Teacher's
Remarks

* Symbols :

Some of the Symbols used in automata.

a, ---- z

A, ---- Z

0, ---- 9

@, #, ----

* Alphabets :-

* By using alphabets (or) the symbols we can define the alphabets the alphabets are denoted by symbol 'Σ'.

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

* Strings :-

* we can only form the strings by using alphabets.

Eg: $\Sigma = \{0,1\}$ = {0, 1, 00, 11, 01, 10, 000, ----}

* the strings are classified into two

1.) finite strings

2.) Infinite strings Eg: $\Sigma = \{a, b\}$

Σ = {a, b, aa, bb, ba, ab, aaa, abb, ----}

* Language :-

* Set of all strings based on some condition is given by using the input alphabet only.

EX: L1 = {all strings length of exactly} Σ = {aa, ab, ba, bb}

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

L2 = {all strings ≤ 2} $\Sigma = \{a, b\}$

Σ = {a, b, aa, ab, ba, bb}

L3 = {all strings ≥ 2}

Σ = {aa, ab, ba, bb, aaa, ----} INFL.

L4 = {all strings ending with b}

Σ = {ab, bb, ----} INL.

16/8/22

* Based on no. of inputs of the alphabets in the file, the length of string $|\Sigma| = 2^n$

Ex: n=2

Ex: $|\Sigma| = 2^2$

= 4

= {aa, ab, ba, bb}

* <u>Power of Σ</u> :

<u>Ex</u>: $\Sigma = \{a, b\}$ $\qquad \Sigma^0 = \{\epsilon\}$

$|\Sigma^1| = \{a, b\}$

$|\Sigma^2| = \Sigma \cdot \Sigma$

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

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

$|\Sigma^3| = \Sigma^2 \cdot \Sigma$

$= \{aa, ab, ba, bb\} \cdot \{a, b\}$

$= \{aaa, aab, aba, abb, baa, bab, bba, bbb\}$

$|\Sigma^n|$

$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup --- \cup \Sigma^n.$

$= \{\epsilon \cup \{a, b\} \cup \{aa, ab, ba, bb\} --- \cup \Sigma^n\}$

$= \epsilon \cup a, b, aa, ab, ba, bb, ---n$

$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup ---- \Sigma^n.$

$= a, b, aa, ab, ---.$

where, Σ^* called kleen closer

Σ^+ called positive closer.

<u>Finite State Machine</u> : (Finite Automata)

- Finite Automata
  - Finite Automata
    - Moore machine
    - Mealy machine
  - Finite Automata without output
    - Deterministic Finite Automata (DFA)
    - Non-Deterministic finite Automata (NFA)
    - NFA-ϵ

* Basic Terminologies :-

* Finite State Machine :- A finite Automata is a machine that has input tape that can be put into any of the states, various symbols are written on the tape before execution.

* The automata begins reading the symbols on the tape from left to right.

* Reading a symbol from the tape, the machine changes its state and increments the tape.

* After reading the input completely, the machine halts.

* Components of finite state machines :-

There are 3 components.

1.) control unit 2.) Read unit 3.) Input tape

* Elements of finite state machine :-

* State :- A state is a complete set of properties transmitted by an object to an observer by using one or more channels.

* Some of the states are

* Start State (or) Initial State : It is a →○ initial state of finite state machine finish a input string and it is in accepting state.

* A State immediately following the current state defined by transition function of finite state machine and the input is the next state.

* Dead State (or) trap state :- A non final state of a finite state machine.

* Final state : ◎ Accepting state (→ transition).

* Transaction :- It is an act of passing from one state to the next state.

The transactions are represented in the following ways,

* Transition diagram.
* Transition table.
* Transition function.

* Mathematical representation of finite state machine :-

* A finite state machine is described by 5 tuple values.

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

where, Q is finite no. of states.

Σ is input alphabet.

q_0 is initial state.

F is final state.

δ is transition function i.e.,

$\delta : Q \times \Sigma \rightarrow Q.$

* Chomsky's hierarchy :-

Chomsky's hierarchy	Grammer	Restrictions on productions	Acceptor	Language
TYPE 0	unrestricted	$(V \cup T)^{+} \rightarrow (V \cup T)^{*}$	TM	Recursive enumerable languages
TYPE 1	context - sensitive	$(V \cup T)^{+} \rightarrow (V \cup T)^{+}$	LBA	context sensitive language.
TYPE 2	context free	$A \rightarrow (V \cup T)^{*}$ A is single non-terminal on left hand side of production.	PDA.	context free
TYPE 3	Regular	$A \rightarrow t$ (or) $A \rightarrow tB$ (or) $A \rightarrow t$ where, $A, B \in V$ & $t \in T$	FA	Regular language.

TYPE - ③ :- It must have a single non-terminal on the left hand side and right hand side consists of the single terminal (or) single terminal followed by single non terminal.

TYPE - ② : only a single non terminal on the left hand side and any number of terminals and non-terminals on right hand side production.

* The grammer is represented by
G: It consists of 4 elements i.e.,

$G = (V, T, P, S)$

where, V - variables (or) non-terminals
T - Terminals (except capital letters)
P - productions
S - Start symbol

* variables are represented by capital letters.

* Finite Automata without output :-

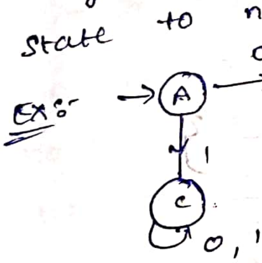
1) DFA 2) NFA 3.) NFA - ϵ

* DFA (Deterministic finite Automata) :-

* A finite Automata is called DFA if there is only one path for specific input value from current state to next state.

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

EX :-



Transition diagram

δ	0	1
A	B	C
B	B	B
C	C	C

Transition table

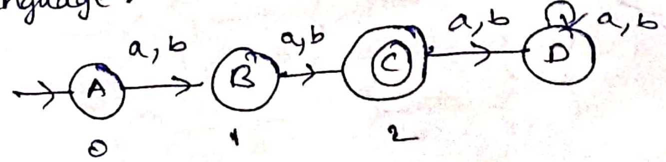
$\delta(A, 0) = B$
$\delta(A, 1) = C$
$\delta(B, 0) = B$
$\delta(B, 1) = B$
$\delta(C, 0) = C$
$\delta(C, 1) = C$.

Transition function.

* Design of DFA :-

* Construct DFA for set of all strings over $\Sigma = \{a, b\}$ where all strings are having length 2.

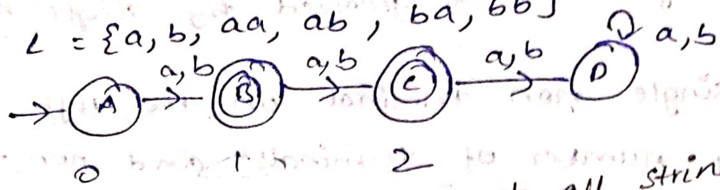
Language :- $L = \{aa, ab, ba, bb\}$



'D' is the dead state.

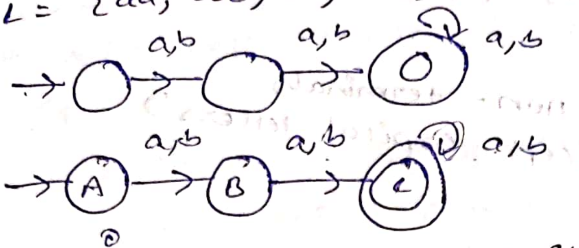
* construct DFA where set of all strings over $\Sigma = \{a, b\}$
all strings having length ≤ 2.

Sol $L = \{a, b, aa, ab, ba, bb\}$



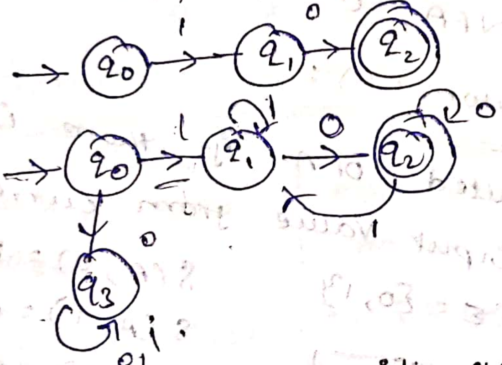
* Construct DFA which accept all strings over $\Sigma = \{a, b\}$
all strings having length ≥ 2.

$L = \{aa, ab, ba, bb, aaa, aba, \text{------}\}$.



