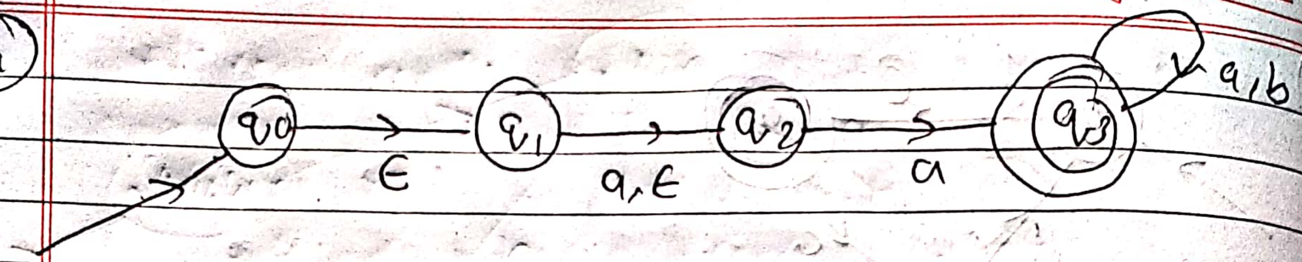
$\delta_D(\{q2, q0, q1\}, a) = \{q2\}$

$\delta_D(\{q2, q0, q1\}, b) = \{q2, q0, q1\}$

Now constructing DFA



Q4



Here initial state is q_0. So the initial state of resulting DFA is

ϵ-closure of $\{q_0\}$

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

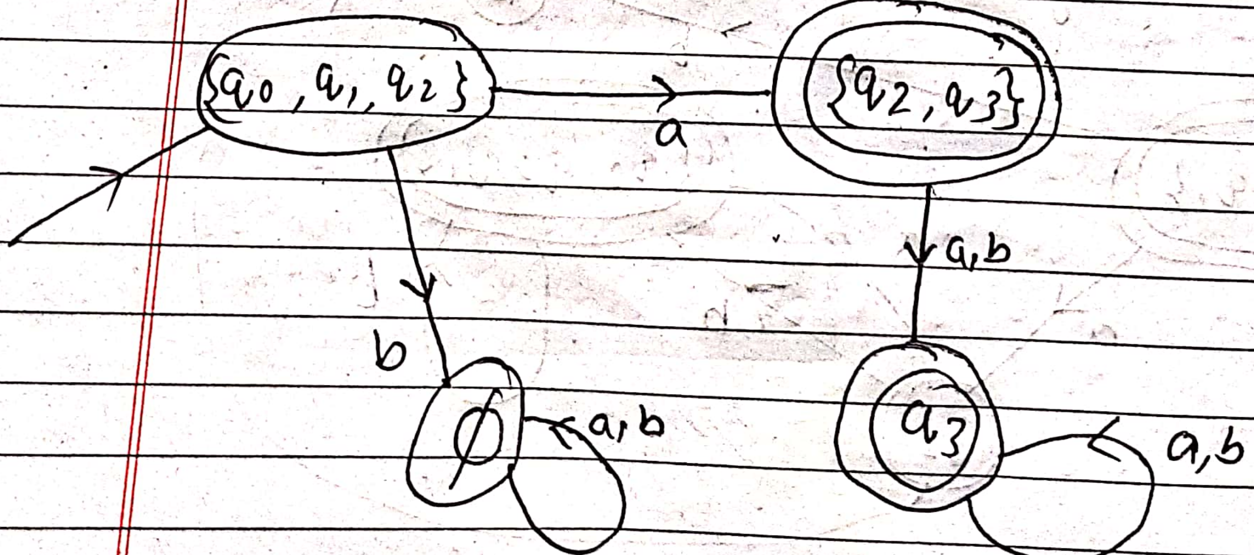
Now,

$\delta_D(\{q_0, q_1, q_2\}, a) = \{q_2, q_3\}$
$\delta_D(\{q_0, q_1, q_2\}, b) = \phi$

