Finite Automata.

- **Deterministic Finite automata :- (DFA)**

A (DFA) automata 'M' is a five tuple $M = (Q, \Sigma, \delta, q_0, F)$ where Q is a finite set of states, Σ is a set of input symbols, $q_0 \in Q$ is the initial state, F is a subset of Q is the set of final states. $(F \subseteq Q)$ and δ is a transition function which is defined as following $\delta : Q \times \Sigma \rightarrow Q$.

- **Delta cap function $\hat{\delta}$:-**

This is the extension of δ function.
It is defined as following.

$$[\hat{\delta} : Q \times \Sigma^* \rightarrow Q.]$$

Σ^* : set of strings.
when the length of the string is $\Sigma^* = 1$
then
$\hat{\delta} = \delta$.

Properties of $\hat{\delta}$

(1) $\hat{\delta}(q, \epsilon) = q$. [machine does not move]

(2) $\hat{\delta}(q, a) = \delta(q, a)$
where $a \in \Sigma$

(3) $\hat{\delta}(q, \omega, a) = \delta(\hat{\delta}(q, \omega), a)$

where $\omega \in \Sigma^*$, $a \in \Sigma$
ω: is a string of any length.

→ **Language accepted by (DFA)** :-
It is denoted by $L(M)$ and defined as
$$L(M) = \{x \in \Sigma^* \mid \hat{\delta}(q_0, x) \in F\}.$$

(Initial state to Final state machine move)

Representation of DFA :-
We can represent DFA in the following two way :-

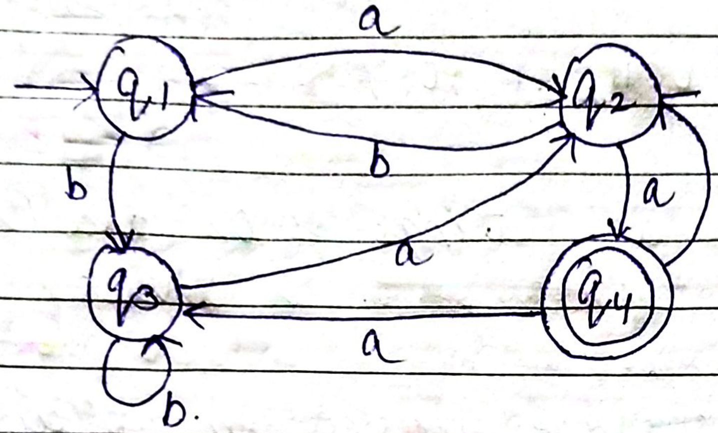
(i) By transition table :- input symbol (a,b)

		a	b
initial state →	q_1	q_2	q_3
	q_2	q_4	q_1
	q_3	q_2	q_3
Final state →	q_4	q_3	q_2

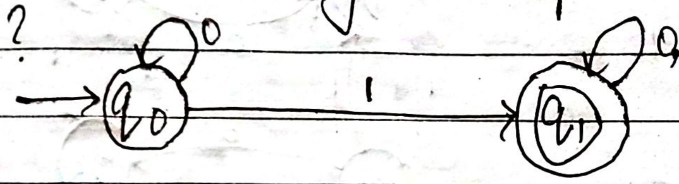
transition function.

Initial state pe →
Final state pe ⊚.

(ii) By transition diagram :-
final state = utne circle draw.



Q-1. Find the lang. accepted by following DFA?



×	0	1
q_0	0	
q_1		

1, 01, 010, 011, 001, 100.........

010

$q_0 \xrightarrow{0} q_0 \xrightarrow{1} q_1 \xrightarrow{0} q_1$

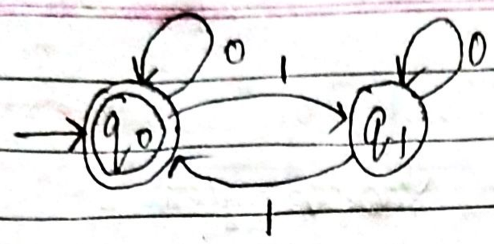
Soln:- $L(M) = \{$

$L(M) = \{$set of all the strings of 0's and 1's which contain 1 atleast once's$\}$

$L(M) = \{0^n 1 w \mid n \geq 0 \text{ and } w \in \{0,1\}^*\}$

jab initial aur final same hoga tab ∈ bhi aiga.

Q-2.

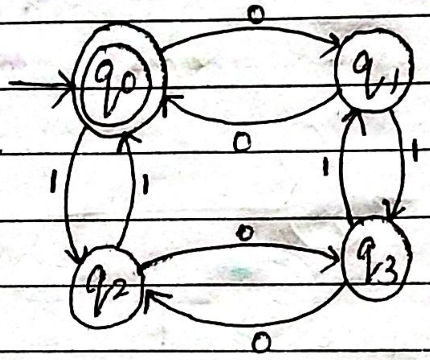


∈, 0, 011, 0011, 11, 101, - - - -

L = {set of all strings of 0's and 1's in which number of 1 is divisible by 2}

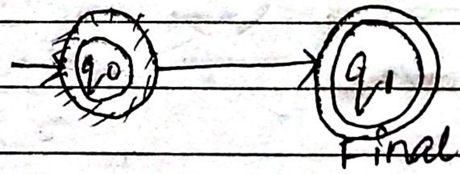
Q-3. Find the language accepted by following DFA.

∈, 11, 00, 0110, 1010, 0101, ...



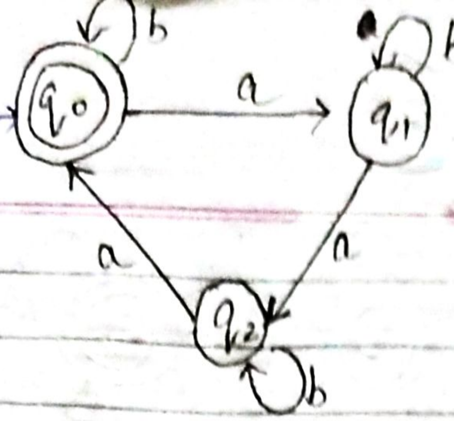
L = {set of all strings of 0's and 1's in which both are in even numbers}.

- if q_1 is ~~final~~
  odd is 0
  even is 1.



- if q_2 is final
  odd is 1.
  even is 0.

- if q_3 is final
  both are odd
  0's and 1's.

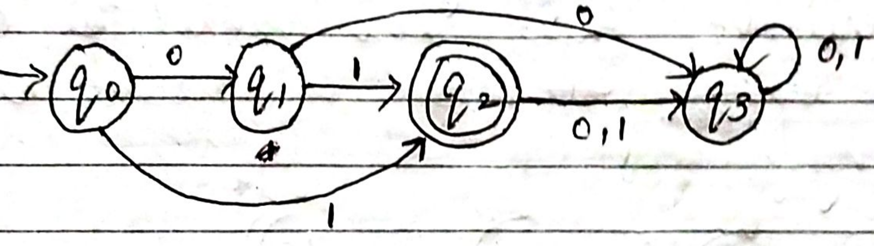
Q-4).


b, bababa, baaab,......

L={set of all strings of a and b in which a is a multipel of 3 and b may be anythink}

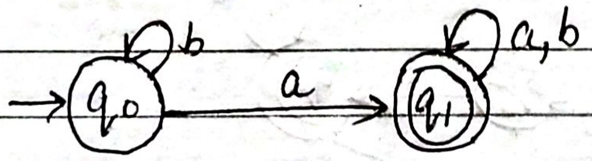
→ if q_1 final   ba, bab, babab,......
$L = \{ w \mid w \in \{a,b\}^* \text{ and } n_a(w) \bmod 3 = 1 \}$.

→ q_2 is final.
$L = \{ w \mid w \in \{a,b\}^* \text{ and } n_a(w) = \bmod 3 = 2 \}$

Q-5).