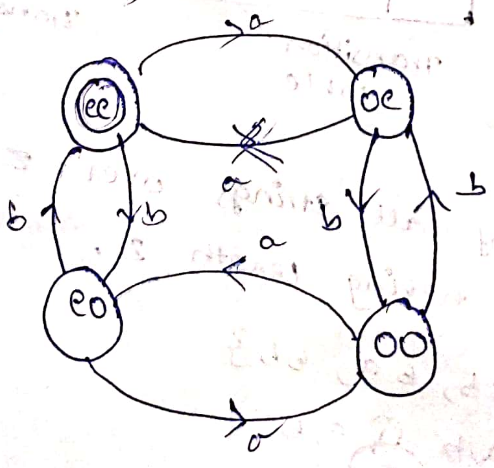
* Construct DFA set of all strings over $\Sigma = \{0, 1\}$ where
all strings are starting with 1 and ending with 0.

Sol $L = \{10, 110, 100, 1110, 1000, 1010, \text{------}\}$



* Design a DFA which accept even no. of 'a' s and
even no. of 'b' s where, $\Sigma = \{a, b\}$.

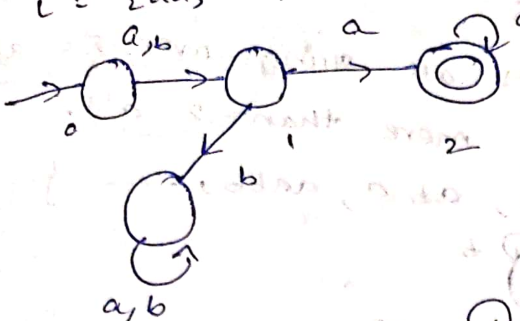
$L = \{aa, bb, aaaa, aaaaaa, bbbb, \text{-----}$
$\quad aabb, abba, abab, \text{------}\}$

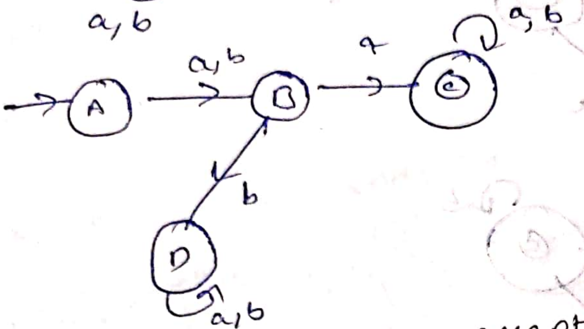


* Design a DFA for set of all strings over $\Sigma = \{a, b\}$ if accept all strings which having 2^{nd} symbol from L.H.S is 'a'.

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

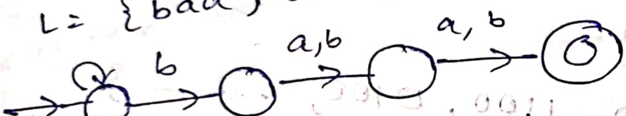
$L = \{aa, ba, aaa, baa, bab, aaaa, aab, aaab, aabb, ----\}$

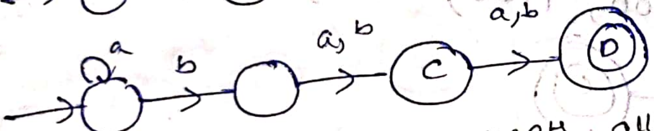




* Design a DFA which accept all strings having the third symbol from RHS is 'b' where $\Sigma = \{a, b\}$.

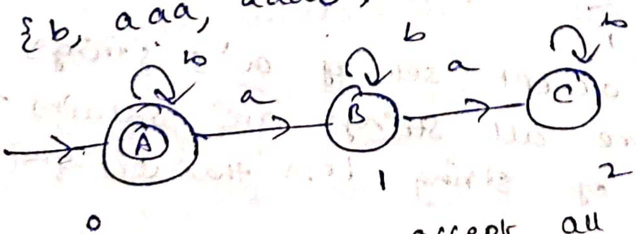
$L = \{baa, bab, bbb, bba, abba, ----\}$





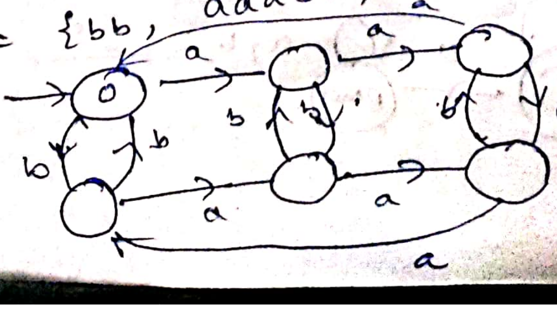
* construct a DFA which accepts all strings over $\Sigma = \{a, b\}$ no of 'a's are divisible by 3 (or) no. of 'a's % 3 = 0.

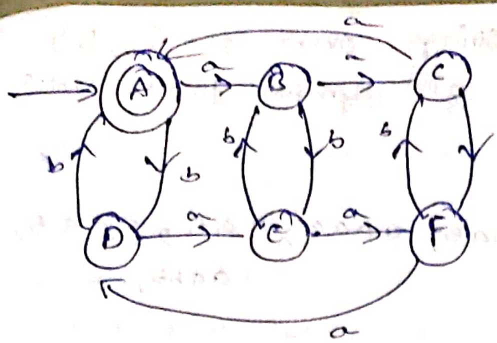
$L = \{b, aaa, aaab, aaabb, abaa, ----\}$.



* Construct DFA which accept all strings $\Sigma = \{A, B\}$ where no. of 'a's % 3 = 0, no. of 'b's % 2 = 0.

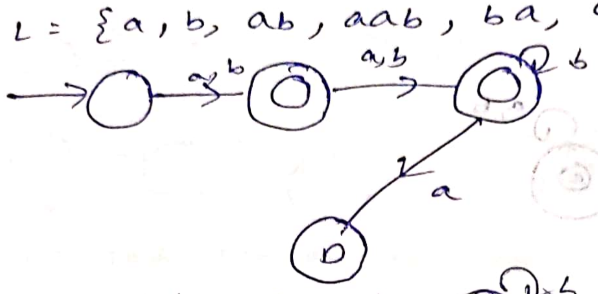
$L = \{bb, aaabb, abaab, ----\}$

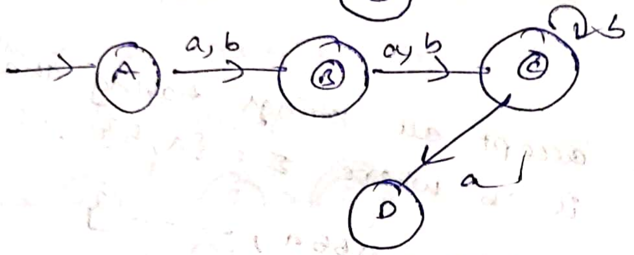




* Design a DFA which accepts all strings over $\Sigma = \{a,b\}$ where all strings not having more than 2 a's.

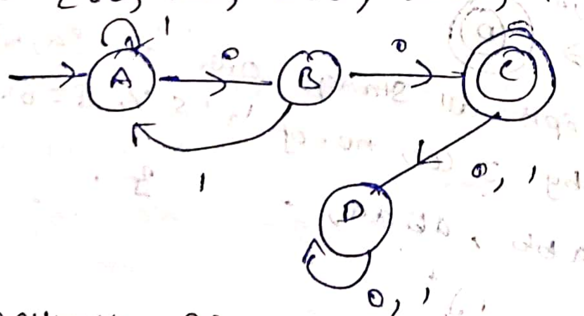
$L = \{a, b, ab, aab, ba, aba, aabb, \ldots\}$





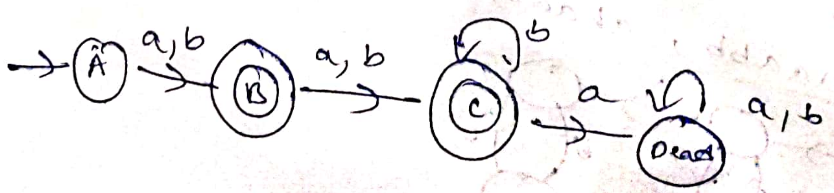
* Construct DFA that accept the strings that always ends with 00 where $\Sigma = \{0,1\}$

$L = \{00, 100, 000, 1000, 1100, 0100, \ldots\}$



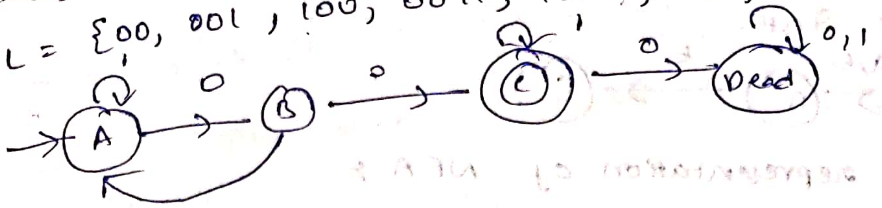
* Construct DFA that accept set of a's strings over $\Sigma = \{a,b\}$ where all strings are contains no. of a's $|w|$ - length of string less than (or) equal to 2.

