PBS

Mod - I

Theory of Computation (TOC)

Complexity Theory
Comutability Theory
Automata theory.

Automata – Greek word

1940-50's finite automata
In 1950 N chomsky → 'Grammar'
In 1969 S. Cook → Interactable problem NP hard.

Mathematical objects, rules and notations

Sets: Groups of elements denoted by uppercase letter and elements are written within the pair of braces.

$S = \{a, b, c, d\}$
$A = \{1, 2, 3, 4\}$
$X = \{5, 10, 13, 20, 25\}$
$X = \{x \in S a$ there $o > a > 6\}$

notations
$\in$ belongs to
$\notin$ not belongs to
$\subseteq$ Subset
$\subset$ Propersubset
$\{\}$ or ϕ null set
$A \cup B$ union
$A \cap B$ intersection
$\bar{A}$ Compliment set $(U - A)$

$A \times B$ Cross multiplication of 2 sets

$A - B$ Set difference

$|A|$ Cardinality of A

2^A power set (includes all the subset)

- **Alphabet:** It is defined as finite nonempty set of symbols
  denoted by Σ, $\Sigma = \{0, 1\}$
  Ex: Binary alphabet 0 & 1
  English A to Z
- **String:** A Sequence of symbols defined over an alphabet.
- **Substring:** x is a substring of w if x appears consequtingly in w
- **Prefix:** Leading symbols of string
- **Suffix:** Trailing symbols of string
- **Empty string:** 'ε' 'Epsilon':
- **Reverse of string:**
  if $\omega = 010111$
  $\omega^r = 111010$
- **Length of string:** $|\omega| = 5$
- **Powers of string:** It is set of all words of length i
  $\Sigma^0 = \{\varepsilon\}$
  $\Sigma^1 = \{0, 1\}$
  $\Sigma^2 = \{00, 01, 10, 11\}$

Σ

Σ

La

Σ

Ex

①

②

③

- Σ^* Kleen closure :

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \ldots\ldots$$

- Σ^+ Kleen plus : $\Sigma^* - \varepsilon$

- Language: A language can be defined as set of strings obtained from Kleen closure where Σ is a set of alphabets

$$L \subseteq \Sigma^*$$

- Sentence: A string that belongs to a language is called word
an
Set of word forms Sentence.

Ex of language :

① The set of strings of 0's and 1's with an equal nos of each

$L = \{\varepsilon, 01, 10, 1100, 0011, 0101, 1010 \ldots\ldots\}$

② The language of all strings made of alphabet 0, 1 consisting of n zeros followed by n 1's where n is ≥ 1

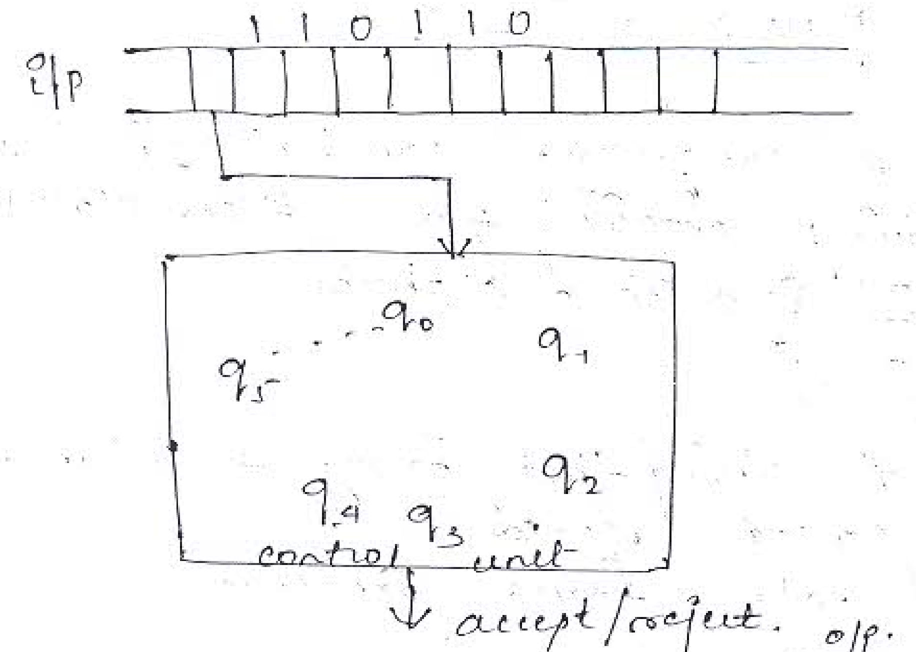
$L = \{01, 0011, 000111, \ldots\ldots\}$

$L = \{0^n 1^n \quad n \geq 1\}$

③ A language over $\Sigma = \{a, b\}$ having substring aa
Ls

$L = \{aa, aab, baa, bbaa, \ldots\ldots\}$

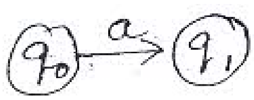
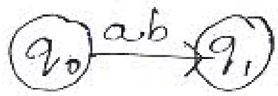
FINITE AUTOMATON (FA)



Automaton is a abstract model of digital compu
It can read a string over an alphabet Σ
Automaton has a control unit which is said to be in one of a finite number of internal state
the output of automaton is either it accepts the string or rejects the string. Finite automaton works as follows
The machine is assumed to be in start state q_0 the input pointer points to the first symbol of the input string After scanning current input symbol Automaton can enter in to any of the state and the input pointer automatically points to next input symbol towords left to right. When the end of the string is encountered the string is accepted if and only if the automaton will be in one of the final states otherwise string is rejected

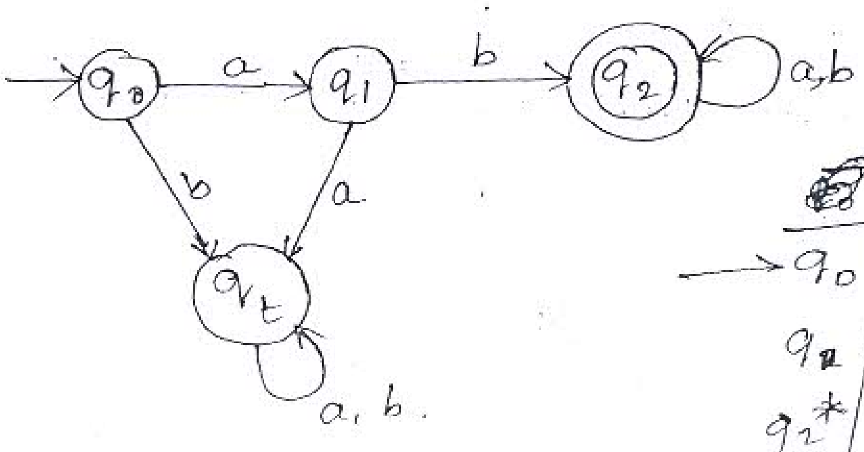
There are two diff type
1) Deterministic FA (DFA)
2) Non Deterministic FA (NFA)

<u>Symbol used in finite automata.</u>

Symbol	Description
	state $q_0, q_1, q_2 \ldots$
	Initial state (start state)
	final ~~Final~~ state (Accept state)
	Transition from state q_0 to q_1 on input symbol a
	Transition from state q_0 to q_0 itself on input symbol a
	Transition from state q_0 to q_1 on input symbol either a or b.

1> Design a DFA to accept all the string over alphabet $\Sigma = \{a, b\}$ which begins with string a, b.

<u>Soln</u>: $L = \{ab, aba, abb \ldots\}$



	a	b
$\rightarrow q_0$	q_1	q_t
q_1	q_t	q_2
q_2^*	q_2	q_2
q_2	q_t	q_t

DFA : defn: It is a 5-tuple or quintuple indicating 5 component

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

where, M - name of the automata

Q - Non empty finite set of states

Σ - Non empty finite set of input symbols (alphabet)

δ - transition function which is a mapping from $\Sigma \times Q \rightarrow Q$

q_0 - initial state or start state

F - Subset of Q is a ~~set of~~ non empty finite no of final state

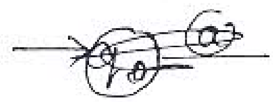
Description: For each input symbol $a \in \Sigma$ ~~from~~ a given there is exactly one transition which can be determined by the current state and the input symbol and hence the machine is called deterministic

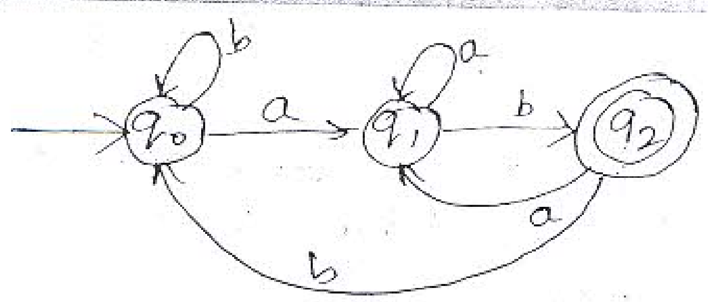
Q) Obtain a DFA to accept strings of a's and b's ending with ab

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

$L = \{\alpha ab \mid \alpha \in (a+b)^*, \Sigma = \{a, b\}\}$

$L = \{ab$

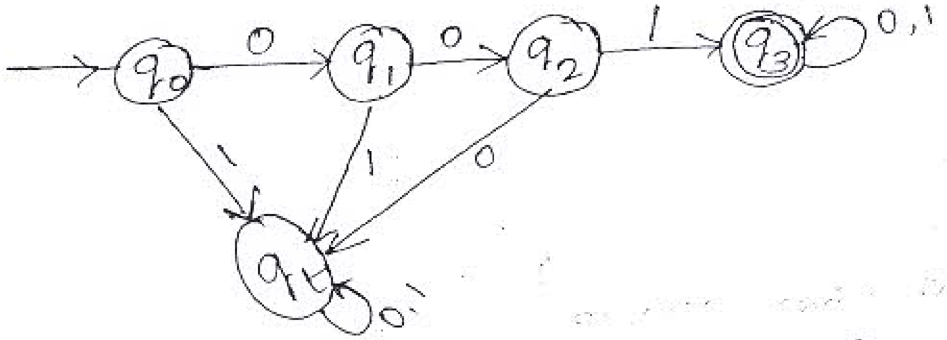




3) Obtain a DFA to accept string 0's and 1's such that string begins with '001'

Sol^n: $L = \{001x \mid x \in (0+1)^*, \Sigma = \{0,1\}\}$

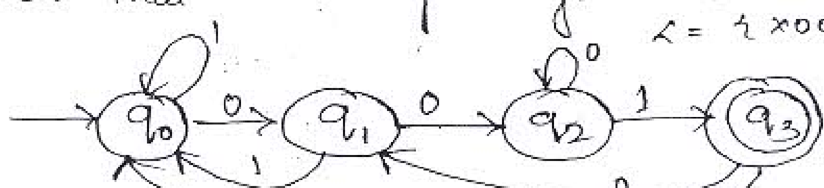
$L = \{001, 0010, 0011.$



4 Obtain a DFA to accept string of 0's and 1's such that string ~~begins~~ is ends with '001'

$L = \{x001 \mid x \in (0+1)^* \ \Sigma = \{0,1\}\}$

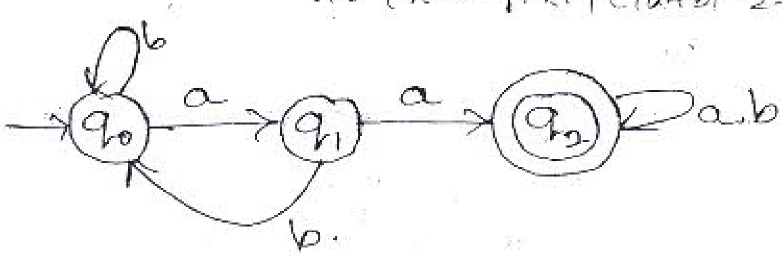
Son:



δ	0	1
$\rightarrow q_0$	q_1	q_0
q_1	q_2	q_0
q_2	q_2	q_3
q_3^*	q_1	q_0.