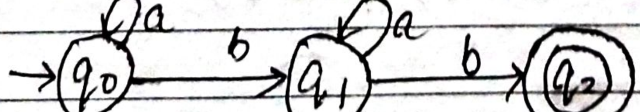
1, 01

$\boxed{L = \{1, 01\}}$

Q-6).


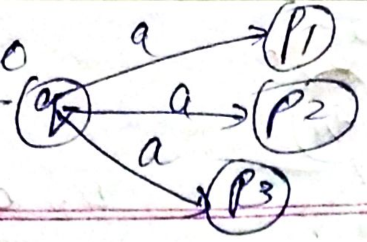
ba, baa, baab, baba, bba, bbaa,......

L={set of all strings of a and b in which a is atleast once and b may be anythink}

Q-7).


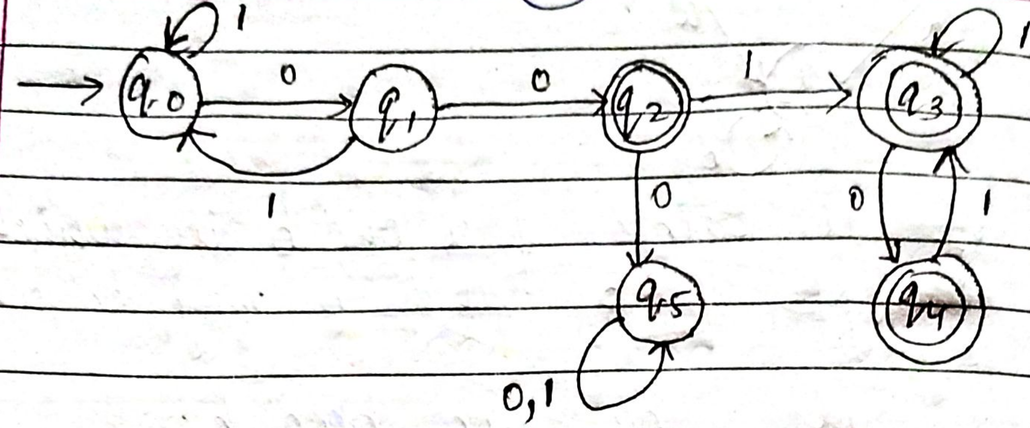
abb, abab, abaab, aabaab, aabab,.........

L={set of all strings of 0's and 1's in which b is only two times and a is anythink} non-deterministic



Date:
Page:

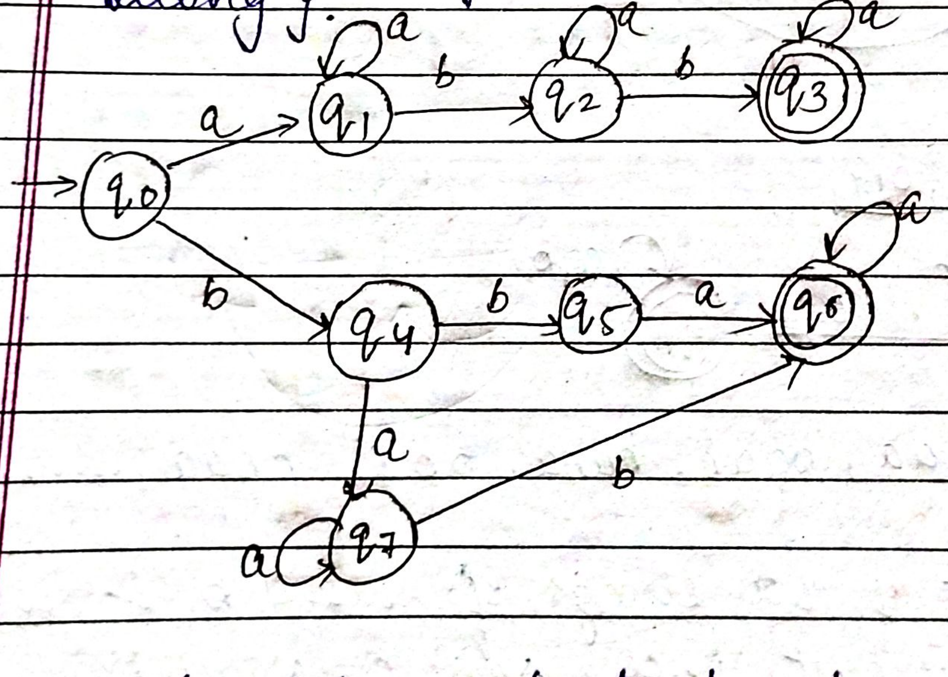
Q-8.



~~00~~, ~~001~~, ~~0011~~, ~~0010~~, ~~010~~

00, 100, 0011, 00110, 00100, ~~001~~,

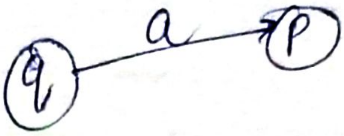
L = {set of all string of 0's and 1's in which consecutive three zero cannot be three}. and ~~atleast two~~ exactly one consecutive zero belong}.



abb, bba, bab, baab, aababa, ...

L = {set of all strings of a and b in which a occur atleast one and b exactly two time}

Deterministic



(ek state se ek hi state pr move karti h.)

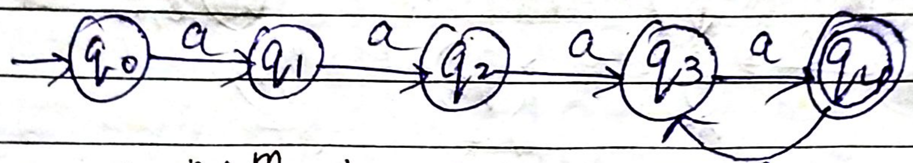
Construct DFA for the following language.

(1) $L = \{ a^{2n} \mid n \geq 2 \}$

aaaa, aaaaaa, aaaaaaaa

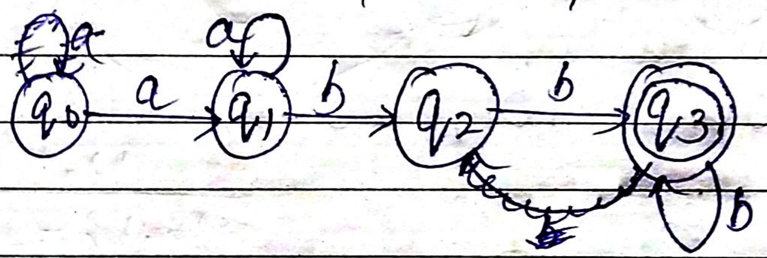
min +1 = states honge

4+1 = 5 = states.



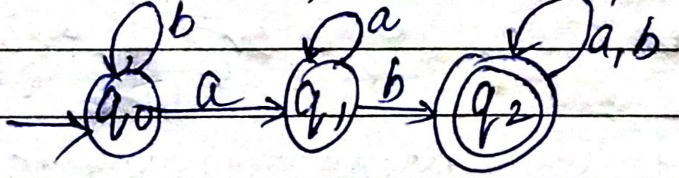
2) $L = \{ a^n b^m \mid n \geq 1, m \geq 2 \}$

abb, aabb, aabbb, abbb,



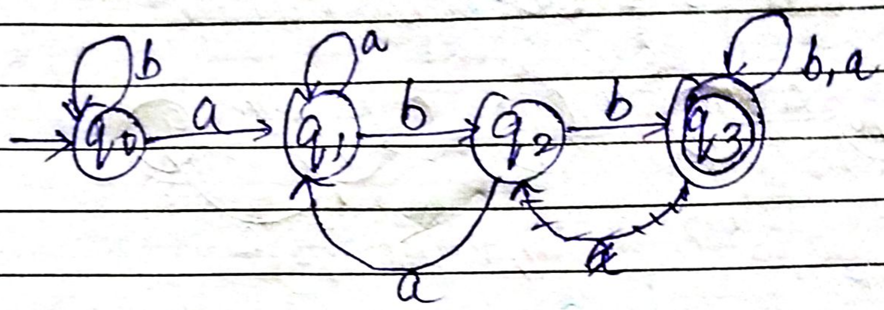
3) L = { which accept all the string of a and b which contains the substring ab.

ab, abab, abbb, babb, aaba, ...