$L = \{a, b, ab, aa, ba, bb, aab, abb, bbb, \ldots\}$



* Design a DFA which accepts set of all strings over $\Sigma = \{0,1\}$ the strings contains only two consecutive zeroes.

$L = \{00, 001, 100, 0011, 1001, 1100, \ldots\}$



condition.

NOTE:

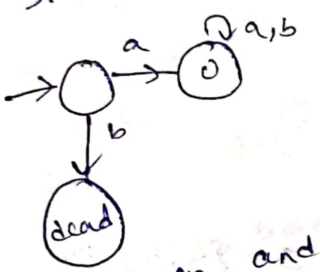
$|w| = n$ — $n+2$

$|w| \leq n$ — $n+2$

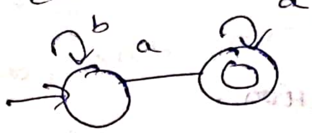
$|w| \geq n$ — $n+1$

minimum no. of states to construct finite automata (or) DFA:

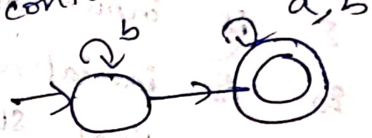
starts 'a'



ends 'a'

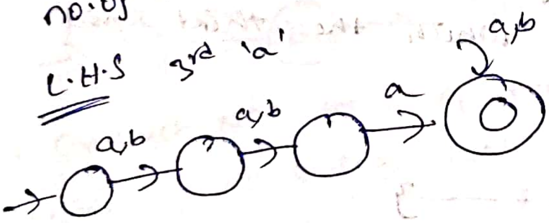


contain a

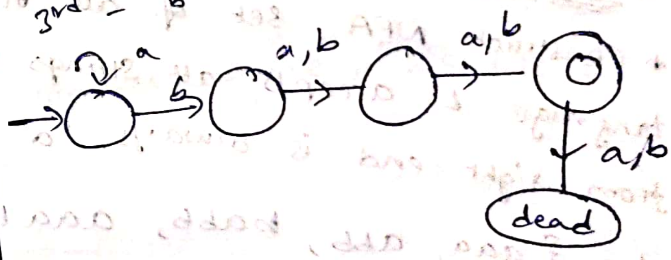


no. of even and odd - standard diagram.

L.H.S 3rd 'a'



R.H.S 3rd - 'b'

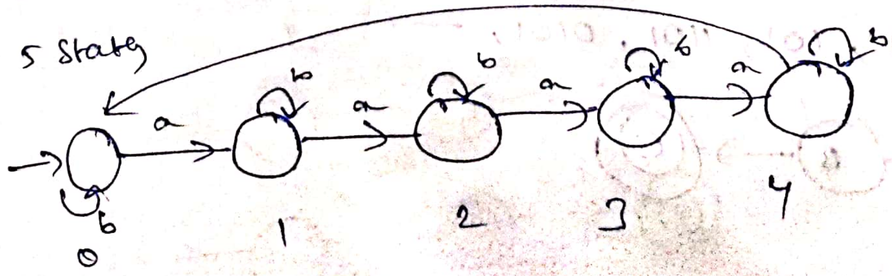


mod $3^n = 0$.

No. of remainder values $= 3(0,1,2) \Rightarrow 3$ states.

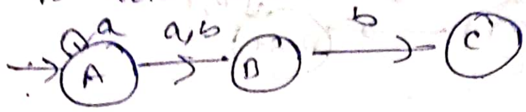
mod 5 = 0 'a's a

5 State



* NFA (Non-deterministic finite Automata):-

* A finite automata is called NFA there exists many paths or transitions for a specific input from current state to next state.

Ex:- 

* Mathematical representation of NFA :-
* There are 5 tuple values.

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

where,

Q is finite no. of states

Σ is input alphabet.

q_0 is initial state

F is final state.

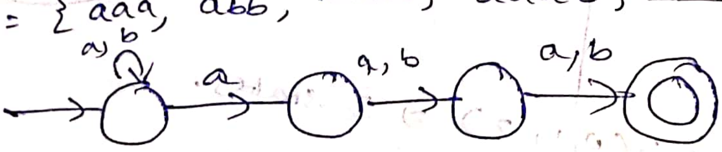
δ is transition function.

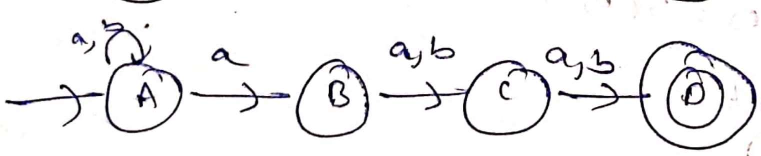
i.e., $\delta = Q \times \Sigma \rightarrow 2^Q$.

* Design of NFA :-

* Construct NFA set of all integers over $\Sigma = \{a, b\}$ the language L accepts all strings in which the third symbol from right end is always 'a'.

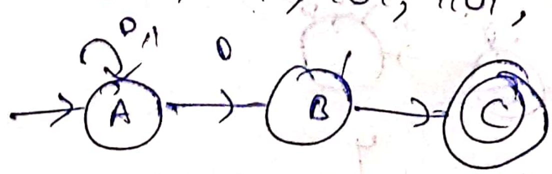
$L = \{aaa, abb, babb, aaab, \text{——}\}$





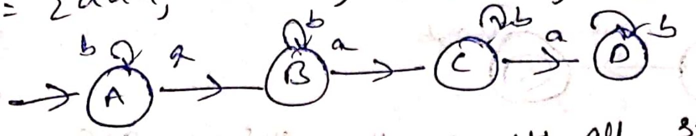
* Construct NFA that accepts all strings are ending with 01 where $\Sigma = \{0, 1\}$.

$L = \{01, 001, 101, 1101, 0101, \text{——}\}$



* Construct NFA that accept all strings are contains only 3 'a' s. $\Sigma = \{a, b\}$

$L = \{aaa, aaab, babaa, bbaaab, \text{------}\}$

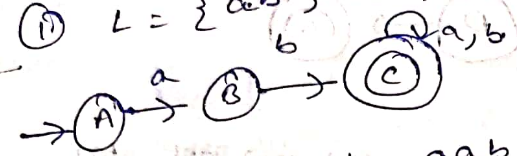


* construct NFA that accepts all strings

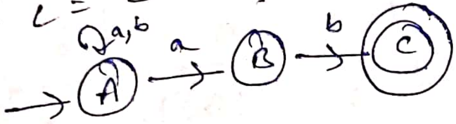
$\Sigma = \{a, b\}$

(1) strings starts with 'ab'.
(2) All strings ends with 'ab'.
(3) All strings contains 'ab'.

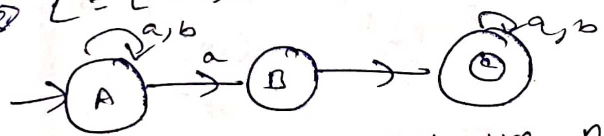
Sol (1) $L = \{ab, aba, abb, abba, \text{----}\}$



(2) $L = \{ab, bab, aab, aaab, baab, \text{---}\}$



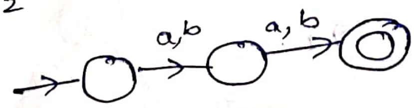
(3) $L = \{ab, bab, aabb, aabbba, \text{------}\}$

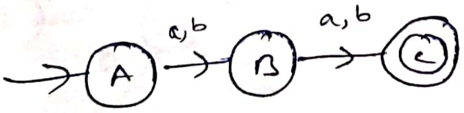


* Construct NFA for set of +ve numbers where, $\Sigma = \{a, b\}$

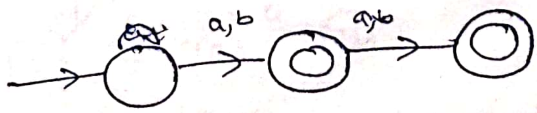
i, set of all strings $|w| = 2$
ii, $|w| \leq 2$.
iii, $|w| \geq 2$

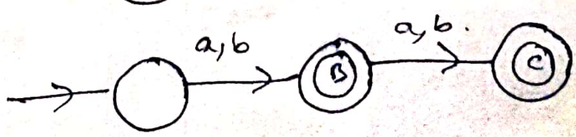
i, $|w| = 2 \Rightarrow L = \{ab, aa, ba, bb\}$



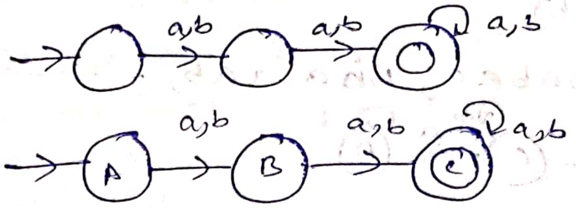


ii, $|w| \leq 2$ $L = \{a, b, aa, ab, ba, bb\}$



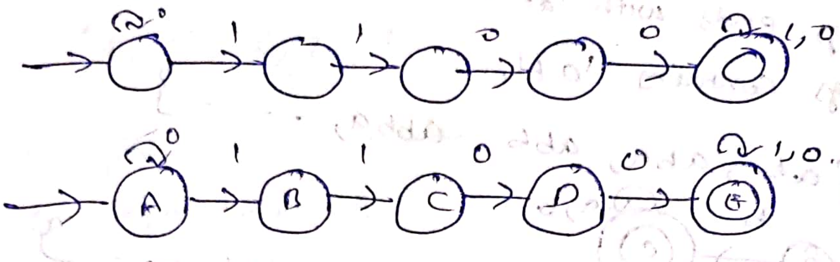


iii) $|w| \geq 2$ $L = \{aa, ab, bb, ba, aaa, \ldots\}$



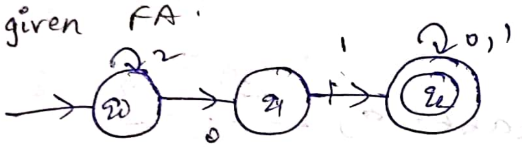
6.) Design a NFA that accepts all the strings are, considered 11 followed by 00 $\Sigma = \{0, 1\}$.