$\delta_D(\{q_2, q_3\}, a) = \{q_3\}$
$\delta_D(\{q_2, q_3\}, b) = \{q_3\}$

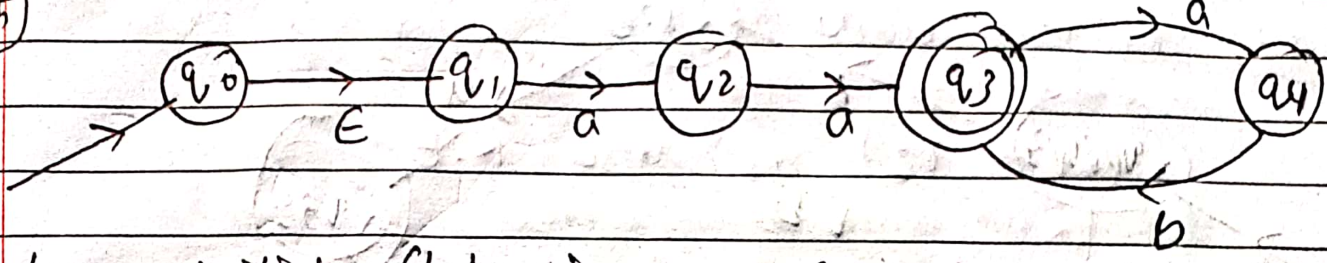
$\delta_D(\{q_3\}, a) = \{q_3\}$
$\delta_D(\{q_3\}, b) = \{q_3\}$

Now constructing DFA :-



Date
Page

Q.5



Here initial State is q0, so the initial state of resulting DFA is

ϵ - closure of $\{q_0\}$

$= \{q_0, q_1\}$

Now,

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

$\delta_D(\{q_0, q_1\}, b) = \emptyset$

$\delta_D(\{q_2\}, a) = \{q_3\}$

$\delta_D(\{q_2\}, b) = \emptyset$

$\delta_D(\{q_3\}, a) = \{q_4\}$

$\delta_D(\{q_3\}, b) = \emptyset$

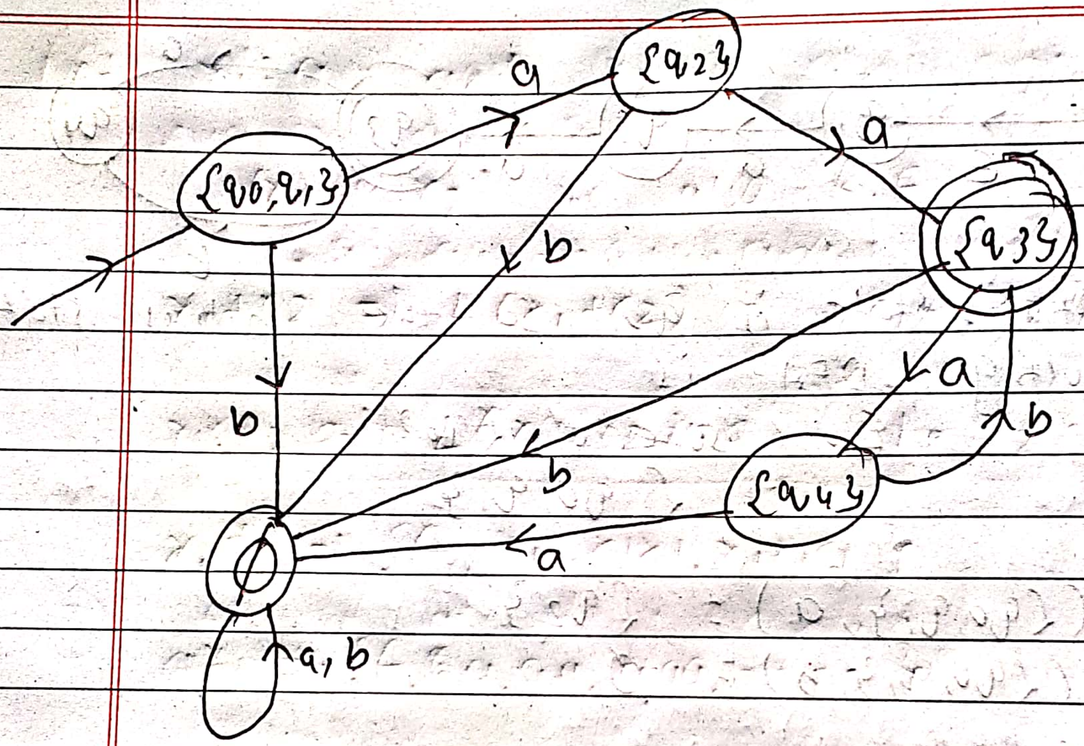
$\delta_D(\{q_4\}, a) = \emptyset$

$\delta_D(\{q_4\}, b) = \{q_3\}$

~~Now~~ $\delta_D(\emptyset, a) = \emptyset$

$\delta_D(\emptyset, b) = \emptyset$

Now constructing DFA


{q0,q1}
{q2}
{q3}
{q4}
∅
a
b
a
b
a
b
b
a
a,b