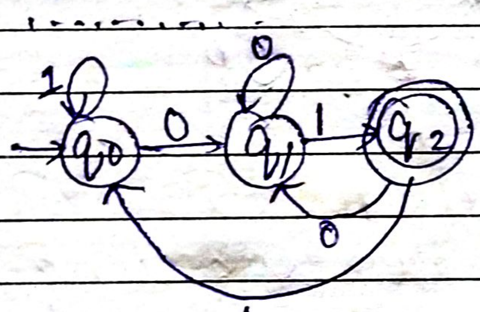
4) set of all the string of a and b which contain the substring of abb.

abb, aabb, babb, babba, ababb, ...



5) set of all the string of 0 and 1 ending with 01.

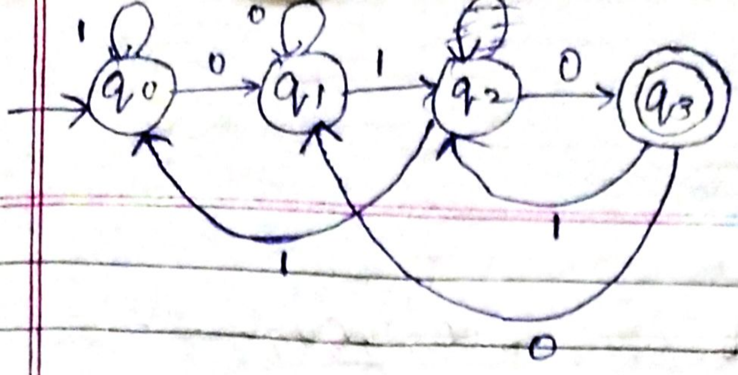
01, 001, 101, 0001, 1101, 0101, ...



6) set of all the string of 0 and 1 ending with 010.

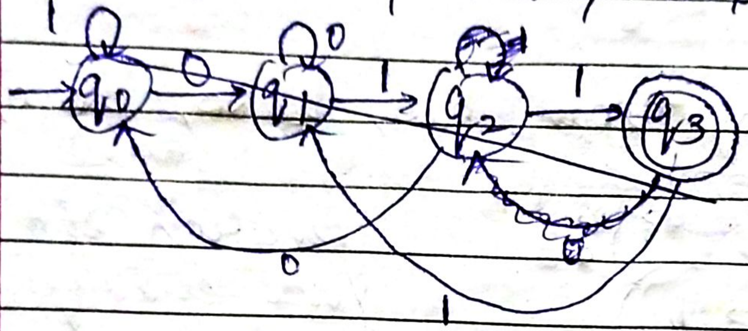
010, 0010, 1010, 11010, 00010, ...

Date: 010 019

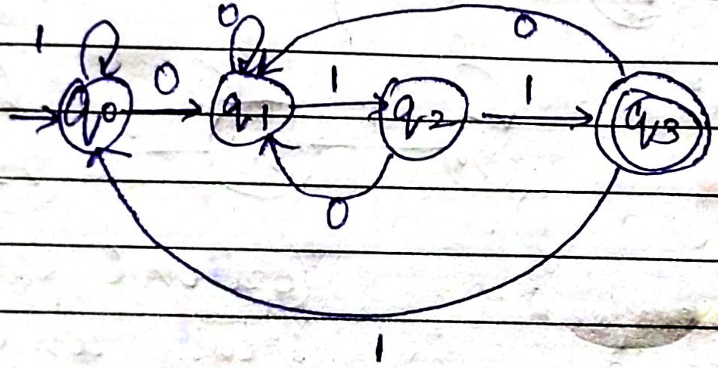


7) set of all string of 0 and 1 ending with 011.

011, 0011, 1011, 01011, 10011

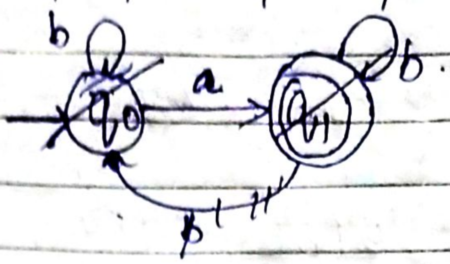


010011
0110
011011
0111011

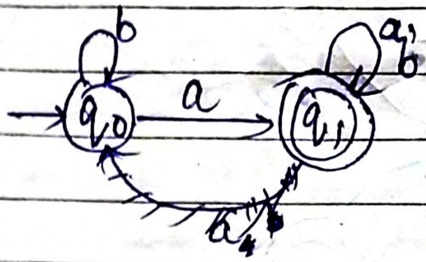


8) For $\Sigma = \{a, b\}$, construct DFA for the following sets
(i) all strings with exactly one a.
(ii) all strings with atleast one a.
iii) all strings no more than 3 a.
iv) all strings with atleast one a and exactly two b.
v). all the strings with exactly two a and more than two b.

(i). ε, a, ab, abb, ba, bab, aba, ------

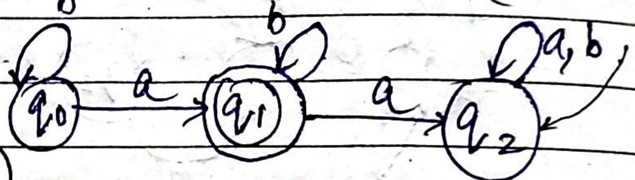


ii) a, ab, aab, aba, bab, aabb, abab,



transition DFA mai hona change.

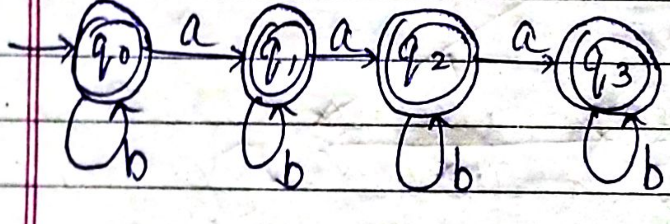
(dead state)



(i).

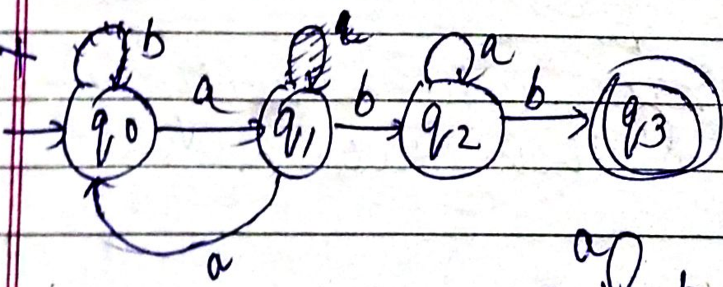
iii). aaa, aaab,
baaa, ababa, bababa,
ε, a, aa, ab, ba, ----

(optional part).
it only completes the DFA definition.