5> Obtain a DFA to accept string a's and b's having ~~such that~~ string aa ~~is a sub~~

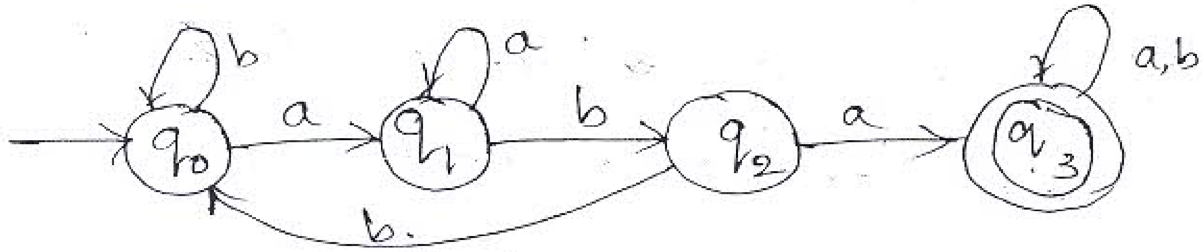
$L = \{x\,aa\,y \mid x, y \in (a+b)^* \ \Sigma = \{a,b\}\}$



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

6) Design a DFA to accept the language over alphabet a, b having substring 'aba'. Somewhere in the string.

Soln $L = \{xabay \mid x, y \in (a+b)^*, \Sigma = \{a, b\}\}$



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

bbaaabbbab

There are two ways to represent finite automata

1) Transition diagram
2) Transition table

Transition Diagram: TD for DFA,

$M = \{Q, \Sigma, \delta, q_0, F\}$ is a graph defined as below

*) Each state of Q corresponds to a node or vertex in the diagram represented using a circle or double circle.

* For each state q_i in Q & each input symbol

$a \in \Sigma$ if, $\delta(q_i, a) \rightarrow q_j$ then, the transition diagram has arc node q_i to q_j labelled with a, If there are more than 1 i/p symbol passing same transition all such input symbols are listed together seperated by a ','.

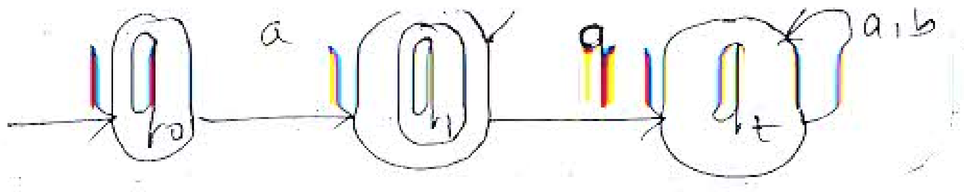
* there is an arrow incidenting on state q_0 indicating that q_0 is a start state
* Only nodes corresponding to accepting state or final state are marked by double circles.

Transition table: T-t for DFA $M = \{Q, \Sigma, \delta, q_0, F\}$ is defined as a conventional tabular representation of a transition function δ. which takes 2 arguments & returns a value. It can be formally defined as below.

* The row's of table corresponds to the states of DFA obtained from Q.
* The columns of the table corresponds to the input symbol Σ
* If q is a current state of DFA & a is a current input symbol. the value written from $\delta(q, a)$ represents the next state of DFA.
* The start state is marked with an 'arrow'
* The final states are marked with a '*'

7) Obtain a DFA to accept a strings of a & b having exactly one 'a'

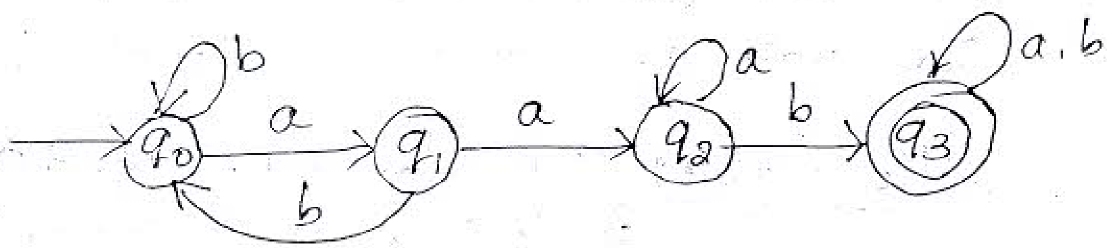
Soln $\{ x a y \mid x, y \in (a+b)^* \quad \Sigma = \{a, b\} \}$

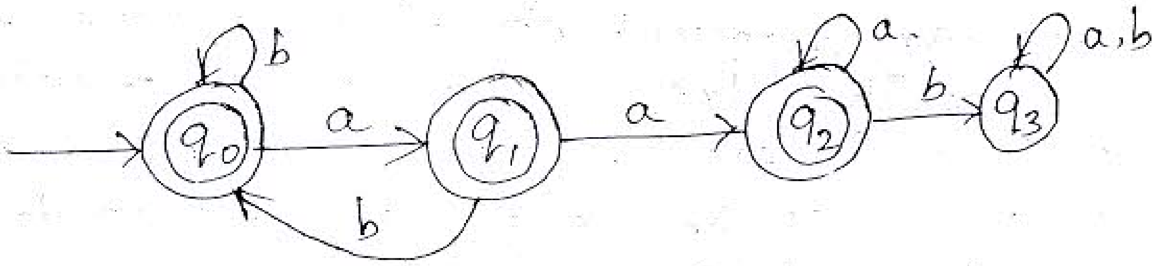


because exactly 3 a's

8) Obtain a DFA to accept strings a & b. accept those having the substring 'aab'

Sol^n:



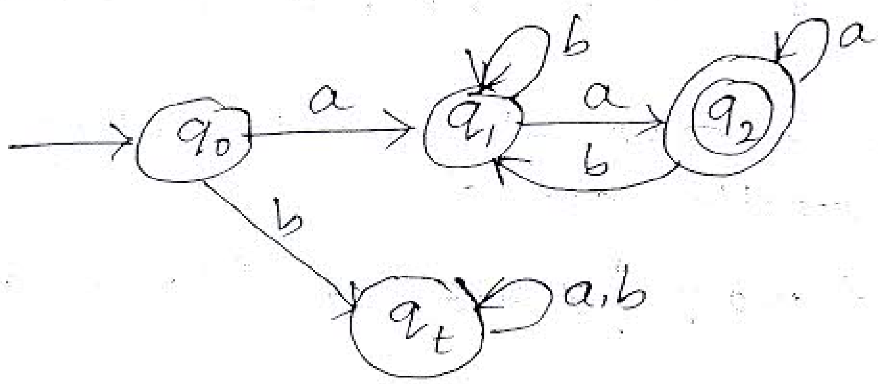


δ	a	b
$\rightarrow q_0^*$	q_1	q_0
q_1^*	q_2	q_0
q_2^*	q_2	q_3
q_3	q_3	q_3

It has 'aab' state so it is not a final state

9) Draw a DFA to accept strings of a & b such that $L = \{awa \mid w \in (a+b)^* \; \Sigma = \{a,b\}\}$

Sol^n:



10)

Sol^n:

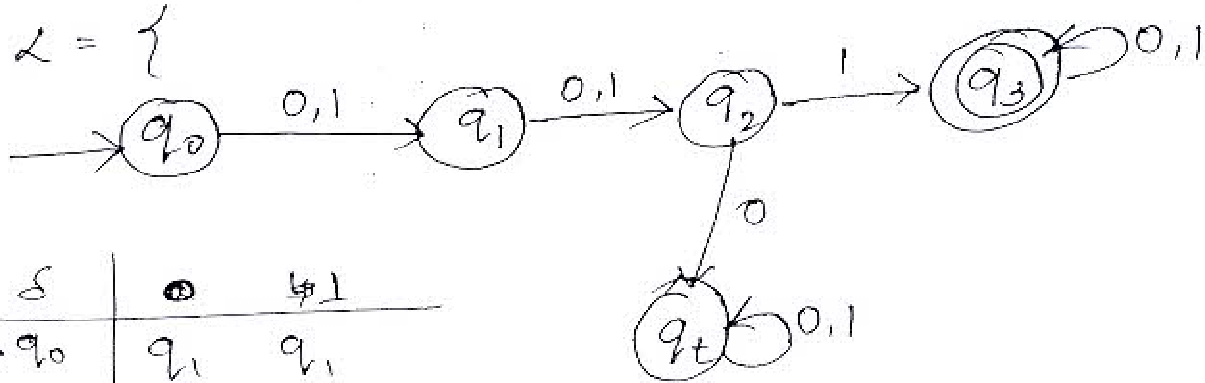
11.

Sol^n

δ	a	b
→ q_0	q_1	q_t
q_1	q_2	q_1
q_2^*	q_2	q_1
q_t	q_t	q_t

10) Construct a DFA of string 0,1 containing 1 in its 3rd position

Sol: $L = \{$

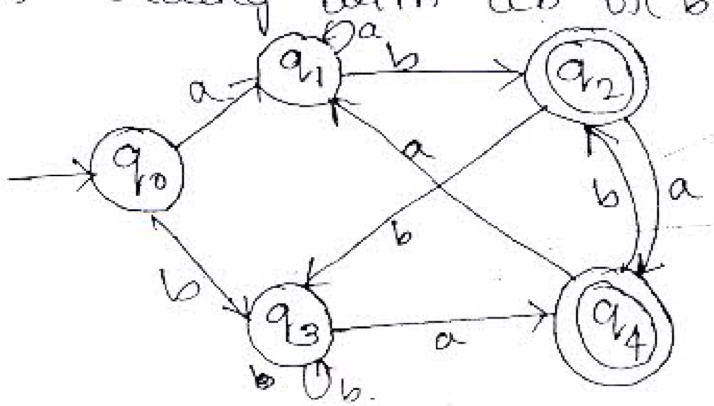


δ	0	1
→ q_0	q_1	q_1
q_1	q_2	q_2
q_2	q_t	q_3
q_3^*	q_3	q_3
q_t	q_t	q_t

6/8/16

11. Obtain a DFA to accept strings of a's and b's ending with ab or ba

Sol:

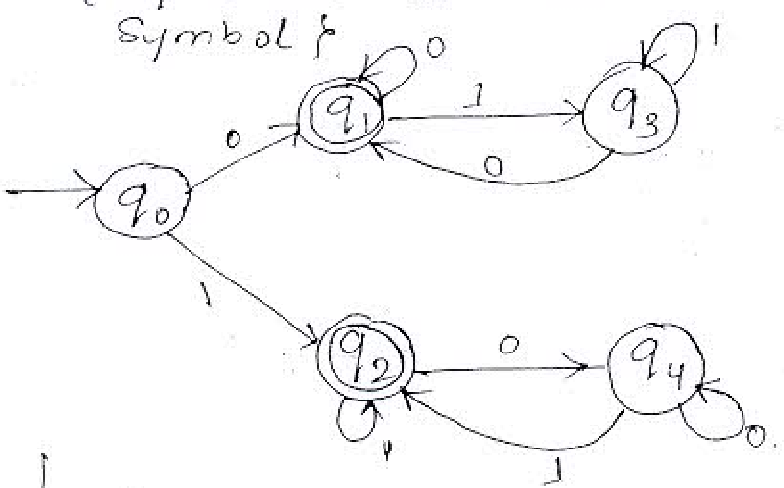


$L = \{ w(ab+ba) \mid w \in (a+b)^* \; \Sigma = \{a,b\}\}$

$\rightarrow q_0$	q_1	q_3
q_1	q_1	q_2
q_2^*	q_4	q_3
q_3	q_4	q_3
q_4^*	q_1	q_2

12) $L = \{w$ / where w starts & ends with same symbol $\}$

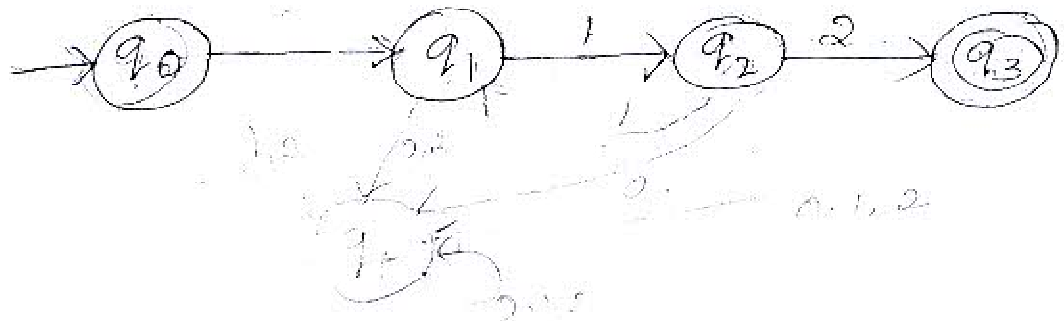
Sol)



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

13) Obtain DFA to ~~obtain~~ accept strings of 0's, 1's & 2's beginning with a 0 followed by odd number of 1's & ending with a 2

Sol):-



δ	0	1	2
$\rightarrow q_0$	q_1	q_t	q_t
q_1	q_t	q_2	q_t
q_2	q_t	q_1	q_3
q_3^*	q_t	q_t	q_t
q_t	q_t	q_t	q_t

Extended transition function of DFA to strings. ($\hat{\delta}$ or δ^*)

Extended transition function describes what happens in an automata if it starts from any state and follows any sequence of inputs. If δ is a transition function, the extended transition function constructed from δ is $\hat{\delta}$ or δ^*

The extended transition function takes a State q and string w and returns state p which is the state the automaton reaches when starting a state q and processing the string w. $\hat{\delta}$ is defined by induction on the length of the input string as below.

d

2

Basis part: $\hat{\delta}(q, \varepsilon) \rightarrow q$ i.e. if we are in state q and read no input symbols then the automaton is still in State q.