$L = \{1100, 01100, 11000, 110010, 101100\}$



String Acceptance :-

* Check the following strings are accepted (or) not for the given FA.



a) 0100111
b) 010101010
c.) 01001
d.) 011000

Transition Table :-

δ'	0	1
q_0	q_1	q_0
q_1	ϕ	q_2
q_2	q_2	q_2

i, $(q_0, 0100111)$
$\Rightarrow (q_1, 100111)$
$\Rightarrow (q_2, 00111)$
$\Rightarrow (q_2, 0111)$
$\Rightarrow (q_2, 111)$
$\Rightarrow (q_2, 11)$
$\Rightarrow (q_2, 1)$
$\Rightarrow (q_2, \epsilon)$
Accepted.

i, 010
$(q_0, 010$
$\Rightarrow (q_1,$
$\Rightarrow (q_2)$
$\Rightarrow (q_2,$
$\Rightarrow (q_2)$
$\Rightarrow (q_2)$
$\Rightarrow (q_2,$
$\Rightarrow (q_2)$
$\Rightarrow (q_2 /$
$\Rightarrow (q_2$
Accepted
2.) check
given FA

δ
q_0
q_1
q_2
q_3
q_4

Sol i, 0100

Accepted

i, 010101010

$(q_0, 010101010)$

$\Rightarrow (q_1, 10101010)$

$\Rightarrow (q_2, 0101010)$

$\Rightarrow (q_2, 101010)$

$\Rightarrow (q_2, 01010)$

$\Rightarrow (q_2, 1010)$

$\Rightarrow (q_2, 010)$

$\Rightarrow (q_2, 10)$

$\Rightarrow (q_2, 0)$

$\Rightarrow (q_2, \epsilon)$

Accepted.

iii, 01001

$\Rightarrow (q_0, 01001)$

$\Rightarrow (q_1, 1001)$

$\Rightarrow (q_2, 001)$

$\Rightarrow (q_2, 01)$

$\Rightarrow (q_2, 1)$

$\Rightarrow (q_2, \epsilon)$

Accepted.

iv, 011000

$(q_0, 011000)$

$\Rightarrow (q_1, 11000)$

$\Rightarrow (q_2, 1000)$

$\Rightarrow (q_2, 000)$

$\Rightarrow (q_2, 00)$

$\Rightarrow (q_2, 0)$

$\Rightarrow (q_2, \epsilon)$

Accepted.

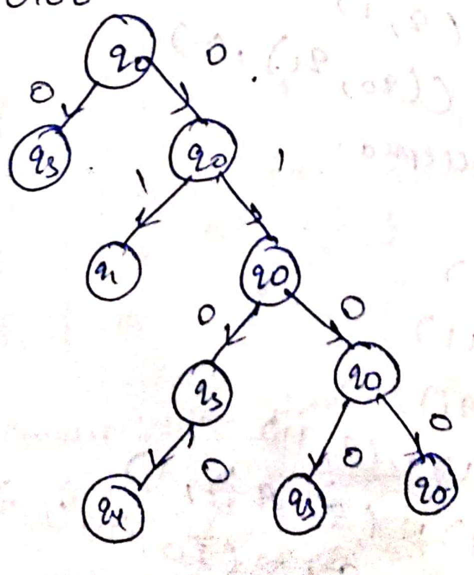
2) check the following strings are accepted or not for the given FA transition table.

δ	0	1
q_0	$\{q_0, q_3\}$	$\{q_0, q_1\}$
q_1	ϕ	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$
q_3	$\{q_4\}$	ϕ
q_4	$\{q_4\}$	$\{q_4\}$

i, 0100 ii, 1011.

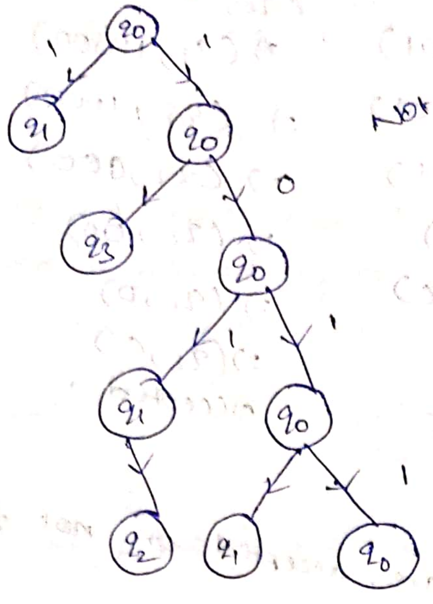
89913000904

Sol: i, 0100



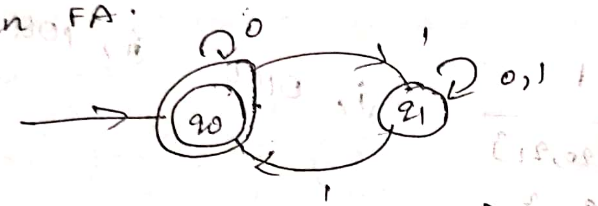
Accepted

ii. 1011



Not accepted.

* check the following strings are accepted or not for the given FA:



1.) 101   2.) 1001   3.) 0101   4.) 1111

1.) 101

$(q_0, 101)$

$\Rightarrow (q_1, 01)$

$\Rightarrow (q, 1)$

$\Rightarrow ((q_0, q_1), \epsilon)$

$\Rightarrow ((q_0, \epsilon), (q_1, \epsilon))$

Accepted.

2.) 1001

$(q_0, 1001)$

$\Rightarrow (q_1, 001)$

$\Rightarrow (q, 01)$

$\Rightarrow (q, 1)$

$\Rightarrow ((q_0, q_1), \epsilon)$

Accepted.

3.) 0101

$(q_0, 0101)$

$\Rightarrow (q_0, 101)$

$\Rightarrow (q_1, 01)$

$\Rightarrow (q_1, 1)$

$\Rightarrow ((q_0, q_1), \epsilon)$

$(q_0, \epsilon) \cup (q_1, \epsilon)$

Accepted.

4.) 1111

$(q_0, 1111)$

$(q_1, 111)$

$((q_0, q_1), 11)$

$(q_0, 11) \cup (q_1, 11)$

$(q_1, 1) \cup ((q_0, q_1), 1)$

$((q_0, q_1), \epsilon) \cup (q_0, 1) \cup (q_1, 1)$

$q_1 \cup (q_0, \epsilon) \cup (q_1, \epsilon)$

Accepted.

* NFA

without

1 state

Eg:

* Conve

transiti

Formula

Transiti

Transition

δ	0
→ q_0	{q_0
→ q_0q_1	{q_0
* q_1	φ

2.) Convert

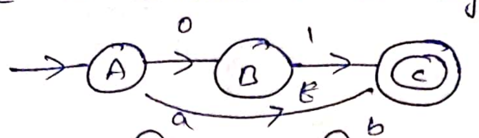
table is

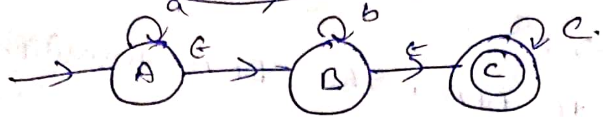
* NFA with ϵ transition or NFA with ϵ moves:

without reading any input symbols we can move or jump 1 state to another state by using NFA ϵ transition.

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

Eg:





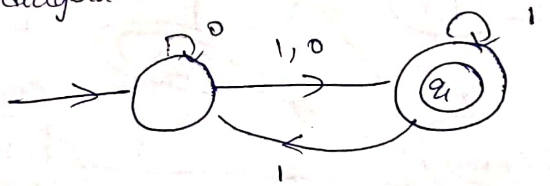
* Conversion of NFA ϵ transition to NFA without ϵ transition.