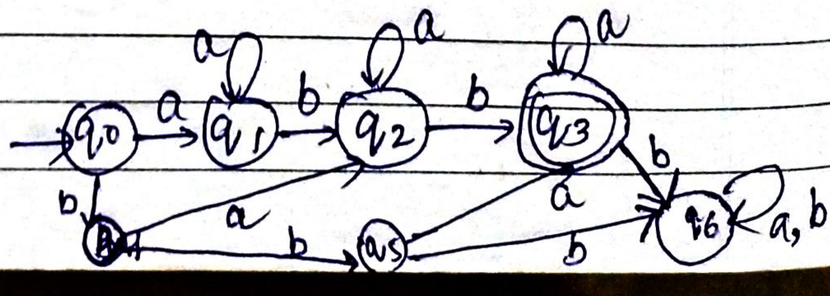


iv) aabb, abb, bba, ——

abb
bab
bba



aaabb

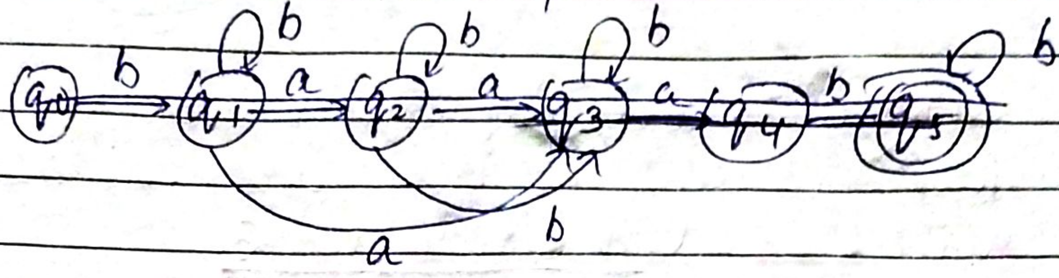


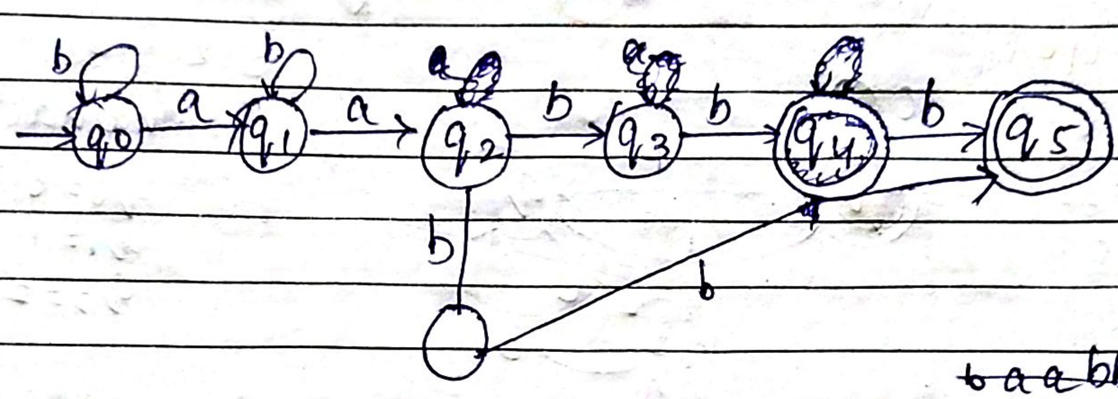
dead state : from where we can't go to the final state again.

Date:
Page:

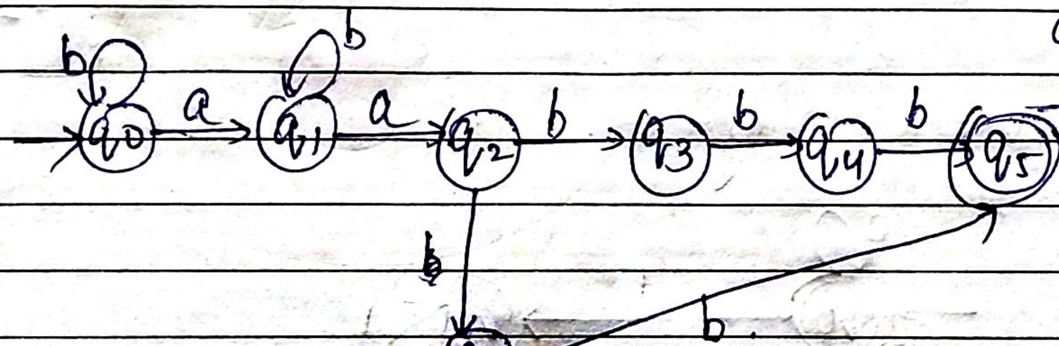
aabbb , ababb , bbbaa , ababbb, ...

v)





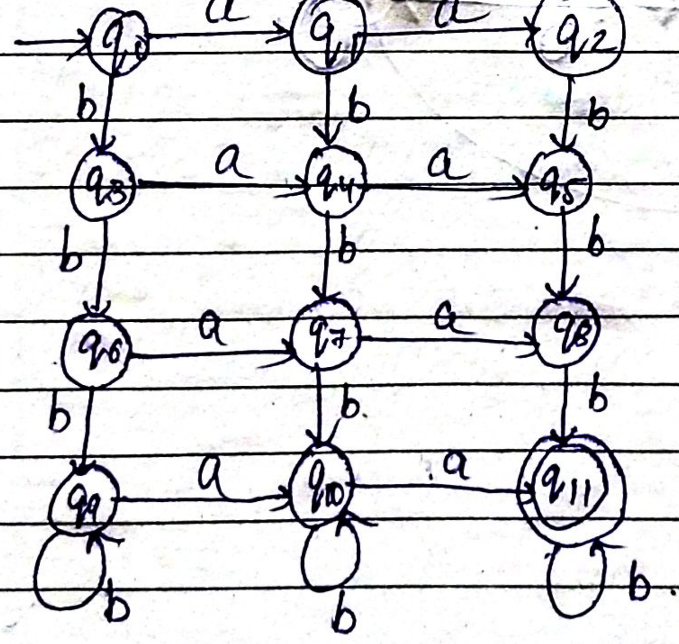
~~baabbb~~.
aabbb
ababb
babab



$\frac{\lfloor 5}{\lfloor 3 \lfloor 2}$ = 10 combination.

use row and column method.

aa bbb.
↓ ↓
3 row 4 column.
corner = final state



initial = final = remainder 0,

Construct a DFA which accepts all the binary strings whose decimal value is divisible by 3.

3 states.

remainder = 0, 1, 2.

8	4	2	1

0

1

10 = 2

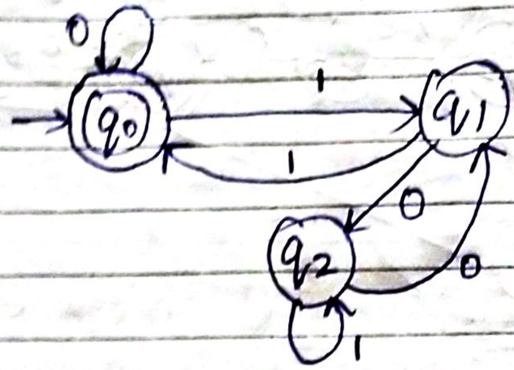
11 = 0

100 = 1

101 = 2

ise pasehna h.

rem



$q_0 = 0$ = final state

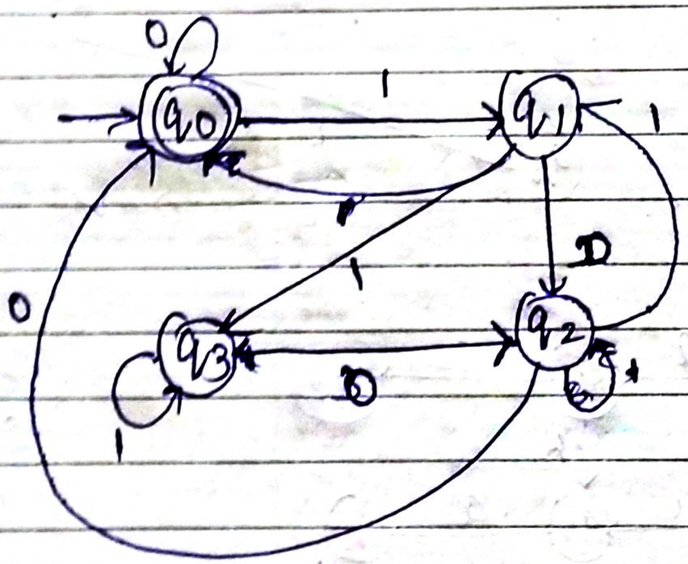
$q_1 = 1$

$q_2 = 2$ correspondly.

eg:- 33 ✓

eg:- 28 ✗.

divisible by 4.