Induction part: Suppose w is a string of the form $w = xa$ where a is the last symbol of w and x is the string consisting of all but the last symbol of w. then,

$$\boxed{\hat{\delta}(q, w) \rightarrow \delta(\hat{\delta}(q, x), a)} \quad \text{i.e}$$

To compute $\hat{\delta}(q,w)$ first we have to compute $\hat{\delta}(q,x)$ i.e. the state the automaton is in after processing all but the last symbol.

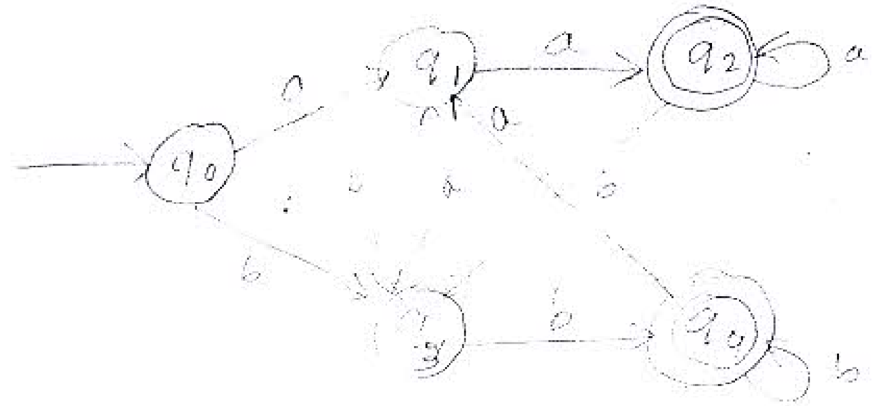
<u>The language of DFA</u>: The language of DFA $M = \{Q, \Sigma, \delta, q_0, F\}$ is denoted by ~~L(M)~~ $L(M)$ and is defined as below.

$$L(M) = \{w \mid \hat{\delta}(q_0, w) \in F\}$$

i.e the language of M is the set of strings w that takes start state q_0 to one of the final state. Any language L accepted by DFA M, then $L(M)$ is known as a <u>regular language</u>

1) Design a DFA to accept all strings of a & b were strings ends with aa or bb.

<u>Soln</u>:



write an extended transition for the string abbbaa

$$\hat{\delta}(q_0, abbbaa) \rightarrow \delta(\hat{\delta}(q_0, abbba), a)$$

$$\rightarrow \delta(\delta(\hat{\delta}(q_0, abbb), a), a)$$

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

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

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

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

$$\rightarrow \delta(\delta(\delta(\delta(\delta(\delta(q, a), b), b), b), a), a)$$

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

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

$$\rightarrow \delta(\delta(\delta(q_4, b), a), a)$$

$$\rightarrow \delta(\delta(q_4, a), a)$$

$$\rightarrow \delta(q_1, a)$$

$$\rightarrow q_2$$

Divisible by K problem

For constructing a DFA that divides a number by K, the transition can be obtained by using the following relation

$\boxed{\delta(q_i, a) \rightarrow q_j}$ where, j is calculated using, $\boxed{j = (r * i + d) \bmod K}$

where, r = radix of the input

i = it is remainder obtained after dividing by K (0 – (K-1))

d = represents the digits

K = divider.

Step to be followed to obtain the DFA using above equation

1. Identify the radix, digits, and divider K

2. Compute the possible remainder, these remainder represents the states of DFA.

3. Find the transitions of DFA using equations

4. Construct the DFA using transition obtained in step 3 and identify the start state at the final stage.

Problems: Give a DFA accepting the string over the alphabet 0,1 where the strings which were interpreted as binary integers is a multiple of 3.

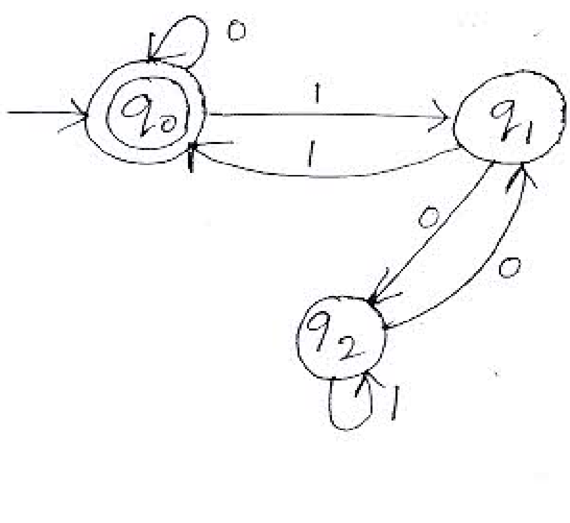
Soln: $r = 2$

$d = \{0, 1\}$

$K = 3.$

$i = 0, 1, 2.$

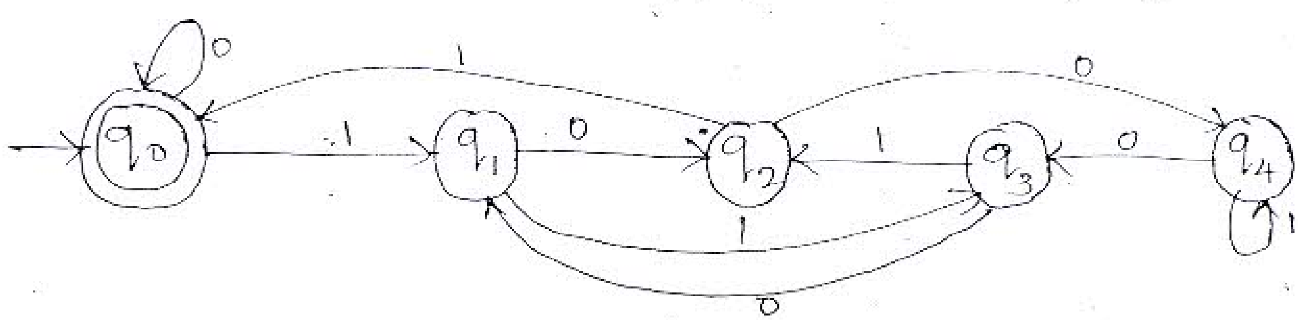
i	d	$j = (r * i + d) \bmod K$	j
q_0	0	$(2*0+0) \bmod 3$	0
	1	$(2*0+1) \bmod 3$	1
q_1	0	$(2*1+0) \bmod 3$	2
	1	$(2*1+1) \bmod 3$	0
q_2	0	$(2*2+0) \bmod 3$	1
	1	$(2*2+1) \bmod 3$	2



2) Obtain DFA to accepting the string over the string 0,1 where the string which were interpreted as binary integer is divisible of 5.

Sol) $r = 2$, $d = \{0,1\}$, $k = 5$, $i = 0,1,2,3,4$.

i	d	$j = (r * i + d) \bmod k$	j
q_0	0	$(2*0+0) \bmod 5$	0
	1	$(2*0+1) \bmod 5$	1
q_1	0	$(2*1+0) \bmod 5$	2
	1	$(2*1+1) \bmod 5$	3
q_2	0	$(2*2+0) \bmod 5$	4
	1	$(2*2+1) \bmod 5$	0
q_3	0	$(2*3+0) \bmod 5$	1
	1	$(2*3+1) \bmod 5$	2
q_4	0	$(2*4+0) \bmod 5$	3
	1	$(2*4+1) \bmod 5$	4



$\hat{\delta}(q_0, 1111) \rightarrow \hat{\delta}(\delta(q_0, 111), 1)$

$\hat{\delta}(\delta(\delta(q_0, 11), 1), 1)$

$\hat{\delta}(\delta(\delta(\delta(q_0, 1), 1), 1), 1)$

$\delta(\delta(\delta(q_1, 1), 1), 1)$

$\delta(\delta(q_3, 1), 1$

$\delta(q_2, 1)$.

q_0.

Q. Design a DFA to accept all the inputs of binary which were interpreted as binary integer is divisible by 5 but Starts with 1.

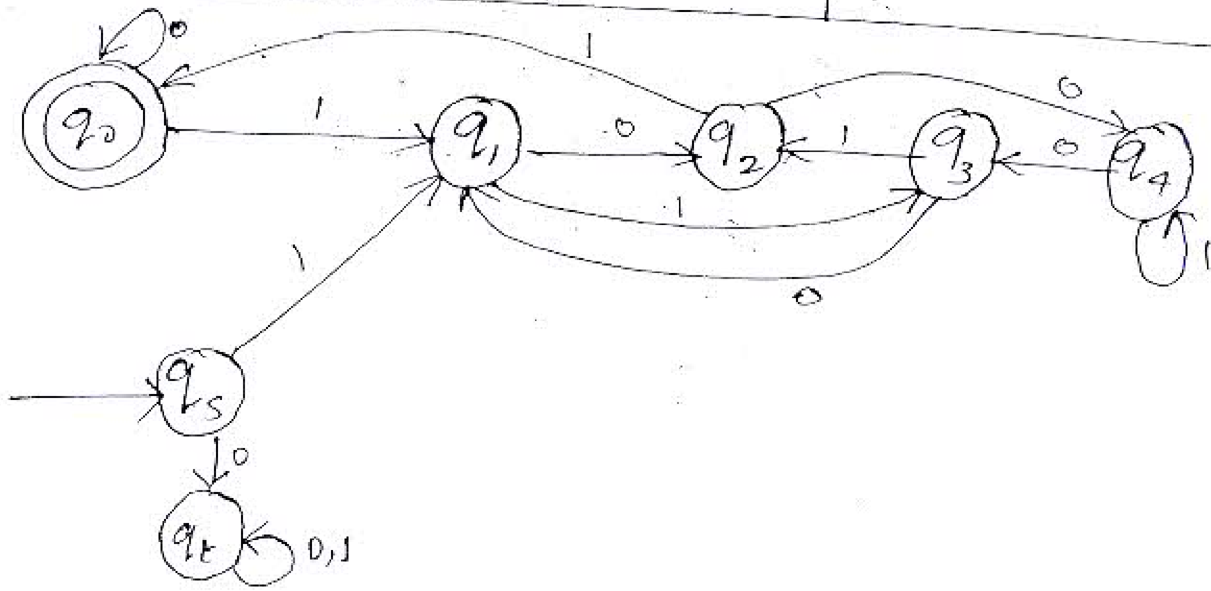
Sol^n: $r = 2$

$d = \{0, 1\}$

$K = 5$

$i = 0, 1, 2, 3, 4$

i	d	$j = (r * i + d) \bmod K$	j
q_0	0	$(2 * 0 + 0) \bmod 5$	0
	1	$(2 * 0 + 1) \bmod 5$	1
q_1	0	$(2 * 1 + 0) \bmod 5$	2
	1	$(2 * 1 + 1) \bmod 5$	3
q_2	0	$(2 * 2 + 0) \bmod 5$	4
	1	$(2 * 2 + 1) \bmod 5$	0
q_3	0	$(2 * 3 + 0) \bmod 5$	1
	1	$(2 * 3 + 1) \bmod 5$	2
q_4	0	$(2 * 4 + 0) \bmod 5$	3
	1	$(2 * 4 + 1) \bmod 5$	4



4) obtain a DFA to accept all the inputs of binary which were interpreted in reverse as binary integer is divisible by 5 but starts with 1.