Formulae :
$$\hat{\delta}(q_0, \epsilon) = \epsilon\text{-closure}(q_0)$$
$$\hat{\delta}(q_0, 1) = \{q_1\}$$
$$\delta(q_1, 0) = \phi$$
$$\delta(q_1, 1) = \{q_0, q_1\}$$

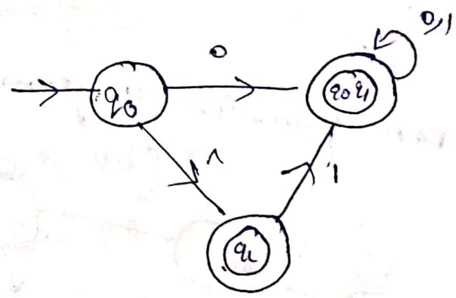
Transition table :

δ	0	1
q_0	$\{q_0, q_1\}$	$\{q_1\}$
q_1	ϕ	$\{q_0, q_1\}$

Transition diagram



δ	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_1\}$
$\rightarrow q_0q_1$	$\{q_0, q_1\}$	$\{q_0 q_1\}$
$* q_1$	ϕ	$\{q_0, q_1\}$



2.) Convert the given NFA to DFA. the NFA transition table is

δ	0	1
p	$\{p, q\}$	$\{p\}$
q	$\{r\}$	$\{r\}$
r	$\{s\}$	–
s	$\{s\}$	$\{s\}$

Given data, the given NFA, $M = \{Q, \Sigma, \delta, q_0, F\}$

where, $Q = \{p, q, r, s\}$

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

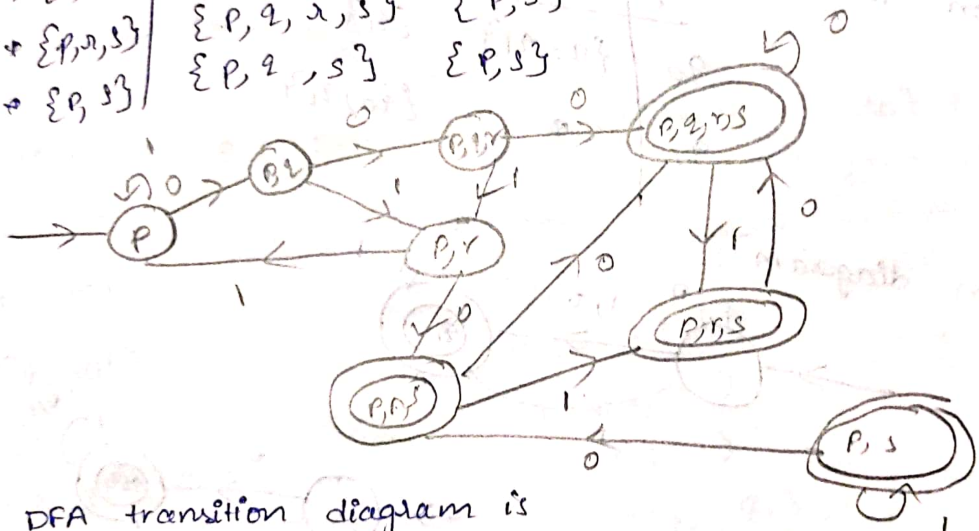
$q_0 = p$

$F = s$

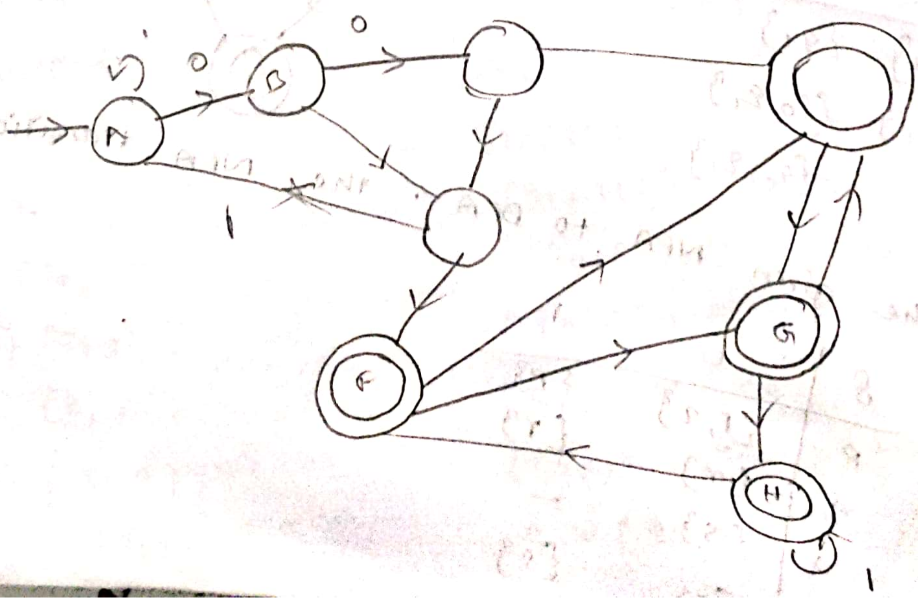
$\delta : Q \times \Sigma \rightarrow 2^Q$

Now, converting NFA to DFA, the DFA transition table

	δ	0	1
A	p	{p, q}	{p}
B	{p, q}	{p, q, r}	{p, r}
C	{p, q, r}	{p, q, r, s}	{p, r}
D	{p, r}	{p, q, s}	{p}
E	* {p, q, r, s}	{p, q, r, s}	{p, r, s}
F	* {p, q, s}	{p, q, r, s}	{p, r, s}
G	* {p, r, s}	{p, q, r, s}	{p, s}
H	* {p, s}	{p, q, s}	{p, s}



The DFA transition diagram is



Q. Convert ... diagram i...

Given dat...
the NFA
M = {Q,
Q = {q...
Σ = {0,
q_0 = q_0
F = {q...
NFA
Transition

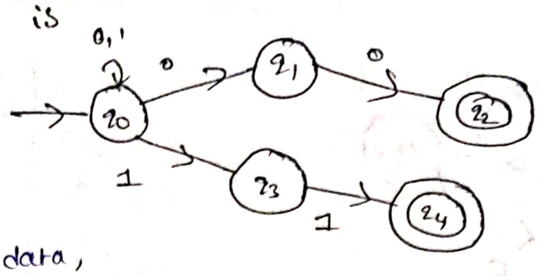
δ
q_0
q_1
q_2
q_3
q_4

DFA transi...

	δ
A.	
B	[q_0 q...
C	[q_0 q...
* D.	[q_0 q...
* E	[q_0 q_3...

DFA transitio...

Q Convert the given NFA to DFA. The NFA transition diagram is



Given data,

the NFA,

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

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

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

$q_0 = q_0$

$F = \{q_2, q_4\}$

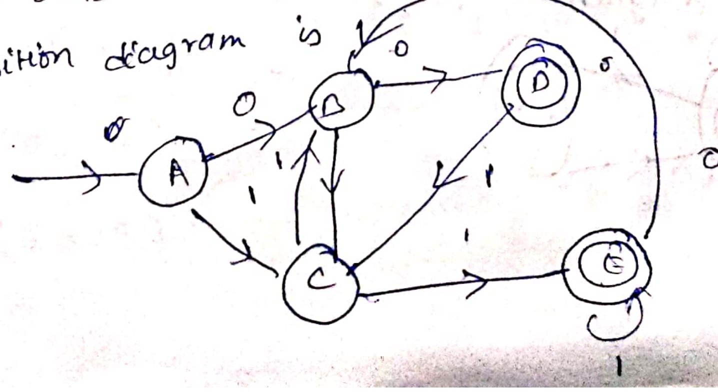
NFA Transition table is

δ	0	1
q_0	$\{q_0, q_1\}$	$\{q_0, q_3\}$
q_1	$\{q_2\}$	ϕ
q_2	$\{\phi$	ϕ
q_3	ϕ	q_4
q_4	ϕ	ϕ

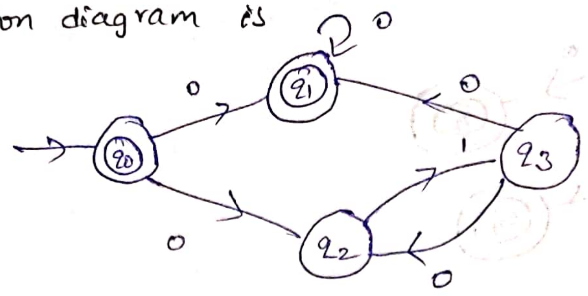
DFA transition table is

	δ	0	1
A.	q_0	$[q_0 q_1]$ B	$[q_0 q_3]$ C
B	$[q_0 q_1]$	$[q_0 q_1 q_2]$ D	$[q_0 q_3]$ C
C	$[q_0 q_3]$	$[q_0 q_1]$ B	$[q_0 q_3 q_4]$ E
* D.	$[q_0 q_1 q_2]$	$[q_0 q_1 q_2]$ D	$[q_0 q_3]$ C
* E	$[q_0 q_3 q_4]$	$[q_0 q_1]$ B	$[q_0 q_3 q_4]$ E