8 4 2 1

1 0 0

0 1 0 1

10

0 1 0 0

0 1 0 1 = 1

0 1 1 0

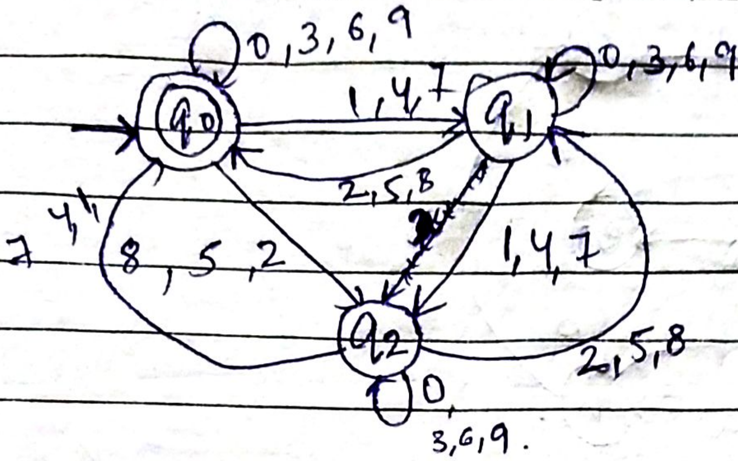
0 1 0

1 1 1 = 3

1 1 0

Date:
Page:

Q- Construct DFA which accepts all the positive decimal number divisible by 3.



$\boxed{0 — 9}$

rem

10 — 1

11 — 2

12 — 0

13 — 1

14 — 2

:

19

rem

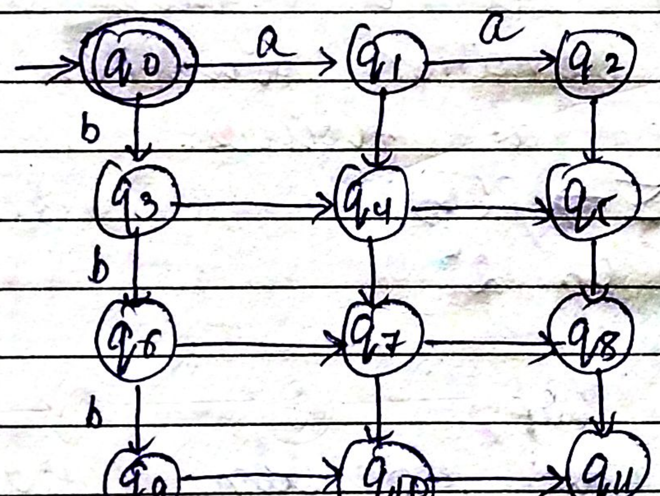
20 — 2

21 — 0

22 — 1

Q-) Construct DFA which accept all the strings of a and b in which no. of a is divisible by 2 and no. of b divisible by 3.

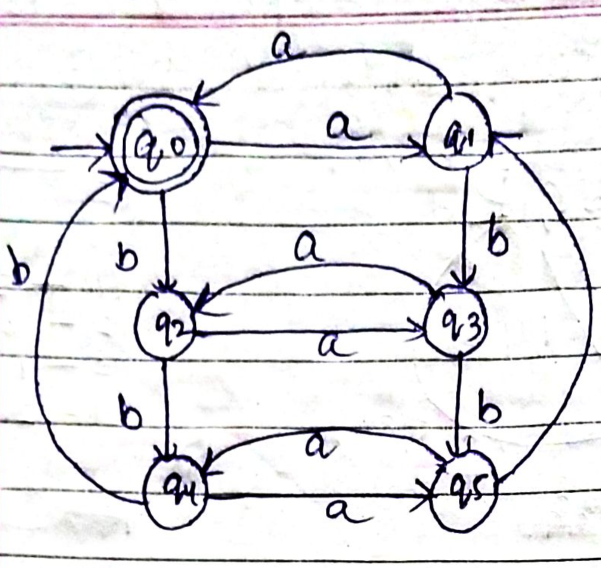
ε, aabbb,



X

minimum
NO. of states.   initial = final

$2 \times 3 = 6$ states.



→ NFA (Non deterministic Finite Automata)

A NFA $M = (Q, \Sigma, \delta, q_0, F)$ is a five tuple. where Q = finite set of states
Σ = finite set of input symbol.
$q_0 \in Q$ is the initial state.
$F \subseteq Q$ is set of final states.
δ is a transition function which is defined as follows.

$$\delta : Q \times \Sigma \rightarrow P(Q).$$
or 2^Q

$\hat{\delta}$ delta cap
This is the extension of delta
it is defined as follows :-

$$\hat{\delta} : Q \times \Sigma^* \rightarrow P(Q).$$

Date:
Page:

Properties of $\hat{\delta}$:-

1) $\hat{\delta}(q, \epsilon) = \{q\}$. all of q.
2) $\hat{\delta}(q, a) = \delta(q, a)$, where $a \in \Sigma$
3) $\hat{\delta}(q, wa) = \bigcup_{p \in \hat{\delta}(q,w)} \delta(p, a)$ where $a \in \Sigma$, $w \in \Sigma^*$

4) $\hat{\hat{\delta}}$: extension of $\hat{\delta}$.

$\hat{\hat{\delta}} : P(Q) \times \Sigma^* \longrightarrow P(Q)$.

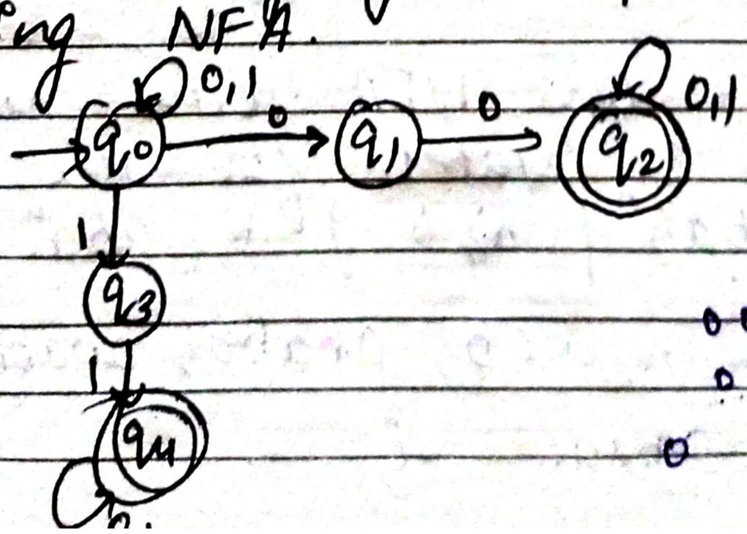
(i) $\hat{\hat{\delta}}(P, x) = \bigcup_{p \in P} \hat{\delta}(p, x)$, $x \in \Sigma^*$, $P \subseteq Q$

Language accepted by NFA

It is define as
$L(M) = \{x \mid x \in \Sigma^* \text{ and } \hat{\delta}(q_0, x) \cap F \neq \phi\}$

(at least one final state).

① Find the language accepted by the following NFA.

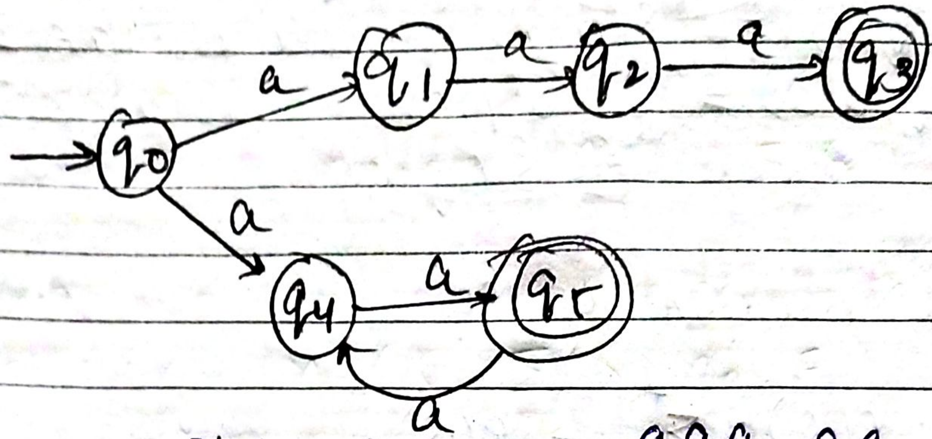


000, 100, 1001,
011, 111, 0111

0

set of all strings of 0 and 1 which contain substring

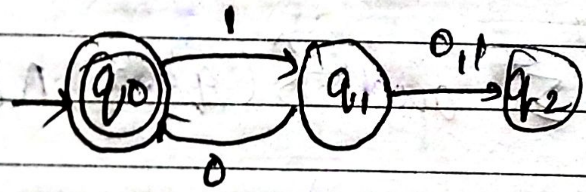
L(M) = { ~~set~~ ~~any~~ ~~string~~ of 00, 11 }.