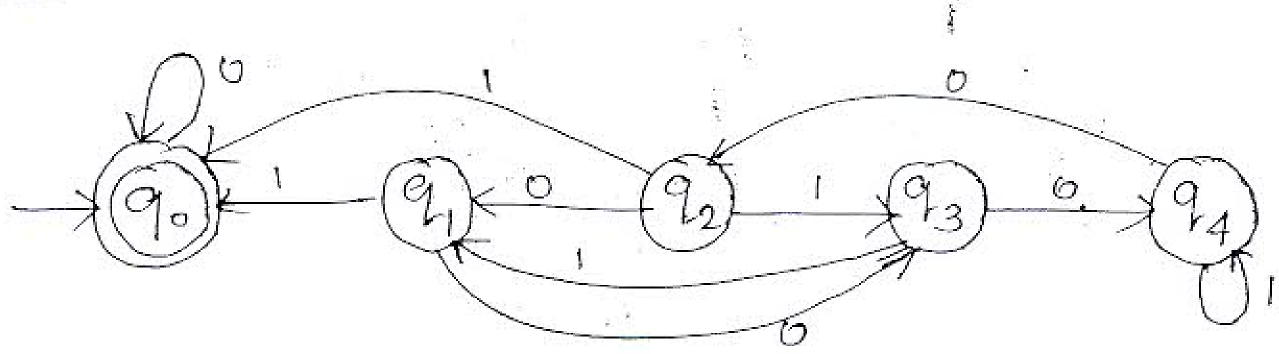
Sol: $r = 2$

$d = \{0,1\}$

$k = 5$

$i = 0,1,2,3,4.$

i	d	$j = (r * i + d) \bmod 5$	j
q_0	0	$(2*0+0) \bmod 5$	0
	1	$(2*0+1) \bmod 5$	1
q_1	0	$(2*1+0) \bmod 5$	2
	1	$(2*1+1) \bmod 5$	3
q_2	0	$(2*2+0) \bmod 5$	4
	1	$(2*2+1) \bmod 5$	0
q_3	0	$(2*3+0) \bmod 5$	1
	1	$(2*3+1) \bmod 5$	2
q_4	0	$(2*4+0) \bmod 5$	3
	1	$(2*4+1) \bmod 5$	4.



Reverse all the directions accept start & self loop

2) Draw a DFA to accept decimal nos divisible by 3

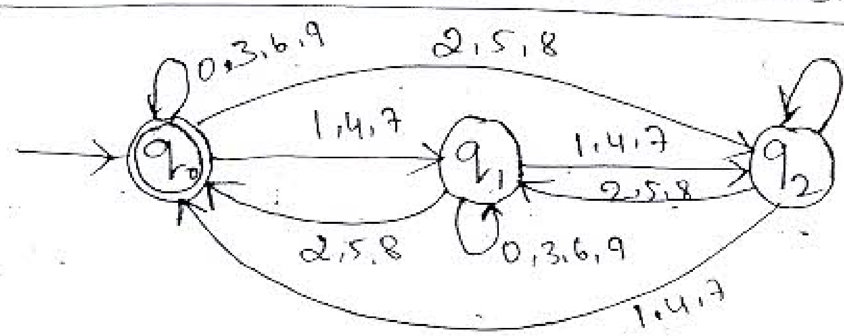
Sol) $r = 10$, $d = \{0,1,2,3,4,5,6,7,8,9\}$
$k = 3$. $i = 0,1,2$.

i $\quad$ d $\quad$ $j = (r * i + d) \bmod k$ $\quad$ j

A group of digits 0-9 can be grouped together based on remainders we get after dividing by 3

$$\left.\begin{array}{l} \{0,3,6,9\} \text{ --- } 0 \\ \{1,4,7\} \text{ --- } 1 \\ \{2,5,8\} \text{ --- } 2 \end{array}\right\} \Rightarrow d = \{0,1,2\}$$

i	d	$j = (r * i + d) \bmod k$	j
q_0	0	$(10 * 0 + 0) \bmod 3$	0
	1	$(10 * 0 + 1) \bmod 3$	1
	2	$(10 * 0 + 2) \bmod 3$	2
q_1	0	$(10 * 1 + 0) \bmod 3$	1
	1	$(10 * 1 + 1) \bmod 3$	2
	2	$(10 * 1 + 2) \bmod 3$	0
q_2	0	$(10 * 2 + 0) \bmod 3$	2
	1	$(10 * 2 + 1) \bmod 3$	0
	2	$(10 * 2 + 2) \bmod 3$	1



Ass: 1) Draw a DFA to accept octal numbers divisible by 4

Imp. 2) Design a DFA to accept all the string over the alphabet 0,1 such that strings are of even length and begins with ~~01~~ 01

3) Give a DFA accepting the strings over alphabet 0,1 Such that strings begins with 01 & ends with 11

4) Design a DFA over alphabet 0,1 which accept the strings which begins with even no of zeroes followed by odd no of 1's and ends with even no of 0's.

Modulo - K - counter problems

1. Obtain the DFA to accept the language
$L = \{ w \mid |w| \bmod 3 = 0 \}$
$\Sigma = \{a\}$

Sol^n: Cases: the $|w| \bmod 3$ results in 3 cases & each of the case represents the state of the automata. The remainder '0' represents q_0, remainder 1 represent q_1 & remainder 2 represent q_2

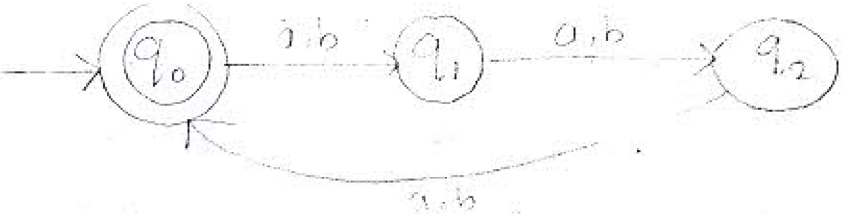
finding the start state.
At the start state of automata no of i/p is zero. and $|w| = 0$ & $|w| \bmod 3 \Rightarrow 0 \bmod 3 = 0$. hence q_0 is the start state

Steps while finding the transitions.

1) From q_0 on reading 'a' the length of w changes to 1 & $|w| \bmod 3 = 1 \bmod 3 = 1$ and hence $\delta(q_0, a) = q_1$

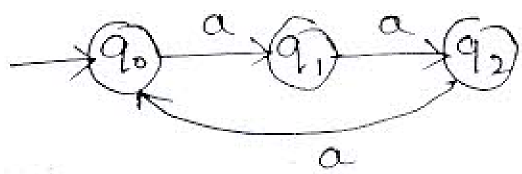
2) from q_1 on reading 'a' the length of w changes to 2 & $|w| \bmod 3 = 2 \bmod 3 = 2$ & hence
$\delta(q_1, a) = q_2$.

3) from q_2 on reading 'a' the remainder changes to '0' again hence $\delta(q_2, a) \rightarrow q_0$.

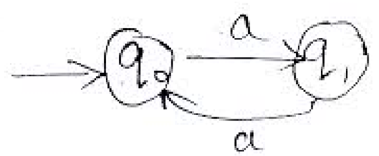


Imp. 2.) Obtain a DFA to accept following language
$L = \{ w \mid |w| \bmod 3 \geq |w| \bmod 2, \ \Sigma = \{a\} \}$

Sol. The above given problem requires us to find out both $|w| \bmod 3$ & $|w| \bmod 2$ which can be represented FA as below.



$Q_1 = \{q_0, q_1, q_2\}$



$Q_2 = \{q_0, q_1\}$

A DFA which has both $|w| \bmod 3$ & $|w| \bmod 2$ functionalities can be obtained by taking cross product of Q_1 & Q_2 which are sets of states of both the FA's.

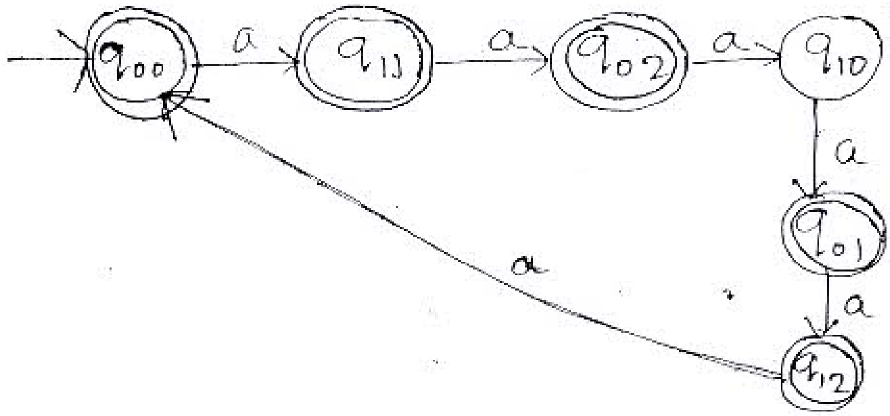
$Q_1 \times Q_2 = \{q_0, q_1\} \times \{Q_0, Q_1, Q_2\}$

$= \{q_{00}, q_{01}, q_{02}, q_{10}, q_{11}, q_{12}\}$

The transition for combined states or pair of state can be found of out as below

$\delta(q_{ij}, a) \rightarrow \delta_1(q_i, a) \ \delta_2(q_j, a)$

	a
q_{00}	q_{11}
q_{01}	q_{12}
q_{02}	q_{10}
q_{10}	q_{01}
q_{11}	q_{02}
q_{12}	q_{00}

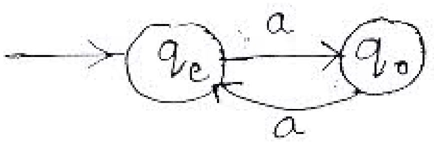


Imp
3) Obtain a DFA to accept strings of ~~A~~ a and ~~B~~ b such that strings having even no of a's & even number of b's

Sol:

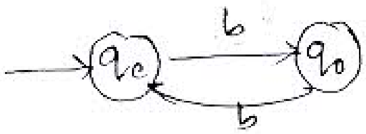
$L = \{w \mid w \in (a+b)^* \; n_a(w) \bmod 2 = n_b(w) \bmod 2 = 0\}$

$n_a(w) \bmod 2$



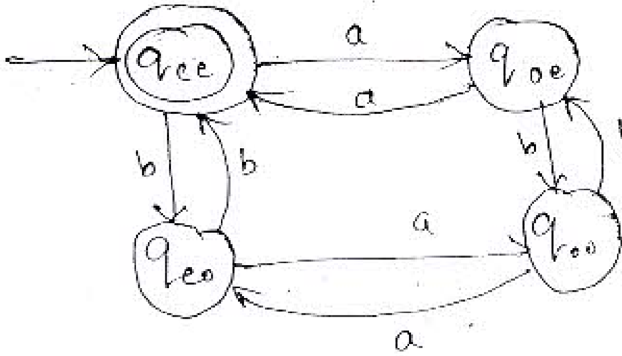
$Q_1 = \{q_e, q_o\}$

$n_b(w) \bmod 2$



$Q_2 = \{q_e \; q_o)$

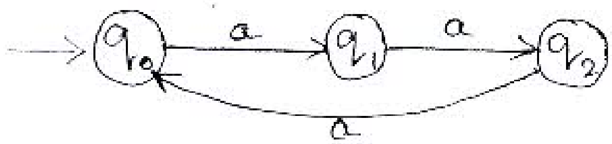
$Q_1 \times Q_2 = \{q_{ee}, q_{eo}, q_{oe}, q_{oo}\}$



4) Obtain a DFA to accept strings of a's & b's such that $L = \{w \mid w \in (a+b)^* \ n_a(w) \bmod 3 = 0$ & $n_b(w) \bmod 2 = 0.\}$

Sol:- $n_a(w) \bmod 3$.

a)



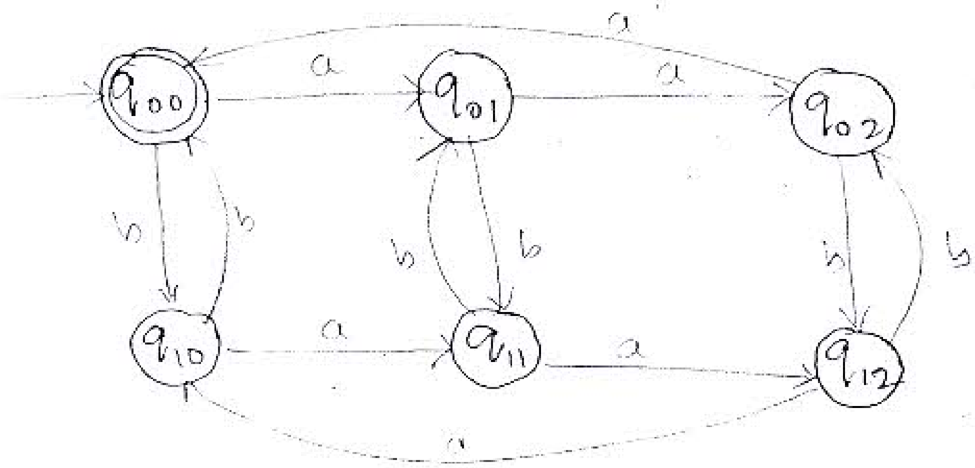
$Q_1 = \{q_0, q_1, q_2\}$

$n_b(w) \bmod 2$

b)

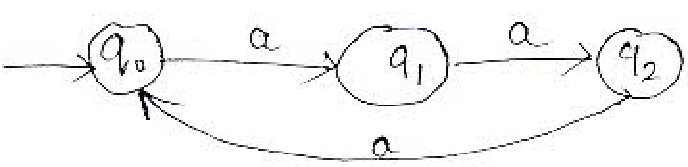
$Q_2 = \{q_0, q_1\}$

$Q_2 \times Q_1 = \{q_{00}, q_{01}, q_{02}, q_{10}, q_{11}, q_{12}\}$



5) Construct a DFA to accept strings of a's and b's such that $L = \{w \mid w \in (a+b)^*$ ~~and~~ $n_a(w) \bmod 3 = n_b(w) \bmod 4\}$.