DFA transition diagram is



To convert the following NFA to DFA. The NFA transition diagram is



NFA transition table is

	0	1
$\rightarrow *\ q_0$	$[q_1, q_2]$	-
$*\ q_1$	$[q_1, q_2]$	-
q_2	-	$[q_1, q_3]$
q_3	$[q_2\ q_1]$	-

Given data,

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

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

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

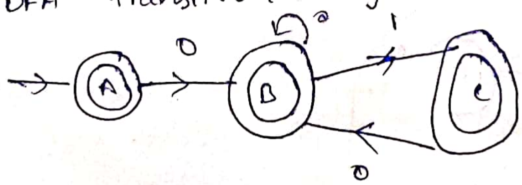
$q_0 = q_0$

$F = q_0, q_1.$

The DFA transition table is

		0	1
A *	q_0	$[q_1, q_2]$	-
B *	$[q_1 q_2]$	$[q_1, q_2]$	$[q_1, q_3]$
C *	$[q_1, q_3]$	$[q_1, q_2]$	-

DFA transition diagram:



Construct DFA for the given NFA transaction table. The NFA transition table is

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

Given data is,

The NFA is,

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

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

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

$q_0 = q_0$

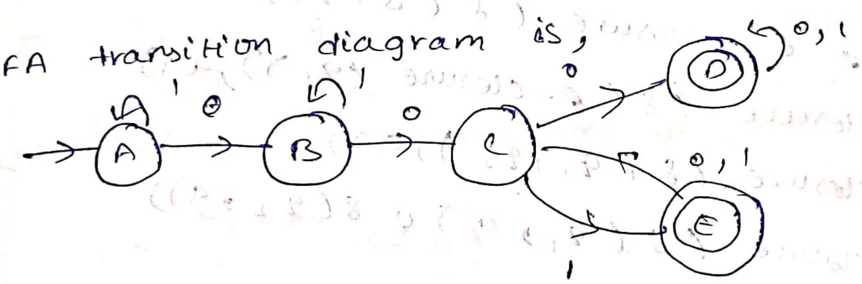
$F = q_3$

$\delta = Q \times \Sigma \rightarrow Q^2$.

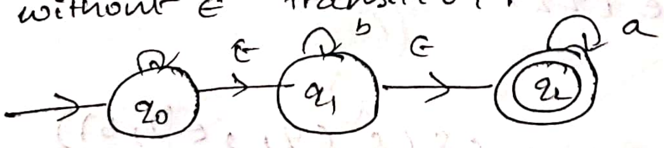
DFA transition table is

δ	0	1
$\rightarrow A\ q_0$	$[q_0 q_1]$	$[q_0]$
$B\ [q_0 q_1]$	$[q_0 q_1 q_2]$	$[q_0 q_1]$
$C\ [q_0 q_1 q_2]$	$[q_0 q_1 q_2 q_3]$	$[q_0 q_1 q_3]$
$* D\ [q_0 q_1 q_2 q_3]$	$[q_0 q_1 q_2 q_3]$	$[q_0 q_1 q_3 q_2]$
$* E\ [q_0 q_1 q_3]$	$[q_0 q_1 q_2]$	$[q_0 q_1 q_2]$

DFA transition diagram is,



* Convert the following NFA ϵ transition to NFA without ϵ transition.



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

ϵ- closure $(q_1) = \{q_1, q_2\}$

ϵ- closure $(q_2) = \{q_2\}$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$= \epsilon\text{-closure}\,(\phi \cup q_1 \cup \phi)$

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

iii, $\delta(q_1, a) = \epsilon\text{-closure}\,(\delta(\hat{\delta}(q_1, \epsilon)), a))$

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

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

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

$= \epsilon\text{-closure}\,(\phi \cup q_2)$

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

$\delta(q_1, a) = q_2$.

iv, $\delta(q_1, b) = \epsilon\text{-closure}\,(\delta(\hat{\delta}(q_1, \epsilon), b))$

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

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

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

$= \epsilon\text{-closure}\,(q_1 \cup \phi)$

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

$\delta(q_1, b) = q_1, q_2$.

v, $\delta(q_2,$

$\delta(q_2,$

vi, $\delta(q_2$

$\delta(q$

Transition

→ q_0

q_1

* q_2

NFA wi

Convert th

NFA - N

v, $\delta(q_2, a) = \epsilon\text{-closure}(\delta(\hat{\delta}(q_2, \epsilon), a))$

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

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

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

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

vi, $\delta(q_2, b) = \epsilon\text{-closure}(\delta(\hat{\delta}(q_2, \epsilon), b))$

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

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

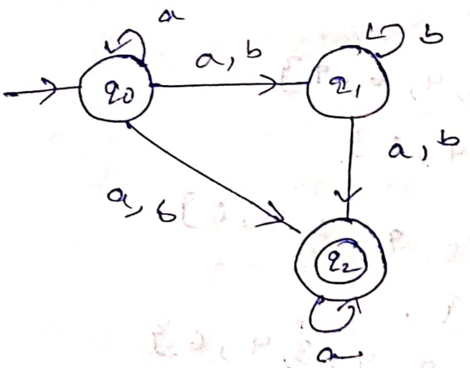
$= \epsilon\text{-closure}(\phi)$

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

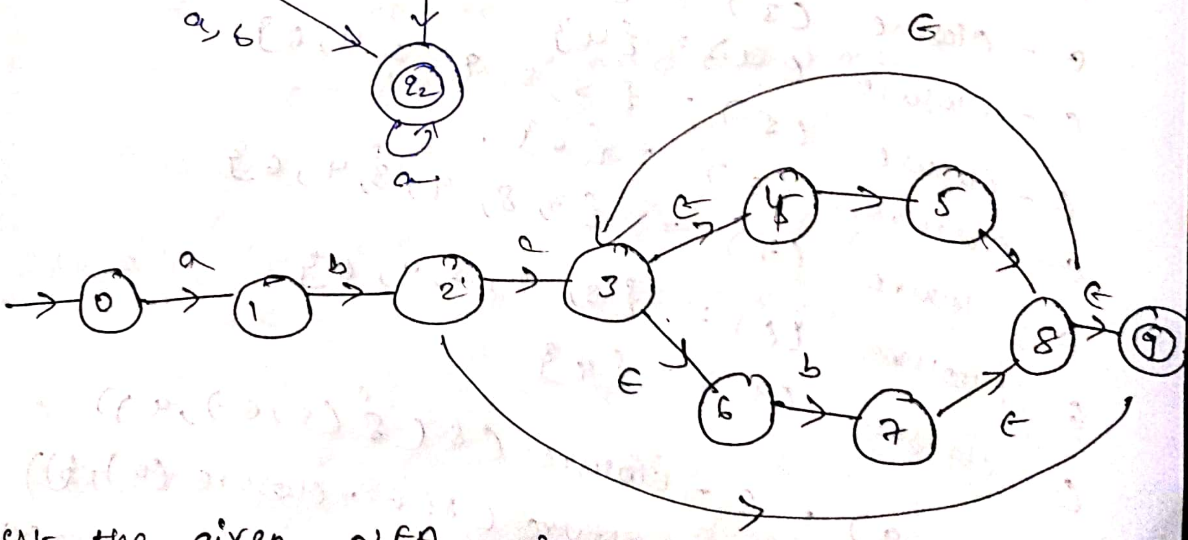
Transition table :

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

NFA without ϵ - transition diagram :



*



Convert the given NFA with ϵ-transition to NFA-without ϵ-transition.

The NFA ϵ - transition diagram is given above.

Sol. Given data,

NFA - ϵ transition given as

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

where,

$Q = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

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

$q_0 = \{0\}$

$F = \{9\}$

$\delta = Q \times \Sigma \rightarrow Q^2$.

Now converting NFA - ϵ transition to NFA without transition.

1st we findout ϵ - closures for each and every state and findout the new transition function for each and every state by using given input alphabets.

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

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

ϵ - closure $(0) = \{0\}$

ϵ - closure $(1) = \{1\}$

ϵ - closure $(2) = \{2, 3, 4, 6, 7\}$

ϵ - closure $(3) = \{3, 4, 6\}$

ϵ - closure $(4) = \{4\}$

ϵ - closure $(5) = \{5, 8, 9, 3, 4, 6\}$

ϵ - closure $(6) = \{6\}$.