② 

$L(M) = \{aaa\} \cup \{a^{2n} \mid n \geq 1\}$. aaa, aa, aaaa, aaaaaa

or

L(M) = { set of all strings of a which contain atleast two a and either 3 a or even no. of a }.

③ 

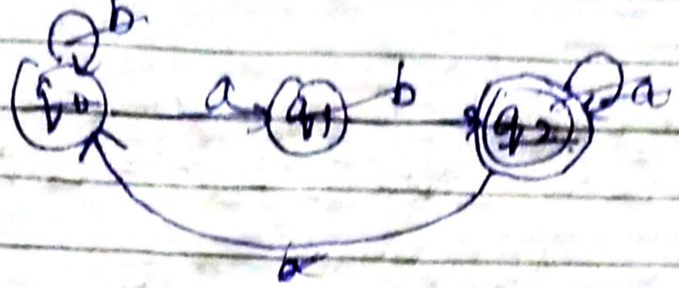
ε, 10, 10101010, ----

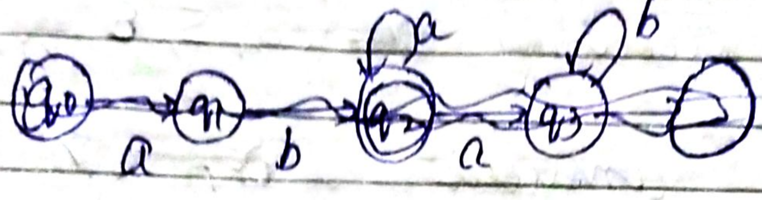
$L(M) = \{(10)^n, n \geq 0\}$.

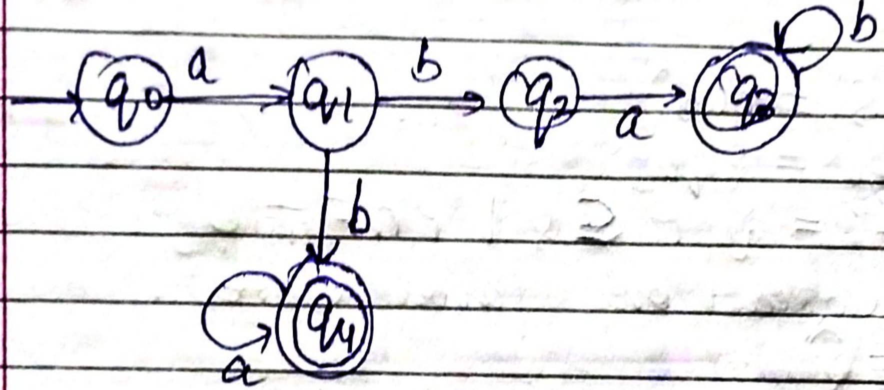
Design an NFA with no more than 5 states for the set $\{abab^n \mid n \geq 0\} \cup \{aba^n \mid n \geq 0\}$

{aba, abab, ababb, ababbb} ∪ {ab, aba, abaa.......}

{ }* ⟶ it means ϵ leke





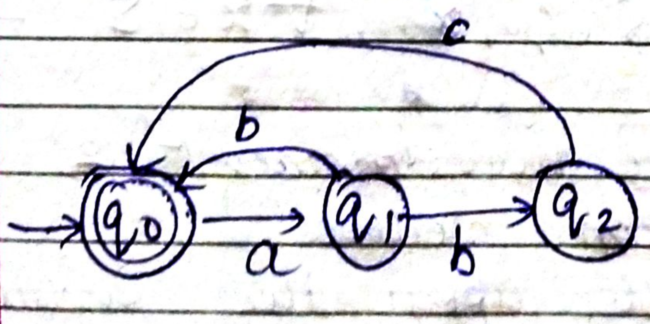


construct an NFA with 3 states that accept the following language.

{ab, abc}*

ϵ, ab, ba, abb, aab, aba, - - - - - -

aaa, bbb, ccc, abc, acb, cba, cab, aabc, - - -



Date:
Page:

Theorem:-

For a given NFA 'M' there exist a DFA 'M'' such that $L(M) = L(M')$

Proof:- Suppose the given NFA 'M' is the following $M = (Q, \Sigma, \delta, q_0, F)$

Now, we construct a DFA 'M'' as following $M' = (Q', \Sigma, \delta', q_0', F')$

where $Q' = P(Q)$

$q_0' = \{q_0\}$

$F' = \{ P \subseteq Q \mid P \cap F \neq \phi \}$

(atleast one final state).

$\delta' = \hat{\delta}$

now, we have to show that

$L(M) = L(M')$ —— ①.

Before proving the statement ①, first we will proof the following statement.

$$\hat{\delta'}(q_0', x) = \hat{\delta}(q_0, x), \forall x \in \Sigma^* \quad —— ②$$

we will prove the statement ② using induction method

Here, we will apply the induction method on the length of the string x.

For $x = \epsilon$

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

using DFA Pro. (1).

$\{\because \hat{\delta}(q, \epsilon) = q\}$

$$= \hat{\delta}(q_0, \epsilon) \qquad \{\because \hat{\delta}(q, \epsilon) = \{q\}\} \text{ NFA (1)}$$

Therefore statement (2) is true for $x = \epsilon$

For $x = a$. where $a \in \Sigma$

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

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

$\because p \in \{q_0\}$

Therefore statement (2) is true for $x = a$.

Suppose the eq^n (2) is true for $x = w$.

Therefore $\hat{\delta}'(q_0', w) = \hat{\delta}(q_0, w)$ ——— (3)

now, we will prove for $x = wa$
where $a \in \Sigma$.

w = length = m
wa = $m+1$ (length).
as a is of length 1

take LHS
$\hat{\delta}'(q_0', wa) = \cancel{\hat{\delta}'(\{q_0\}, wa)}$

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

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

$= \delta'(\hat{\delta}(q_0, w), a)$ [using DFA II]

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

$= \cup\, \hat{\delta}(p, a)$ $p \in \hat{\delta}(q_0, w)$. [NFA II]

$\hat{\delta} = \delta$ when $a = 1$.
$= \hat{\delta}(q_0, wa)$

Therefore statement (2) is also true for
$x = wa$
Hence, equation (2) is proved

Let string $x \in L(M') \Leftrightarrow \hat{\delta}'(q_0', x) \in F'$
($\because$ Lang. accepted by DFA)
$\Leftrightarrow \hat{\delta}(q_0, x) \in F'$ [using eq^n (2)]
$\Leftrightarrow \hat{\delta}(q_0, x) \cap F \neq \phi$ [using F']
$\Leftrightarrow x \in L(M)$

$\therefore L(M') \subseteq L(M)$

Therefore $L(M') = L(M)$.
Hence Proved.