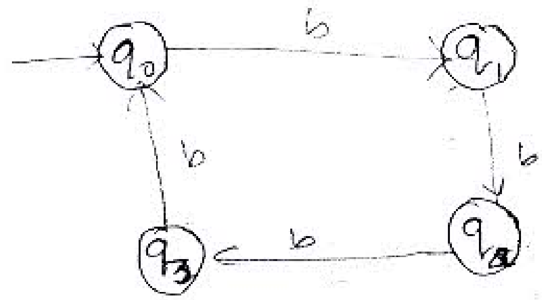
Sol:-



$Q_1 = \{q_0, q_1, q_2\}$

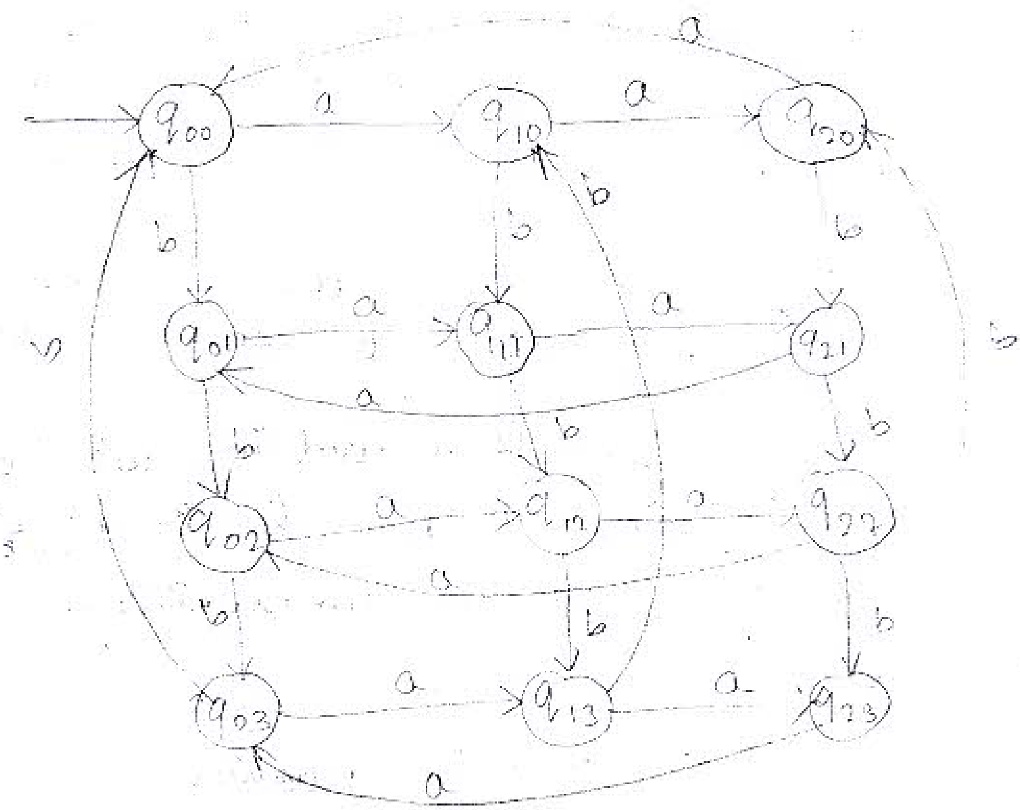
$n_a(w) \bmod 3$

$n_b(w) \bmod 4$



$Q_2 = \{q_0, q_1, q_2, q_3\}$

$Q_1 \times Q_2 = \{q_{00}, q_{01}, q_{02}, q_{03}, q_{10}, q_{11}, q_{12}, q_{13}, q_{20}, q_{21}, q_{22}, q_{23}\}$



Applications of finite automata

1) String matching or processing:-

To designing software for large bodies of text such as collectn of web pages textial documents etc. It is used to find occurance of words or phrases

FA is used for designing & checking the behaviour of digital circuits, design of automatic trafic signals etc.

3) Compiler construction: DFA's are used in various phase of compiler such as.

a) lexical analysis: To identify tokens identifiers, punctuares

b) Syntax analysis: check the syntax of each statement in program

c) Code optimization: It removes un wanted codes

d) Code generation; Convert the high level language to machine code.

4) Software designing: It is used in building softwares. To verify systems having finite number of state such as communication protocol in network

5) FA's are also used in computer graphics, artificial intelligence, & also in game theory

Non Deterministic finite Automata.

It is a quintuple or 5-tuple.

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

where M is a name of the NFA
Q is the nonempty finite set of state
Σ is the " " " input alphabet
$\delta : Q \times \Sigma \to 2^Q$ i.e. mapping from $Q \times \Sigma \to 2^Q$

$q_0 \in Q$
$F \subseteq Q$ set of final state.

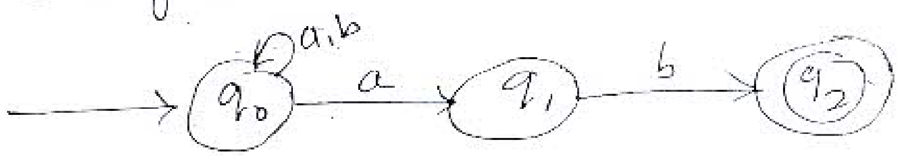
Problems;

1) Construct an NFA to accept all string of the form $L = \{ab a^n \mid n \geq 0\}$

Sol?

δ	a	b
$\to q_0$	$\{q_1\}$	ϕ
q_1	ϕ	$\{q_2\}$
q_2^*	$\{q_2\}$	ϕ

2) Construct an NFA to accept strings of a's & b's ending with ab



* Language accepted by NFA

Extended transition function of NFA to strings
The transition $\delta(q, a) \to P$ accepts two parameters namely state q and input symbol a and returns a set of state P given by
$P = \{p_1\ p_2\ p_3 \dots p_n\}$ which represents possible states of NFA at a given instant
The change of state from q to set of state P on an input string w is denoted using extended transition fn δ^* or $\hat{\delta}$ i.e
$\hat{\delta}$ defines what happens ~~on~~ ~~process~~ to state of machine on processing given input

recursively as below.

$\hat{\delta}(q,\varepsilon) \rightarrow \{q\}$ indicates that if the machine is in state q & reads no information then it is at same state q

Induction part:

Let $w = \alpha a$ where a is the last symbol of w & α is a remaining string of w

Let q be the current state & α is a string to be processed & after processing α the machine can be any one of the $\{p_1, p_2, p_3 \cdots p_n\}$ state i.e

$$\hat{\delta}(q,\alpha) = \{p_1, p_2, p_3 \ldots p_n\}$$

The transition from set $\{p_1 p_2 p_3 \cdots p_n\}$ on input symbol a can be given as below

$$\delta(\{p_1, p_2, p_3 \ldots p_n\}, a) = \delta(p_1, a) \cup \delta(p_2, a) \cup \delta(\ldots$$

$$\hat{\delta}(q, \alpha a) = \bigcup_{i=1}^{n} \delta(p_i, a)$$

Language accepted by NFA is defined as below

Let $M = \{Q, \Sigma, \delta, q_0, F\}$ be an NFA. A string w is accepted by machine M iff the string w take the start state q_0 to any of the accepting state

$L = \{w / \hat{\delta}(q_0, w) \cap F \neq \phi\}$ ie $L(M)$ is the set of string w in Σ^* such that $\hat{\delta}(q_0, w)$ contains atleast one accepting string

* Conversion from NFA to DFA

All NFA's can be converted in to its equavalent DFA using any of two methods.

1) Subset construction method
2) Lazy evaluation method

1) Subset construction method:

proove that there exist a DFA for any NFA.

Given an NFA $M_N = \{Q_N, \Sigma, \delta_N, q_0, F_N\}$ which accepts the language $L(M_N)$, an equivalent DFA $M_D = \{Q_D, \Sigma, \delta_D, q_0, F_D\}$ can be found such that $L(M_N) = L(M_D)$ using subset construction method as below.

S1: Identify the start state of DFA

If q_0 is the start state of NFA then, $\{q_0\}$ is the start state of equivalent DFA

S2: Identify input alphabets.

Input alphabet Σ of DFA are same as that of given NFA

S3: Identify the states of DFA (Q_D)

Set of all subsets ie powerset of Q_N will be the states of DFA. If Q_N has 'n' no of states then Q_D will have 2^n no of states.

S4: Identify the finalstates of DFA

If the set $\{q_i\ q_j \ldots q_l\}$ is a state in DFA and if any one of $\{q_i, q_j \ldots q_l\}$ is a ~~finite~~ final state of NFA then the subset $\{q_i, q_j \ldots q_l\}$ is a final state of DFA.

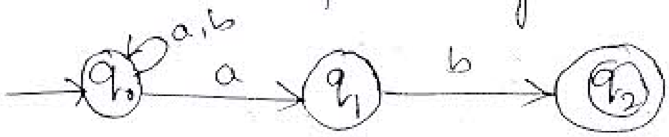
S5: Identify the transition of DFA (δ_D):

For each state $\{q_i, q_j \ldots q_l)$ including (Q_D) and each alphabet $a \in \Sigma$ the transition can be found by

$$\boxed{\delta_D(\{q_i, q_j \ldots q_l\}, a) \rightarrow \delta_N(q_i, a) \cup \delta_N(q_j, a) \cup \ldots \ldots \delta_N(q_l, a)}$$

Imp Q

1) Convert the following NFA to DFA



q_0
q_1
q_2

So

Sol?: Step 1: q_0 is the start state of the NFA & hence $\{q_0\}$ is the start state of DFA

S:

Step 2: $\Sigma = \{a, b\}$

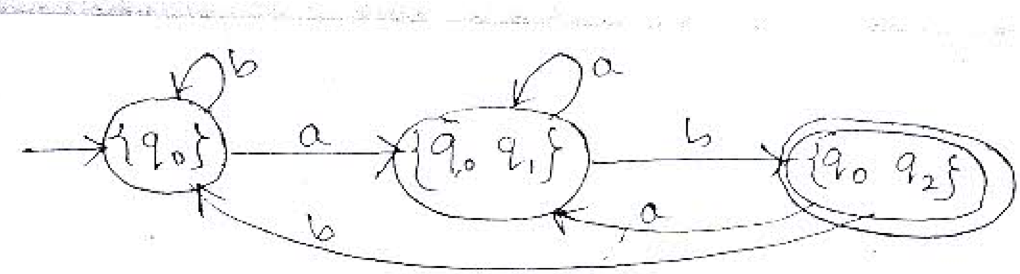
Step 3: $Q_D = \{\phi, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0 q_1\}, \{q_0, q_2\} \{q_1 q_2\} \{q_0, q_1, q_2\}\}$

Step 4: final states $\{q_2\}, \{q_0, q_2\} \{q_1 q_2\} \{q_0 q_1 q_2\}$

Step 5: find the transition table

	a	b
ϕ	ϕ	ϕ
$\{q_0\}$	$\{q_0 q_1\}$	$\{q_0\}$
$\{q_1\}$	ϕ	$\{q_2\}$
$\{q_2\}$	ϕ	ϕ
$\{q_0 q_1\}$	$\{q_0 q_1\}$	$\{q_0 q_2\}$
$\{q_0 q_2\}$	$\{q_0 q_1\}$	$\{q_0\}$
$\{q_1 q_2\}$	ϕ	$\{q_2\}$
$\{q_0 q_1 q_2\}$	$\{q_0 q_1\}$	$\{q_0 q_2\}$

δ_N	a	b
q_0	$\{q_0 q_1\}$	$\{q_0\}$
q_1	ϕ	$\{q_2\}$
q_2	ϕ	ϕ



Imp

Q) Convert the following NFA to DFA using Subset construction method.