ϵ - closure $(7) = \{7, 8, 9, 3, 4, 6\}$

ϵ - closure $(8) = \{8, 9, 3, 4, 6\}$

ϵ - closure $(9) = \{9\}$

i, $\delta(0, a) = \epsilon\text{-closure}(\delta(\hat{\delta}(0, \epsilon), a))$

$= \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(0), a))$

$= \epsilon\text{-closure}(\delta(0, a)) \cancel{\epsilon\text{-closure}(a)}$

$\cancel{= \phi}$ ϵ-closure $(\phi) = \{1\}$

ii, $\delta(0,$

iii, $\delta(1$

iv, $\delta($

v, $\delta(2$

vi, $\delta(2$

vii,

vi

ithout

state

and

ii, $\delta(0,b) = \epsilon\text{-closure}(\delta(\hat{\delta}(0,\epsilon),b))$

$= \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(0),b))$

$= \epsilon\text{-closure}(\delta(0,b)) = \epsilon\text{-closure}(\phi)$

$= \phi$

iii, $\delta(1,a) = \epsilon\text{-closure}(\delta(\hat{\delta}(1,\epsilon),a))$

$= \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(1),a))$

$= \epsilon\text{-closure}(\delta(1,a))$

$= \epsilon\text{-closure}(\phi) = \phi.$

iv, $\delta(1,b) = \epsilon\text{-closure}(\delta(\hat{\delta}(1,\epsilon),b))$

$= \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(1),b))$

$= \epsilon\text{-closure}(\delta(1,b))$

$= \epsilon\text{-closure}(2)$

$= \{2,3,4,6,9\}$

v, $\delta(2,a) = \epsilon\text{-closure}(\delta(\hat{\delta}(2,\epsilon),a))$

$= \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(2),a))$

$= \epsilon\text{-closure}(\delta(2,3,4,6,9),a))$

$= \epsilon\text{-closure}(\phi \cup \phi \cup 5 \cup \phi \cup \phi)$

$= \epsilon\text{-closure}(5) = \{5,8,9,3,4,6\}$

$\Rightarrow \epsilon\text{-closure}$

vi, $\delta(2,b) = \epsilon\text{-closure}(\delta(\epsilon\text{-closure}(2),b))$

$= \epsilon\text{-closure}(\delta(2,3,4,6,9),b))$

$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup 7 \cup \phi)$

$= \{7,8,9,3,4,6\}$

vii, $\delta(3,a) = \epsilon\text{-closure}(\delta(3,4,6),a)$

$= \epsilon\text{-closure}(\phi \cup 5 \cup \phi)$

$= \{5,8,9,3,4,6\}$

viii, $\delta(3,b) = \epsilon\text{-closure}(\delta(3,4,6),b))$

$= \epsilon\text{-closure}(\phi \cup \phi \cup 7)$

$= \{7,8,9,3,4,6\}.$

ix) $\delta(4,a) = \epsilon\text{-closure}(\delta(4), a))$
$= \epsilon\text{-closure}(5)$
$= \{5, 8, 9, 3, 4, 6\}$

x) $\delta(4,b) = \epsilon\text{-closure}(\delta(4), b))$
$= \phi$.

xi) $\delta(5,a) = \epsilon\text{-closure}(\delta(5, 8, 9, 3, 4, 6), a))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup 5 \cup \phi)$
$= \{5, 8, 9, 3, 4, 6\}$

xii) $\delta(5,b) = \epsilon\text{-closure}(\delta(5, 8, 9, 3, 4, 6), b))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup \phi \cup 7)$
$= \{7, 8, 9, 3, 4, 6\}$

xiii) $\delta(6,a) = \epsilon\text{-closure}(\delta(6), a))$
$= \phi$.

xiv) $\delta(6,b) = \epsilon\text{-closure}(\delta(6), b))$
$= \epsilon\text{-closure}(7)$
$= \{7, 8, 9, 3, 4, 6\}$

xv) $\delta(7,a) = \epsilon\text{-closure}(\delta(7, 8, 9, 3, 4, 6), a))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup 5 \cup \phi)$
$= \{\cancel{7}, 8, 9, 3, 4, 6\}$

xvi) $\delta(7,b) = \epsilon\text{-closure}(\delta(7, 8, 9, 3, 4, 6), b))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup \phi \cup 7)$
$= \{7, 8, 9, 3, 4, 6\}$

xvii) $\delta(8,a) = \epsilon\text{-closure}(\delta(8, 9, 3, 4, 6), a))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup 5 \cup \phi)$
$= \{5, 8, 9, 3, 4, 6\}$

xviii) $\delta(8,b) = \epsilon\text{-closure}(\delta(8, 9, 3, 4, 6), b))$
$= \epsilon\text{-closure}(\phi \cup \phi \cup \phi \cup \phi \cup 7)$
$= \{7, 8, 9, 3, 4, 6\}$

xix. $\delta'(9, a) = \epsilon\text{-closure}(\delta(9), a)$

$= \phi.$

xx. $\delta(9, b) = \epsilon\text{-closure}(\delta(9, b))$

$= \phi.$

NFA without transition:

δ	a	b
→ 0	{1}	ϕ
1	ϕ	{2, 3, 4, 6, 9}
2	{5, 8, 9, 3, 4, 6}	{7, 8, 9, 3, 4, 6}
3	{5, 8, 9, 3, 4, 6}	{7, 8, 9, 3, 4, 6}
4	{5, 8, 9, 3, 4, 6}	ϕ
5	{5, 8, 9, 3, 4, 6}	{7, 8, 9, 3, 4, 6}
6	ϕ	{7, 8, 9, 3, 4, 6}
7	{5, 8, 9, 3, 4, 6}	{7, 8, 9, 3, 4, 6}
8	{5, 8, 9, 3, 4, 6}	{7, 8, 9, 3, 4, 6}
* 9	ϕ	ϕ

Minimization of DFA : (Reduction of DFA).

Eg: Minimize the following DFA. The DFA transition table is given below

δ	0	1
$\rightarrow q_0$	q_1	q_3
q_1	q_2	q_4
q_2	q_1	q_4
q_3	q_2	q_4
$* q_4$	q_4	q_4

Sol: The given DFA, $M = \{Q, \Sigma, \delta, q_0, F\}$

$Q = \{q_0, q_1, q_2, q_3, q_4\}$.

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

$q_0 = \{q_0\}$

$F = \{q_4\}$

$\delta = Q \times \Sigma \rightarrow Q$.

Minimise the given DFA as

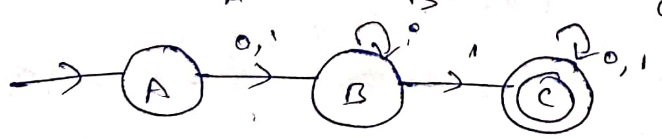
$\Pi_0 = (q_0, q_1, q_2, q_3, q_4)$

$\Pi_1 = \underbrace{(q_0, q_1, q_2, q_3)}_{I} \underbrace{(q_4)}_{II}$

$\Pi_2 = \underbrace{(q_0)}_{I} \quad \underbrace{(q_1, q_2, q_3)}_{II} \quad \underbrace{(q_4)}_{III}$

$\Pi_3 = \underbrace{(q_0)}_{I} \quad \underbrace{(q_1, q_2, q_3)}_{II} \quad \underbrace{(q_4)}_{III}$

A B C.



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

where $Q = \{A, B, C\}$

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

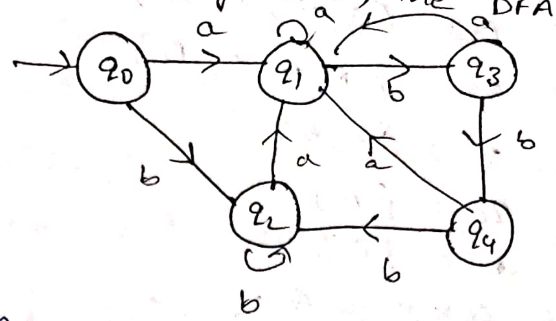
$q_0 = \{A\}$

$F = \{C\}$

$\delta = Q \times \Sigma \rightarrow Q$.

Scanned with CamScanner

* Minimize the following DFA, the DFA transition diagram is



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

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

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

$q_0 = \{q_0\}$

$F = \{q_4\}$

$\delta = Q \times \Sigma \rightarrow Q$.