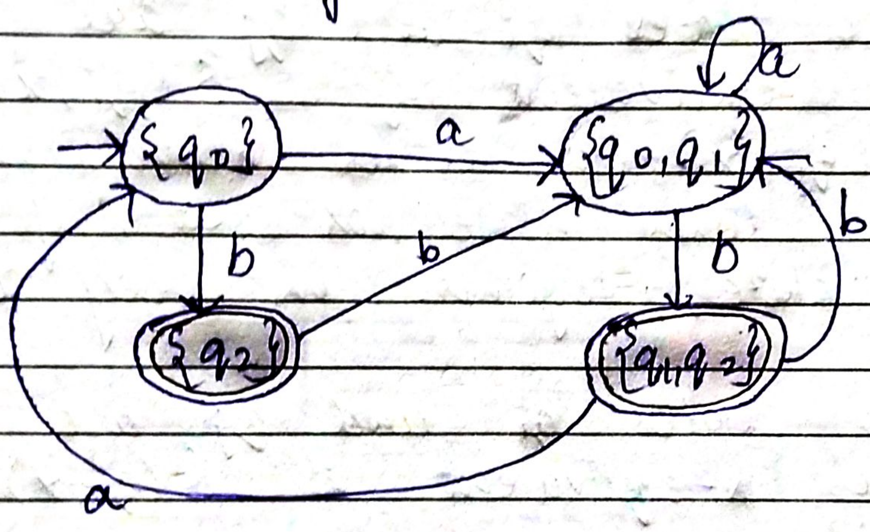
Find the deterministic automata equivalent to the following NFA.

δ	a	b
$\rightarrow q_0$	q_0, q_1	q_2
q_1	q_0	q_1
q_2		q_0, q_1

jaha q_2 hoga woh final state hojaega.

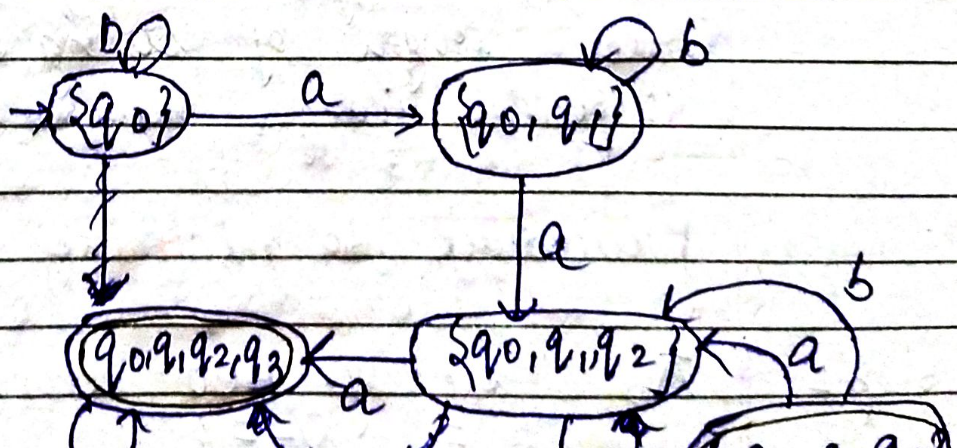
Max. no. of states $= 2^3 = 8$.

$a \rightarrow q_0 = \{q_0 q_1\}$
$b \rightarrow q_1 = q_0$
$q_0 q_1$

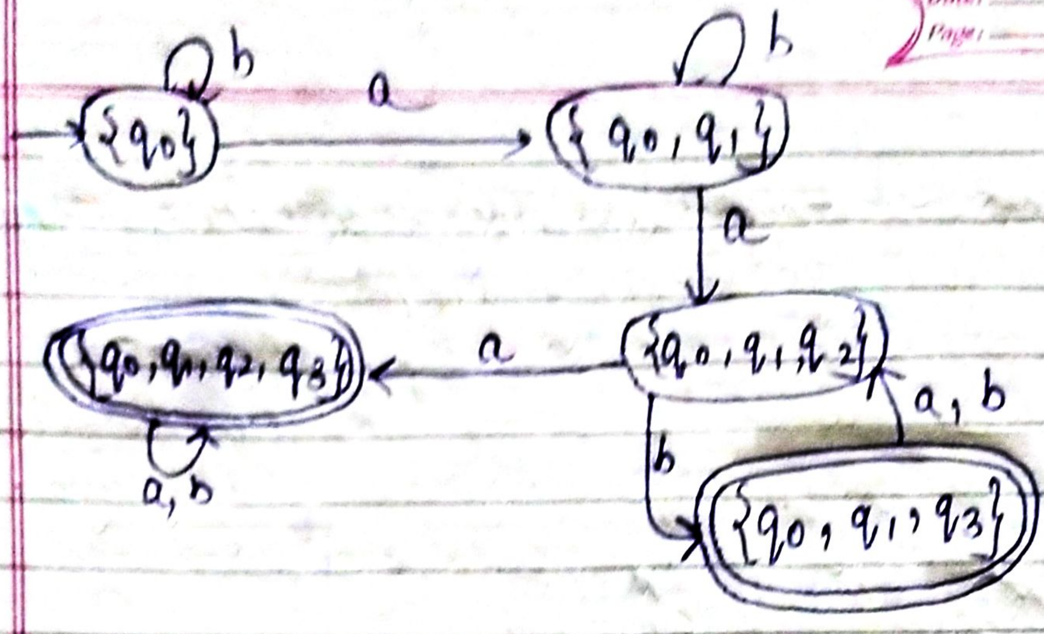


ek state se doosre state se union lena h.

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



max. satete $2^4 = 16$.



Minimization of finite automata

Definition:- Two states q_i and q_j are said to be equivalent if $\hat{\delta}(q_i, x)$ and $\hat{\delta}(q_j, x)$ are either belong into set of final states or set of non-final state together. for every $x \in \Sigma^*$.

Definition:- Two states q_i and q_j are said to be k-equivalent if $\hat{\delta}(q_i, x)$ and $\hat{\delta}(q_j, x)$ are either belong into set of final or set of non-final states $\forall x \in \Sigma^*$. and length of string $|x| \leq K$.
K: positive integer number.

Procedure to minimize a given DFA :-

Step 1: π_0 : set of two sets
$\{F, Q-F\}$

F: set of final state
Q-F: set of non-final state.

$\pi_1 = \{ \quad \}$

step 2 π_1 = if there is any singleton set then copy it in π_1.

q_i, q_j

$\delta(q_i, a) = q_{k_1}$ } same set mai belong
$\delta(q_j, a) = q_{k_2}$ } karenge tabhi equivalent honge.

(input symbol).

→ then q_i and q_j will be equivalent when belong to same set.

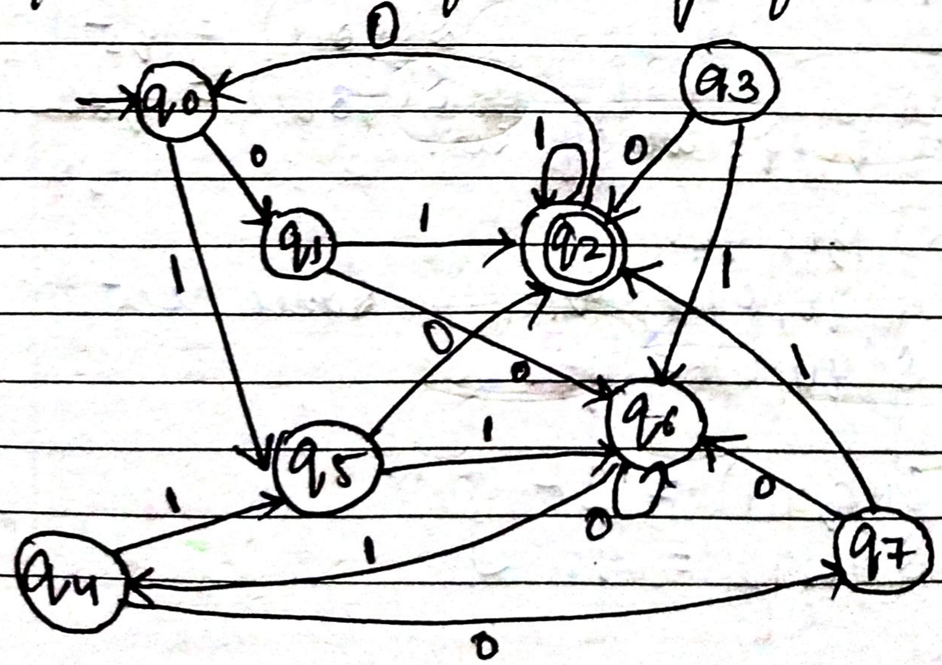
→ After calculation π_1. if π_0 and π_1 both are same then

if π_1 is diff then cal. π_2.