Soln:

	0	1
→ p	{p, q}	{p}
q	{r}	{r}
r	{s}	ϕ
s*	{s}	{s}

Soln:

S1 : p is the start state

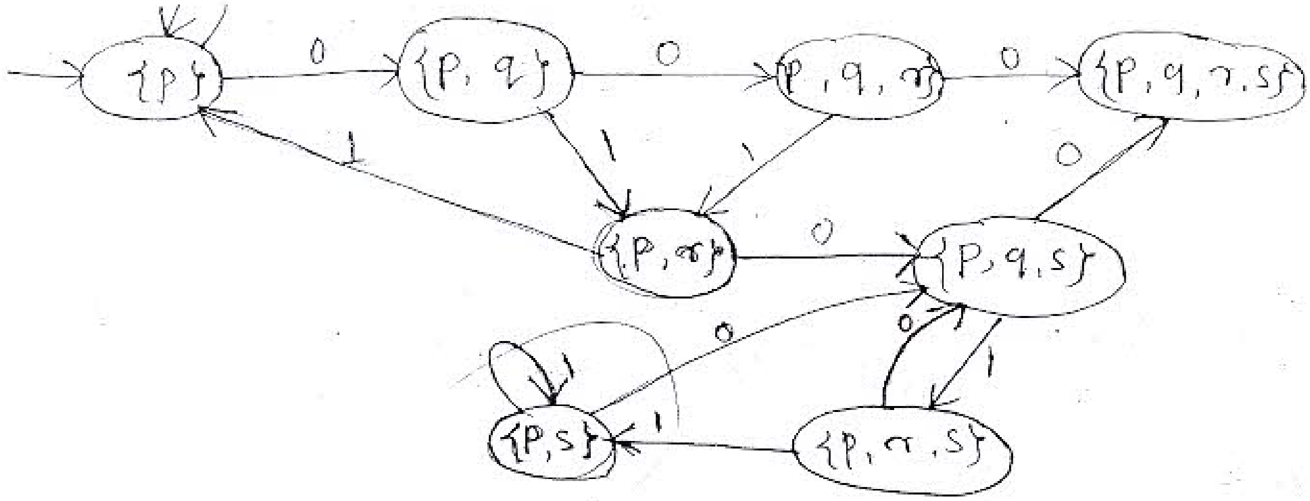
S2 : $\Sigma = \{0, 1\}$

S3 : {ϕ, {p}, {q}, {r}, {s}, {p, q} {p r} {p, s} {q r} {q, s} {r, s} {p, q, r, s} {p, q, r} {p, q, s} {q, r, s} {p, r, s}

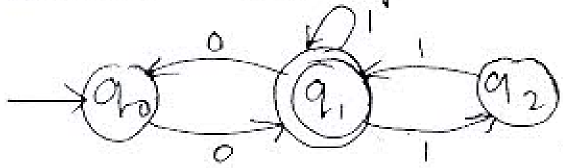
S4 : Final state : {{s} {p s} {q, s} {r, s} {p, q, r, s} {p, q, s} {q, r, s} {p, r, s}

S5 : Transition table

	0	1
ϕ	ϕ	ϕ
→ {p}	{p, q}	{p}
{q}	{r}	{r}
{r}	{s}	ϕ
{s}	{s}	{s}
{p, q}	{p, q, r}	{p, r}
{p r}	{p, q, s}	{p}
{p s}	{p, q, s}	{p, s}
{q r}	{r, s}	{r}
{q s}	{r, s}	{r, s}
{r s}	{s}	{s}
{p q r}	{p, q, r, s}	{p, r}
{p q s}	{p, q, r, s}	{p, r, s}
{q r s}	{r, s}	{r, s}
{p r s}	{p, q, s}	{p s}
{p, q, r, s}	{p, q, r, s}	{p, r, s}



3) convert the given NFA in to DFA



Sol) S1 : q_0 is the start state

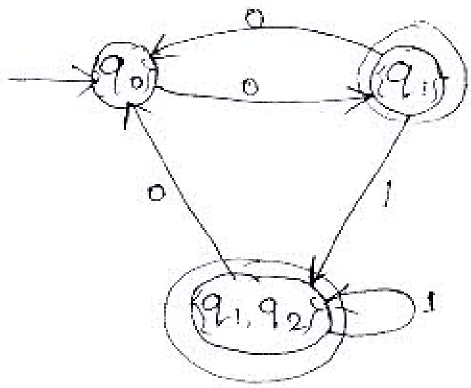
S2 : $\Sigma = \{0, 1\}$

S3 = $\{\phi, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0 q_1\} \{q_0 q_2\} \{q_1, q_2\}, \{q_0, q_1, q_2\}\}$

S4 = $\{\{q_1\}, \{q_0 q_1\} \{q_1, q_2\} \{q_0 q_1, q_2\}\}$

S5

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



Lazy Evaluation Method

Given an NFA $M_N = \{Q_N, \Sigma, \delta_N, q_0, F_N\}$ which accepts the language $L(M_N)$ we can find an equalend DFA

$M_D = \{Q_D, \Sigma, \delta_D, \{q_0\}, F_D\}$ Such that $L(M_D) = L(M_N)$ using Lazy evaluation method as below

S1: Identify the start state

If q_0 is the start state of NFA then $\{q_0\}$ is the start state of DFA add set $\{q_0\}$ to Q_D.

S2: Identify the alphabets.

Σ is same for both

S3: Identify the transition

for each state $\{q_i, q_j \dots q_k\}$ in Q_D & for each input alphabet $a \in \Sigma$ transition can be obtained as below.

$$\delta_D(\{q_i, q_j \dots q_k\}, a) = \delta_N(\{q_i, a) \cup \delta_N(q_j, a) \cup \dots \cup \delta_N(q_k, a)$$

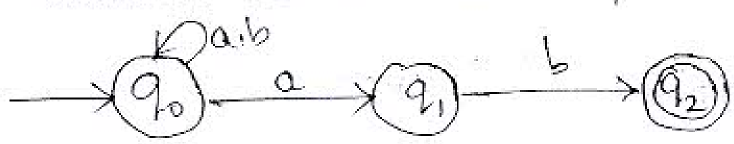
assume the transition leads to a new state $\{q_l, q_m \dots q_n\}$ add the new state to Q_D and repeat the step 3 for each ~~step~~ new state added to Q_D.

identify the final state

If the state $\{q_i, q_j \ldots q_k\}$ has any one of the final state in it then it is a final State of DFA.

Problems

1) Obtain DFA for following NFA.



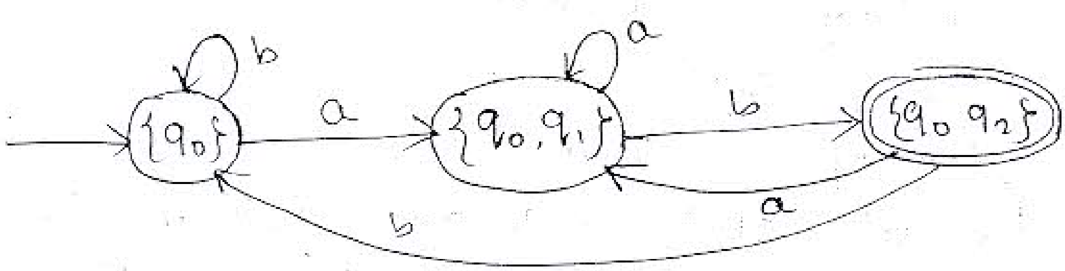
Sol: S1: q_0 is the start state in NFA $= q_0$

DFA $= \{q_0\}$

S2: $\Sigma = \{a, b\}$

S3:

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

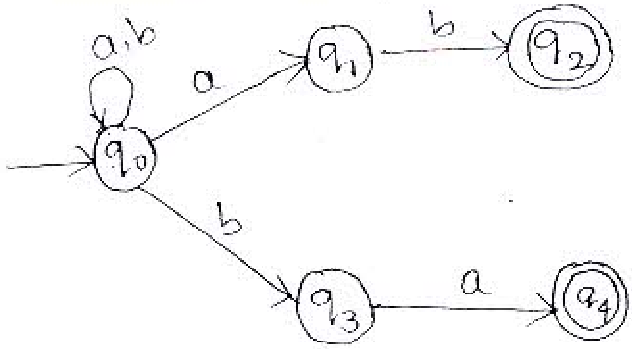


Imp

2) Obtain a NFA to accept strings of a's & b's ending with ab or ba & convert the NFA in to DFA

Sol

3)

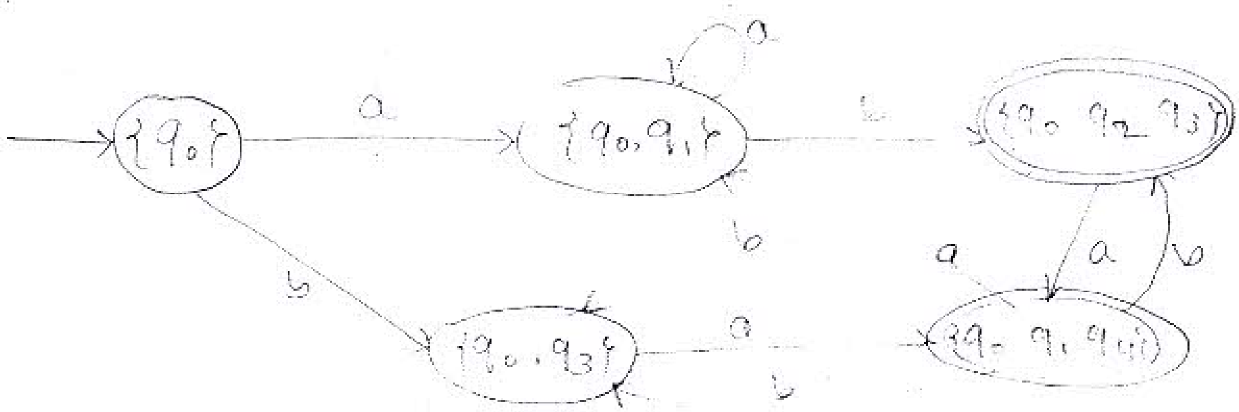


S1 : q_0 is the start state of NFA.
$\{q_0\}$ is the start state of DFA.

S2 = $\Sigma = \{a, b\}$

S3 =

	a	b
$\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_0, q_3\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2, q_3\}$
$\{q_0, q_3\}$	$\{q_0, q_1, q_4\}$	$\{q_0, q_3\}$
$\{q_0\ q_2\ q_3\}^*$	$\{q_0, q_1, q_4\}$	$\{q_0, q_3\}$
$\{q_0\ q_1\ q_4\}^*$	$\{q_0, q_1\}$	$\{q_0, q_2, q_3\}$



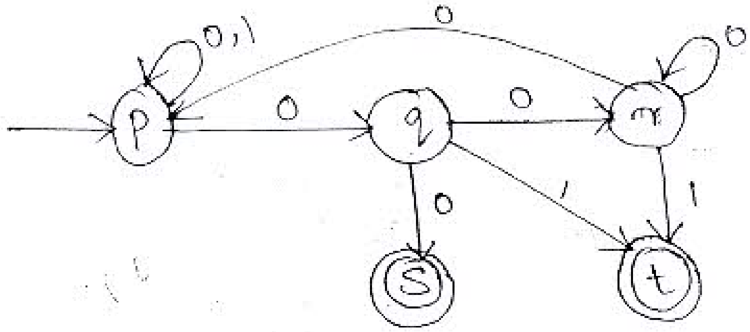
i's

3) Convert the following NFA to DFA & process the string 101 using extended transition on obtained DFA

Soln: given

N		
→ p	{p,q}	{p}
q	{r,s}	{t}
r	{p,r}	{t}
s *	φ	φ
t *	φ	φ

Sol:

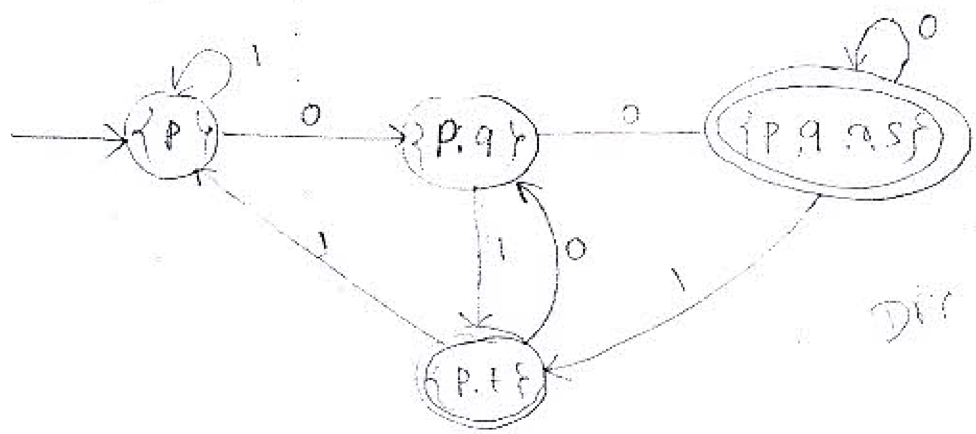


S1. p is the start state.