The Minimization of DFA is

Transition table

δ	a	b
$\rightarrow q_0$	q_1	q_2
q_1	q_1	q_3
q_2	q_1	q_2
q_3	q_1	q_4
* q_4	q_1	q_2

Minimization of DFA,

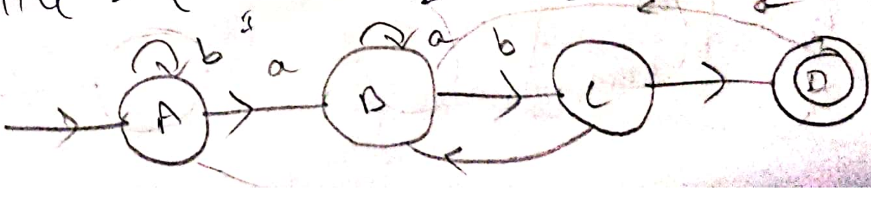
$\Pi_0 = \{q_0, q_1, q_2, q_3, q_4\}$

$\Pi_1 = \{q_0, q_1, q_2, q_3) \; (q_4)$

$\Pi_2 = (q_0, q_1, q_2) \; (q_3) \; (q_4)$

$\Pi_3 = (q_0, q_2) \; (q_1) \; (q_3) \; (q_4)$

$\Pi_4 = (q_0, q_2) \; (q_1) \; (q_3) \; (q_4)$



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

$Q = \{A, B, C, D\}$

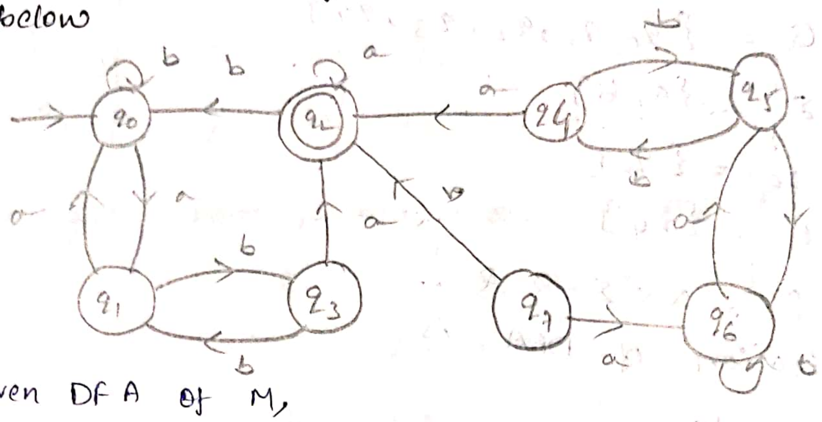
$\Sigma = \{a, b\}$

$q_0 = \{A\}$

$F = \{D\}$

$\delta = Q \times \Sigma \to Q$.

* Minimize the following DFA, the DFA transition is given below



The given DFA of M,

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

where, $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$.

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

$q_0 = \{q_0\}$

$F = \{q_2\}$

$\delta = Q \times \Sigma \to Q$.

δ	a	b
$\to q_0$	q_1	q_0
q_1	q_0	q_3
* q_2	q_2	q_0
q_3	q_2	q_1
q_4	q_2	q_5
q_5	q_6	q_4
q_6	q_5	q_6
q_7	q_6	q_2

To minimi

Π_0 = (

Π_1 = (

Π_2 = (

Π_3 = (

Π_4 = (

Transition

The given

* Finite Au

* Moore ma

in which

and) of

in time

To minimize the DFA,

$\Pi_0 = (q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7)$

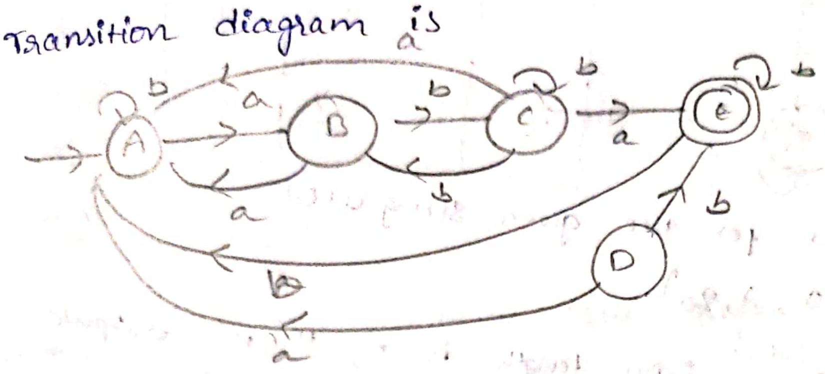
$\Pi_1 = (q_0, q_1, q_3, q_4, q_5, q_6, q_7)\ (q_2)$.

$\Pi_2 = (q_0, q_1, q_5, q_6)\ (q_3, q_4, q_7)\ (q_2)$

$\Pi_3 = (q_0, q_6)\ (q_1, q_5)\ (q_3, q_4)\ (q_7)\ (q_2)$

$\Pi_4 = (q_0, q_6)\ (q_1, q_5)\ (q_3, q_4)\ (q_7)\ q_2)$

Transition diagram is



The given DFA of M,

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

$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$

(or)

$\{A, B, C, D, E\}$

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

$q_0 = \{A\}$

$F = \{E\}$.

* Finite Automata with output :-

* Moore machine :: Moore machine is a finite state machine in which the next state is decided by current state and the input symbol and the output end at a given time depends only on present state of the machine. The moore machine is defined by 6 triple values. They are,

$M_0 = (Q, \Sigma, \delta, \Delta, \lambda, q_0)$

where, Q = finite set of states.

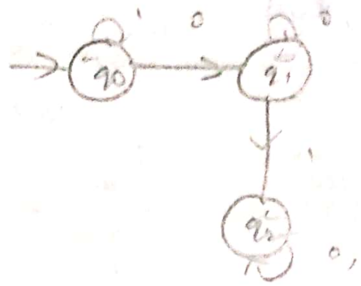
Σ : input Alphabets.

δ : output alphabets

q_0 : Initial states.

Δ : Transition function (or) Input transition function i.e., $Q \times \Sigma \rightarrow Q$.

λ : output transition function. $\Rightarrow \lambda' = Q \rightarrow \Delta$.



δ	0	1	λ
q_0	q_1	q_0	a
q_1	q_1	q_2	b
q_2	q_2	q_2	c

* find out the output for the given string 0100.

$M_o = (Q, \Sigma, \delta, \Delta, \lambda, q_0)$.

If in moore machine, input length is n, then output string is n+1.

$\lambda : Q \rightarrow \Delta$

$\rightarrow q_0 \xrightarrow{0} q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_2 \xrightarrow{0} q_0$.

a b c c c.

Mealy machine:

The mealy machine is a finite state machine, in which output symbols depends upon the input symbols of the current state.

*The mealy machine is defined by 6

where, $M_e = (Q, \Sigma, \delta, \Delta, \lambda, q_0)$

Q : final set of states.

Σ : Input alphabets

q_0 : Initial state.

Δ : output alphabets.

δ : Input transition function.

λ : output transition function.

$\delta = Q \times \Sigma \rightarrow Q$

$\lambda : Q \rightarrow \Delta$

* In mo

* In mo

Example :-

δ	0
q_0	q_1
q_1	q_1
q_2	q_2

* find out th

0100.

Me : (

$\lambda : Q \rightarrow$

* In mealy

output string

* Conversion

* $\lambda'(q, a$

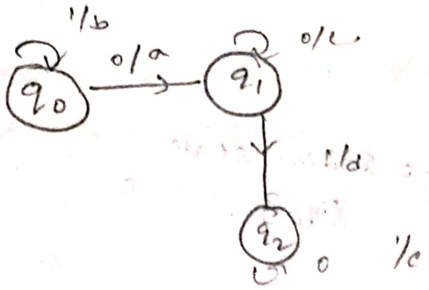
* Convert the

transition table

δ	0
q_0	q_1
	q_3

* In moore machine, the input string length = n then
* In moore machine, the output string length = n+1.

Example :-



δ	0	1
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_2	q_2

λ	0	1
q_0	a	b
q_1	c	d
q_2	e	c.

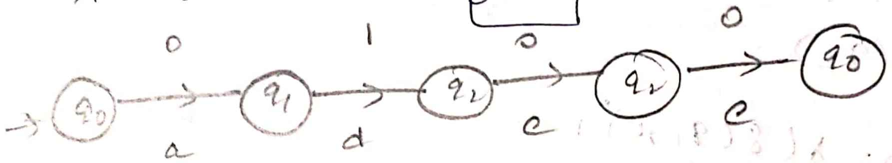
* find out the output for the given string.

0100.

$M_e = (Q, \Sigma, \delta, \Delta, \lambda, q_0)$

$\lambda : Q \rightarrow \Delta$ $\boxed{0100}$. $|w| = m$

o/p = m.



* In mealy machine, the input string length is 'n' then, output string is also 'n'.

* conversion of moore machine to mealy machine :-

* $\lambda'(q, a) = \lambda(\delta(q, a))$.

* Convert the moore machine to mealy machine. The moore transition table is given below.

δ	0	1	λ
$\rightarrow q_0$	q_1	q_2	1
q_1	q_3	q_2	1
q_2	q_2	q_1	0
q_3	q_0	q_3	1

Transition diagram.