→ Do consecutive π ki value same hogi toh terminate kar jayega.

→ then the value in the state is the no. of states then draw the transition.

Q-1) Construct a minimum state automata equivalent to following finite automata.



Date:
Page:

Step 1. $\pi_0 = \{ \{q_2\}, \{q_0, q_1, q_3, q_4, q_5, q_6, q_7\} \}$

$\downarrow$ copy

Step 2. $\pi_1 = \{ \{q_2\}, \{q_0, q_4, q_6\}, \{q_1, q_7\}, \{q_3, q_5\} \}$

singleton.

take any two state (set) q_0, q_1

$\left.\begin{aligned} \delta(q_0, 0) &= q_1 \\ \delta(q_1, 0) &= q_6 \end{aligned}\right\}$ same set mai h isliye equivalent hosakte h.

$\left.\begin{aligned} \delta(q_0, 1) &= q_5 \\ \delta(q_1, 1) &= q_2 \end{aligned}\right\}$ diff. sets mai h. isliye equivalent nhi h.

now check. q_0, q_3.

~~$\delta(q_0, 0) = q_1$~~
~~$\delta(q_0, 0) = q_6$~~.

$\left.\begin{aligned} \delta(q_0, 0) &= q_1 \\ \delta(q_3, 0) &= q_2 \end{aligned}\right\}$ diff

$\left.\begin{aligned} \delta(q_0, 1) &= q_5 \\ \delta(q_3, 1) &= q_6 \end{aligned}\right\}$ same.

Hence q_0, q_3 are diff set.

again check q_0, q_4.

$\left.\begin{aligned} \delta(q_0, 0) &= q_1 \\ \delta(q_4, 0) &= q_7 \end{aligned}\right\}$ same set.

$\left.\begin{aligned} \delta(q_0, 1) &= q_5 \\ \delta(q_4, 1) &= q_5 \end{aligned}\right\}$

Hence q_0, q_4 are on same set

q_0, q_5.

$$\left.\begin{aligned}\delta(q_0, 0) &= q_1 \\ \delta(q_5, 0) &= q_2\end{aligned}\right\} \text{diff set.}$$

$$\left.\begin{aligned}\delta(q_0, 1) &= q_5 \\ \delta(q_5, 1) &= q_6\end{aligned}\right\} \text{same}$$

q_0, q_5 are on diff set Hence we don't take it into set.

q_0, q_6.

$$\left.\begin{aligned}\delta(q_0, 0) &= q_1 \\ \delta(q_6, 0) &= q_6\end{aligned}\right\} \text{same set.} \quad \left.\begin{aligned}\delta(q_0, 1) &= q_5 \\ \delta(q_6, 1) &= q_4\end{aligned}\right\} \text{same set}$$

q_0, q_6 are equivalent

q_0, q_7.

$\delta(q_0, 0) =$ $\qquad$ $\delta(q_0, 1) =$

$\delta(q_7, 0) =$ $\qquad$ $\delta(q_7, 1) =$

Remaing state q_1, q_3, q_5, q_7 Ko aapas mai check karenge.

q_1, q_3.

$$\left.\begin{aligned}\delta(q_1, 0) &= q_5 \\ \delta(q_3, 0) &= q_2\end{aligned}\right\} \text{diff} \quad \left.\begin{aligned}\delta(q_1, 1) &= q_2 \\ \delta(q_3, 1) &= q_6\end{aligned}\right\} \text{diff.}$$

q_1, q_3 are not equivalent.

q_1, q_5

$\delta(q_1, 0) = q_6$ $\quad$ $\delta(q_1, 1) = q_2$

$\delta(q_5, 0) = q_2$ $\quad$ $\delta(q_5, 1) = q_6$

q_1, q_5 are not equivalent.

q_1, q_7.

$\left.\begin{array}{l}\delta(q_1, 0) = q_6 \\ \delta(q_7, 0) = q_6\end{array}\right\}$ same set $\quad$ $\left.\begin{array}{l}\delta(q_1, 1) = q_2 \\ \delta(q_7, 1) \; q_2\end{array}\right\}$ same set

Hence, q_1 and q_7 are equivalent.

q_3, q_5.

$\delta(q_3, 0) = q_2$ $\quad$ $\delta(q_3, 1) =$

$\delta(q_5, 0) = q_2$ $\quad$ $\delta(q_5, 1) =$

equivaent.

if π_1 is different then calculate π_2.

$\pi_2 = \{\{q_2\}, \{q_0, q_4\}, \{q_6\}, \{q_1, q_7\} \{q_3, q_5\}\}$.

Check

q_0, q_4

$\left.\begin{array}{l}\delta(q_0, 0) = q_1 \\ \delta(q_4, 0) = q_7\end{array}\right\}$ same $\quad$ $\left.\begin{array}{l}\delta(q_0, 1) = q_5 \\ \delta(q_4, 1) = q_5\end{array}\right\}$ same

q_0, q_4 equivalent.

q_0, q_6.

$\left.\begin{array}{l}\delta(q_0, 0) = q_1 \\ \delta(q_6, 0) = q_6\end{array}\right\}$ diff $\quad$ $\left.\begin{array}{l}\delta(q_0, 1) = q_5 \\ \delta(q_6, 1) = q_4\end{array}\right\}$ diff.

q_0, q_6 are not equivalent.

q_1, q_7.

$\delta(q_0, 0) = q_1$ $\quad$ $\delta(q_0, 1) = q_2$

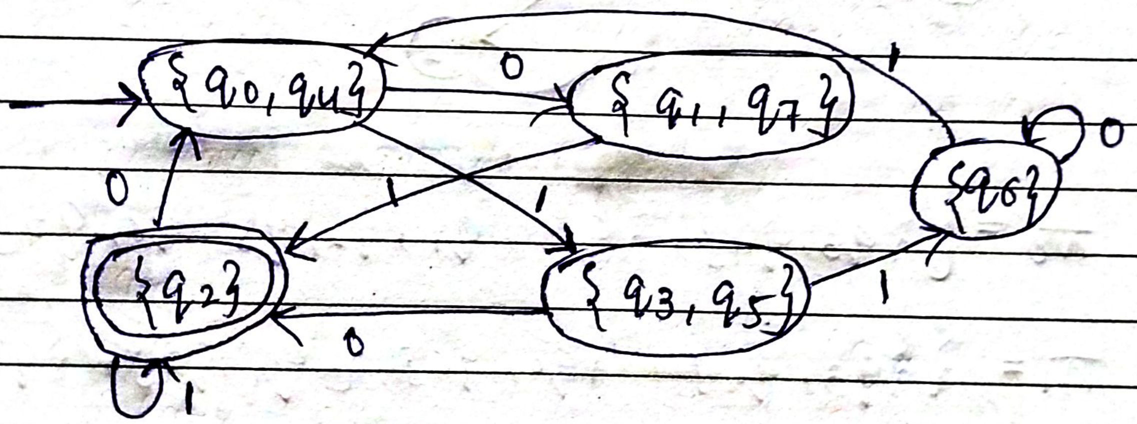
$\delta(q_7, 0) = q_6$. $\quad$ $\delta(q_7, 1) = q_2$

q_1, q_7 equivalent.
q_3, q_5 equivalent

$\pi_3 = \{\{q_2\}, \{q_0, q_4\}, \{q_6\}, \{q_1, q_7\}, \{q_3, q_5\}\}$

$\boxed{\pi_2 = \pi_3}$. terminated

5 total state.
Now draw the transition.



Jaha q_0 hoga woh initial state hoga aur jo final state hoga wo final hoga