S2 Σ = {0,1}

S3.

	0	1
→ {p}	{p,q}	{p}
{p,q}	{p,q,r,s}	{p,t}
{p,t} *	{p,q}	{p}
{p,q,r,s} *	{p,q,r,s}	{p,t}



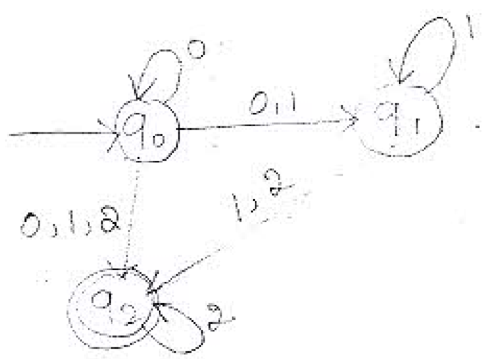
Extended transition

$$\hat{\delta}(p, 101) = \delta(\hat{\delta}(p, 10), 1)$$
$$= \delta(\delta(\hat{\delta}(p, 1), 0), 1)$$
$$= \delta(\delta(\delta(\hat{\delta}(p, \epsilon), 1), 0), 1)$$
$$= \delta(\delta(\delta(p, 1), 0), 1)$$
$$= \delta(\delta(p, 0), 1)$$
$$= \delta(p, q, 1)$$
$$= p$$

4) Convert the following NFA to DFA.

	0	1	2
$\rightarrow q_0$	$\{q_0, q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
q_1	ϕ	$\{q_1, q_2\}$	$\{q_2\}$
q_2	ϕ	ϕ	$\{q_2\}$

Soln :

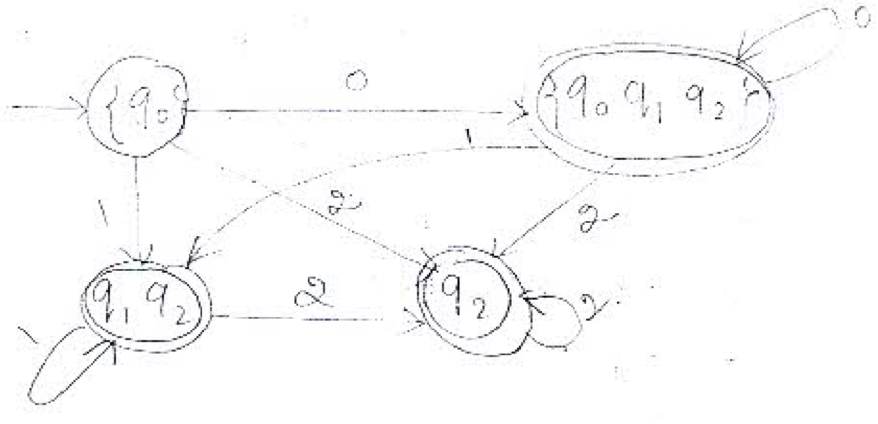


S1 : q_0 is the start state

S2 : $\Sigma = \{0, 1, 2\}$

S3 :

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



20/8/16 1) Define following:

i) Power of an alphabet ii) NFA iii) ~~Des~~

2) Design dfa to accept the following languages over the alphabet (10m)

$L = \{w / w$ is a even number$\}$

$L = \{0^{i+} (01)^i 1^{2i} \mid i \geq 1, j \geq 1\}$

String starts with 01 or ends with 01

3) Define a DFA. construct a DFA to accept a String of even number of Zero's and odd number of 1's. (10m)

4) Define following with ex
String, alphabet, powerset, language

5) Design a DFA ~~cons~~' to accept the binary no divisible by 5.

6) Convert the following NFA to its equalent DFA using subset construction.

	0	1
p	{p, q}	{p}
q	{r}	{r}
r	{s}	ϕ
s	{s}	{s}

7) Write a DFA for the languages (6m)
1) Set of all strings not containing 110 as its subset
2) String ~~of~~ with exactly 3 consicutive Zeros.

8) Convert the following NFA to DFA.

δ	0	1
q_0	q_0	$q_0 q_1$
q_1	q_2	q_2
q_2	ϕ	ϕ

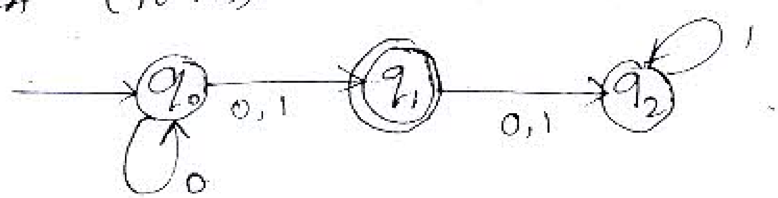
9) Proove that if $D = \{Q_D, \Sigma, \delta_D, \{q_0\}, F_D\}$ is the DFA constructed from $N = \{Q_N, \Sigma, \delta_N, \{q_0\}, F_N\}$ by subset construction then $L(D) = L(N)$. (8M)

10) Obtain a DFA to accept the language
$L\{w \mid w \in (a+b)^*, \begin{array}{l} n_a(w) \bmod 3 = 2 \\ n_b(w) \bmod 2 = 1 \end{array}\}$ (8m)

11) Obtain an NFA to accept language consist a's b's ending with ab or ba construct equivalent DFA.

12) Show that the language
$L = \{awa \mid w \in (a+b)^*\}$ is regular

13) Give the procedure to convert NFA to DFA & convert the NFA shown below to its equavalent DFA (10 m) I



1

Imp

ε - NFA

Definition : It is a quintouple or 5-touple $M = \{Q, \Sigma, \delta, q_0, F\}$ where

M = Name of ε-NFA

Q = Non empty finite set of state.

Σ = alphabets

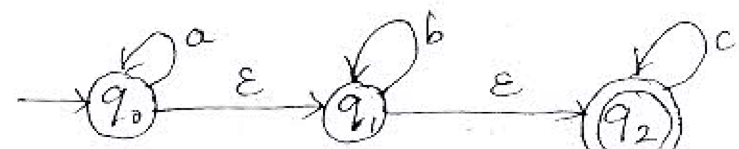
δ = $Q \times (\Sigma \cup \varepsilon) \rightarrow 2^Q$ fn or mapping

On a current state that can be transition to other state with or without any input symbols

$q_0 \in Q$ start state

$F \subseteq Q$ set of terminal states

1)



δ	a	b	c	ε
q_0	q_0	ϕ	ϕ	q_1
q_1	ϕ	q_1	ϕ	q_2
q_2	ϕ	ϕ	q_2	ϕ

Imp

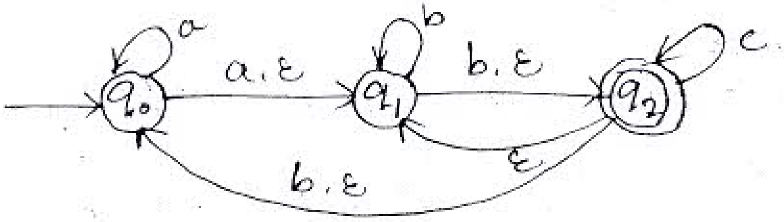
ε-closure or ~~ε-close~~ ECLOSE : ECLOSE of state q is denoted by ECLOSE(q) is the set of all states reachable from state q only through ε-transition and it is recursively defined as below.

1) The state q ~~is~~ in itself is in ECLOSE(q)

2) If ECLOSE(q) contain a state p and if there is transition from p to r on ε then the state r is also in ECLOSE(q)

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

1) Write the ECLOSE of all state in given ε-NFA



Soln

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

Conversion from ε-NFA to DFA

Let $M_E = \{Q_E, \Sigma, \delta_E, q_0, F_E\}$ be an ε-NFA we can find an equivalent DFA say $M_D = \{Q_D, \Sigma, \delta_D, q_0, F_E\}$ such that $L(M_E) = L(M_D)$ using following step

S1 : If q_0 is the start state of ε-NFA then, ECLOSE of q_0 be the start state of DFA

S2 : For any state in Q_D compute the transition ie if $\{q_i, q_j, \ldots q_k\}$ is the states of DFA then the transition for any input symbol can be calculated as below.

$$\delta_D(\{q_i, q_j \ldots q_k\}, a) = \delta_E(q_i, a) \cup \delta(q_j, a) \ldots \cup \delta_E(q_k, a)$$
$$= \{P_1, P_2, P_3 \ldots P_n\}$$
$$= \text{ECLOSE}(\{P_1 P_2 P_3 \ldots P_n\})$$

$= ECLOSE(P_1) \cup ECLOSE(P_2) \ldots \cup ECLOSE(P_n)$

S3: Identify the final state.

If this state contain atleast 1 final state of ε-NFA then it is final state of DFA

1) Convert the following ε-NFA to DFA.

Soln:

S1: $ECLOSE(q_0) = \{q_0, q_1, q_2\}$.

$ECLOSE(q_1) = \{q_1, q_2\}$.

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

Since q_0 is the start state of ε-NFA, the ECLOSE of q_0 i.e $\{q_0, q_1, q_2\}$ is the start state of DFA

S2: $\delta_D(\{q_0, q_1, q_2\}, a) = \{q_0\}$ ~~ECLOSE(q_0) = {q_0,~~

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

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

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

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

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

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

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

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

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

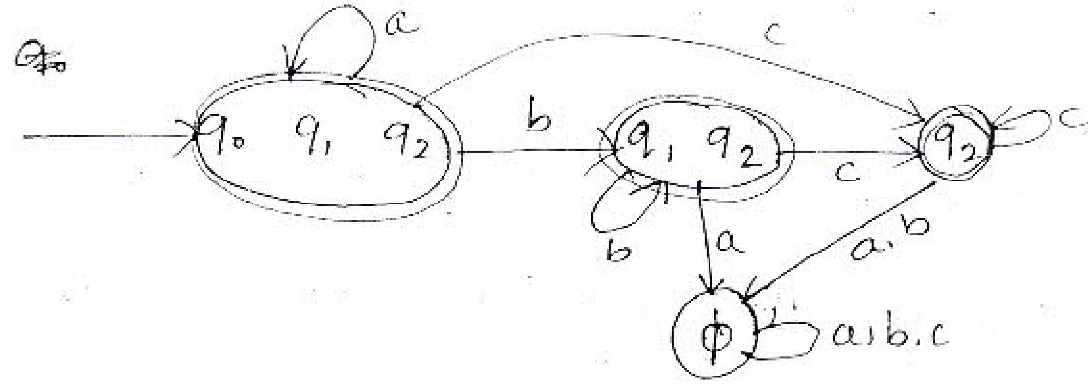
$\delta_D(\{q_1, q_2\}, c) = \{q_2\} = ECLOSE(q_2) = \{q_2\}$

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

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

$\delta_D(\{q_2\}, c) = \{q_2\}$ $ECLOSE(q_2) = \{q_2\}$.

$\therefore Q_D = \{\{q_0, q_1, q_2\}, \{q_1, q_2\}, \{q_2\}, \phi\}$



2) Convert ε-NFA to DFA

δ	ε	a	b	c
→ p	φ	{p}	{q}	{r}
q	{p}	{q}	{r}	φ
r *	{q}	{r}	φ	{p}

Sol:

~~ECLOSE (p) = {φ p, ε~~

S1 ECLOSE(p) = {p, φ}
ECLOSE(q) = {q, p, φ}
ECLOSE (r) = {r, q, p, φ}

Since p is start state of ε-NFA hence ECLOSE (p₀) is the start state of DFA
i.e. {p, φ} $\therefore Q_D = \{\{p, φ\}\}$

S2: $\delta_D(\{p, \phi\}, a) = \{p\}$
$\delta_D(\{p, \phi\}, b) = \{q\}$ = ECLOSE (q, p)
$\delta_D(\{p, \phi\}, c) = \{r\}$ = ECLOSE (r, q, p)

Imp ** 2

$\therefore Q_D = \{\{p\}, \cancel{\{q\}, \{r\}}\ \{q,p\}\ \{r,p,q\}\}$

$\delta_D(\{q,p\}, a) = \{q,p\} = ECLOSE(p,q) = \{q,p\}$

$\delta_D(\{q,p\}, b) = \{r,q\} = ECLOSE(rq) = \{r,q,p\}$

$\delta_D(\{q,p\}, c) = \{r\} = ECLOSE(r) = \{r,q,p\}$

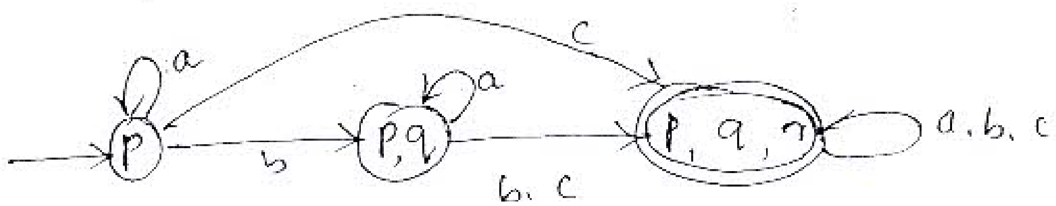
$\therefore Q_D = \{\{p\}, \{q,p\}\ \{r,p,q\}\}$

$\delta_D(\{r,p,q\}, a) = \{p,q,r\}\quad Eclose(p,q,r) = \{p,q,r\}$

$\delta_D(\{r,p,q\}, b) = \{q,r\}\quad Eclose(q,r) = \{p,q,r\}$

$\delta_D(\{p,q,r\}, c) = \{p,r\}\quad ECLOSE(p,r) = \{p,q,r\}$

$\therefore Q_D = \{\{p\}, \{q,p\}, \{r,p,q\}, \Phi\}.$

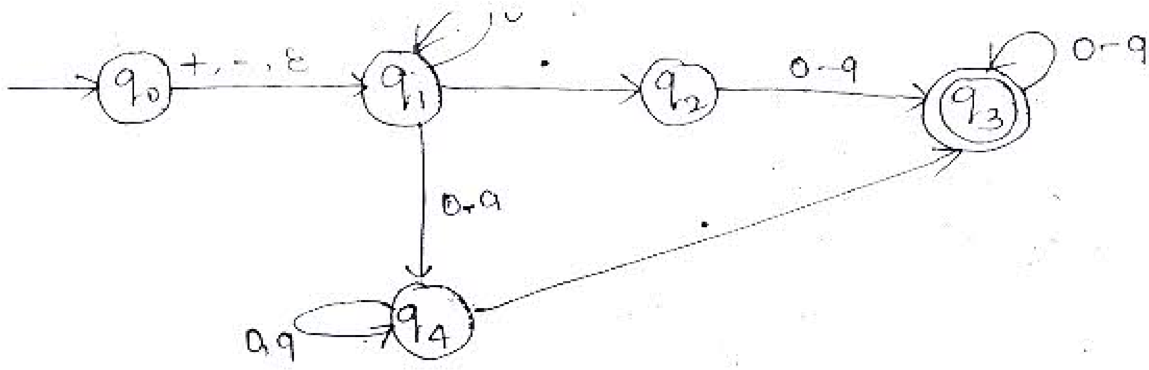


Imp 3) ** Design an ε-NFA which accepts decimal numbers consisting of

i) An optional ± sign
ii) A string of digits (0-9)
iii) A decimal point
iv) Another string of digit

either first or 2nd string of digits can be empty but atleast one of them should be nonempty. Convert the ε NFA obtained to DFA.

Soln:



S1:

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

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

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

$ECLOSE(q_3) = \{q_3\}$

$ECLOSE(q_4) = \{q_4\}$

Since q_0 is the start state of ε-NFA. ECLOSE(q_0) is the start state of DFA.

$\therefore Q_0 = \{q_0, q_1\}$

S2: $\delta_D(\{q_0, q_1\}, +) = \{q_1, \phi\} \Rightarrow ECLOSE\{q_1, \phi\} = \{q_1\}$

$\delta_D(\{q_0, q_1\}, -) = \{q_1, \phi\} \Rightarrow ECLOSE\{q_1, \phi\} = \{q_1\}$

$\delta_D(\{q_0, q_1\}, 0\text{-}9) = \{\phi, q_1\} \Rightarrow ECLOSE\{q_1, \phi\} = \{q_1\}$

$\delta_D(\{q_0, q_1\}, \cdot) = \{\phi, q_2\} \Rightarrow ECLOSE(\phi, q_2) = \{q_2\}$

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

$\delta_D(\{q_1\}, +) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_1\}, -) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_1\}, 0\text{-}9) = \{q_1, q_4\} \Rightarrow ECLOSE(q_1, q_4) = \{q_1, q_4\}$

$\delta_D(\{q_1\}, \cdot) = \{q_2\} \Rightarrow ECLOSE(q_2) = \{q_2\}$

$\therefore Q_D = \{\{q_0, q_1\}, \{q_1\}, \{q_2\}, \{q_1, q_4\}\}$

$\delta_D(\{q_2\},+) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_2\},-) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_2\},0\text{-}9) = \{q_3\} \Rightarrow ECLOSE(q_3) = \{q_3\}$

$\delta_D(\{q_2\},\cdot) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\therefore Q_D = \{\{q_0,q_1\},\{q_1\},\{q_2\},\{q_1,q_4\}\{q_3\}\}$

$\delta_D(\{q_1,q_4\},+) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_1,q_4\},-) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D\{(q_1,q_4\},0\text{-}9) = \{q_1,q_4\} \Rightarrow ECLOSE(q_1,q_4) = \{q_1,q_4\}$

$\delta_D\{(q_1,q_4\},\cdot) = \{q_2,q_3\} \Rightarrow ECLOSE(q_2,q_3) = \{q_2,q_3\}$

$\therefore Q_D = \{\{q_0,q_1\},\{q_1\},\{q_2\},\{q_1,q_4\}\{q_3\},\{q_2,q_3\}\}$

$\delta_D(\{q_3\},+) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_3\},-) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_3\},0\text{-}9) = \{q_3\} \Rightarrow ECLOSE(q_3) = \{q_3\}$

$\delta_D(\{q_3\},\cdot) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\therefore Q_D = \{\{q_0q_1\},\{q_1\},\{q_2\},\{q_1,q_4\},\{q_3\},\{q_2,q_3\}\}$

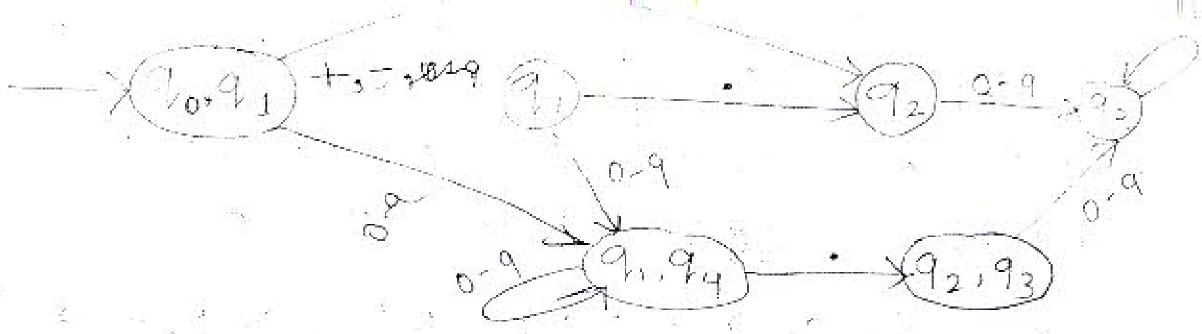
$\delta_D(\{q_2,q_3\},+) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{q_2,q_3\},-) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

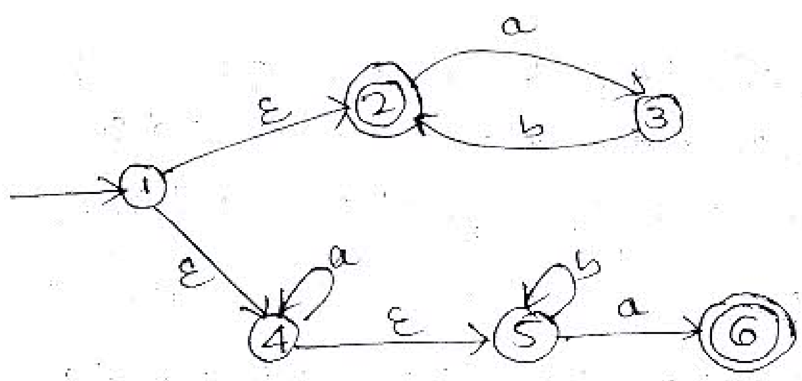
$\delta_D(\{q_2,q_3\},0\text{-}9) = \{q_3\} \Rightarrow ECLOSE(q_3) = \{q_3\}$

$\delta_D(\{q_2,q_3\},\cdot) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\therefore Q_D = \{\phi,\{q_1,q_4\}\{q_0,q_1\}\{q_1\}\{q_2\}\{q_3\}\{q_2,q_3\}\}$



4)



Sol:

S1: ECLOSE(1) = {1,2,4,5}

ECLOSE(2) = {2}

ECLOSE(3) = {3}   ECLOSE(6) = {6}.

ECLOSE(4) = {4,5}

ECLOSE(5) = {5}

Since 1 is the initial state ECLOSE(1) is the initial state

$Q_D = \{\{1,2,4,5\}\}$

S2: $\delta_D(\{1,2,4,5\},a) = \{3,4,6\} \Rightarrow ECLOSE(3,4,6) = \{3,5,4,6\}$

$\delta_D(\{1,2,4,5\},b) = \{5\} \Rightarrow ECLOSE(5) = \{5\}$.

$\cancel{\delta_D(\{1,2,4,5\}}$, $Q_D = \{\{1,2,4,5\},\{3,5,4,6\},\{5\}\}$

$\delta_D(\{3,5,4,6\},a) = \{6,4\} \Rightarrow ECLOSE(6,4) = \{4,5,6\}$

$\delta_D(\{3,5,4,6\},b) = \{2,5\} \Rightarrow ECLOSE(2,5) = \{2,5\}$

$\therefore \delta_D = \{\{1,2,4,5\},\{3,5,4,6\}\{5\},\{2,5\}\}$.

$\delta_D(\{5\}, a) = \{6\} \Rightarrow ECLOSE(6) = \{6\}$

$\delta_D(\{5\}, b) = \{5\} \Rightarrow ECLOSE(5) = \{5\}$

$\therefore Q_D = \{\{1,2,4,5\}, \{3,5,4,6\}, \{5\}, \{2,5\} \{6\}\}$

$\delta_D(\{2,5\}, a) = \{3,6\} \Rightarrow ECLOSE(3,6) = \{3,6\}$

$\delta_D(\{2,5\}, b) = \{5\} \Rightarrow ECLOSE(5) = \{5\}$

$\therefore Q_D = \{\{1,2,4,5\}, \{3,5,4,6\} \{5\}, \{2,5\} \{6\} \{3,6\}\}$

$\delta_D(\{6\}, a) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\delta_D(\{6\}, b) = \{\phi\} \Rightarrow ECLOSE(\phi) = \{\phi\}$

$\therefore Q_D = \{\{1,2,4,5\}, \{3,5,4,6\}, \{5\}, \{2,5\}, \{6\}, \{3,6\}\}$

$\delta_D(\{3,6\}, a) = \{\phi\} \Rightarrow ECLOSE(\phi) = \phi$

$\delta_D(\{3,6\}, b) = \{2\} \Rightarrow ECLOSE(2) = \{2\}$

$\therefore Q_D = \{\{1,2,4,5\}, \{3,5,4,6\}, \{5\} \{2,5\}, \{6\}, \{3,6\}, \{2\}\}$

$\delta_D(\{2\}, a) = \{3\} \Rightarrow ECLOSE(3) = \{3\}$

$\delta_D(\{2\}, b) = \{\phi\} \Rightarrow ECLOSE(\phi) = \phi$

$\therefore Q_D = \{$

$\delta_D(\{3\}, a) = \{\phi\} \Rightarrow ECLOSE(3) = \{3\}$

$\delta_D(\{3\}, b) = \{2\} \Rightarrow ECLOSE(2) = \{2\}$

$\Rightarrow Q_D = \{\{1,2,4,5\}, \{3,5,4,6\}, \{5\}, \{2,5\}, \{6\}, \{3,6\}, \{2\}, \{3\}\}$

$\}$ $\delta_D(\{2,3\}, a) = \{$

49