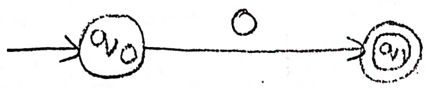
~~Examples~~

QUESTION AND ANSWERS

(1) Construct an NFA for the regular expression $0+01^*$

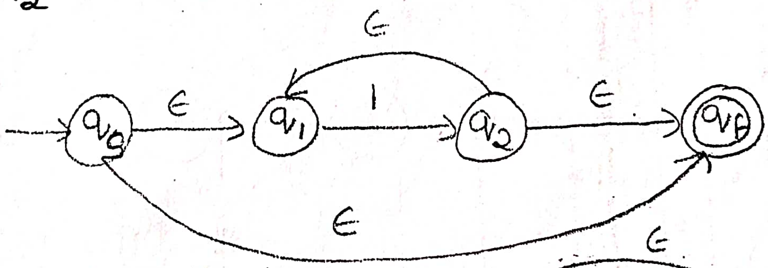
Sol:- Let us consider $r = \underbrace{0}_{r_1} + \underbrace{01^*}_{r_2}$ $(r = r_1 + r_2)$

The r.e of r_1 is

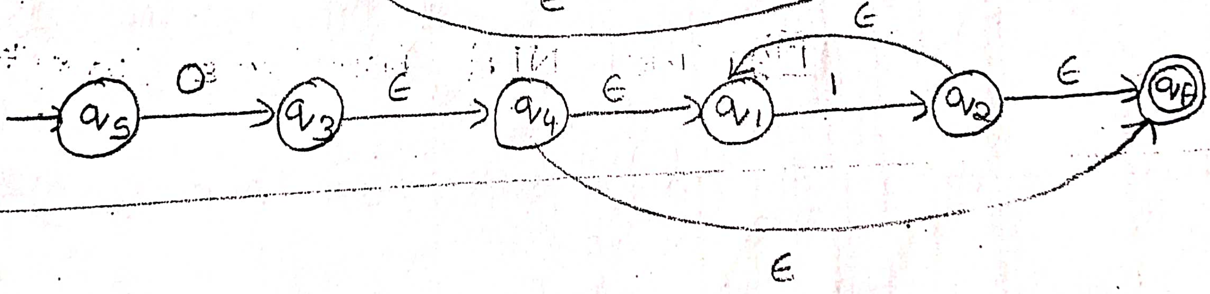


The r.e of r_2 is

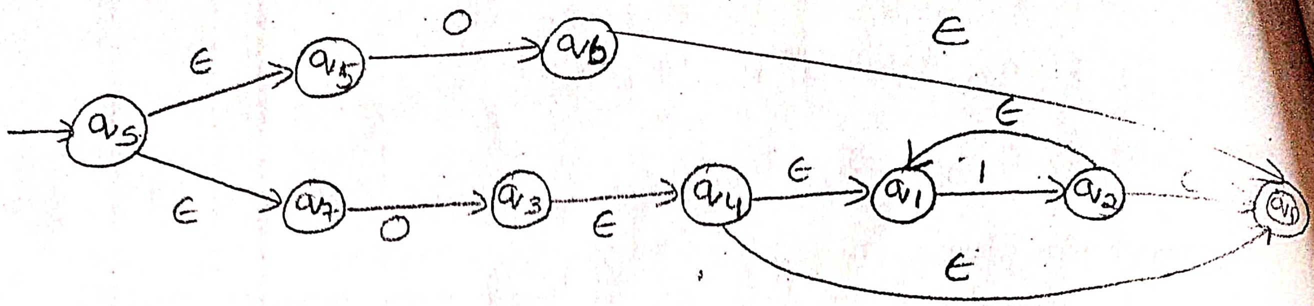
1^* is



01^* is

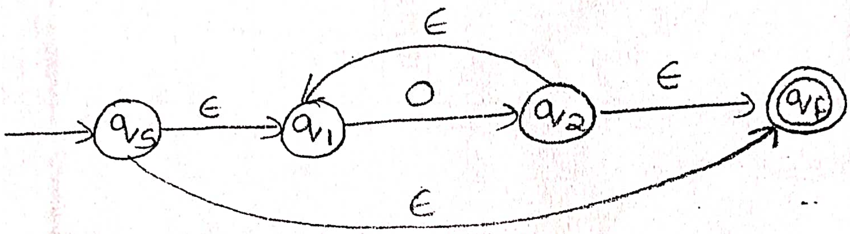


The r.E for r is:-

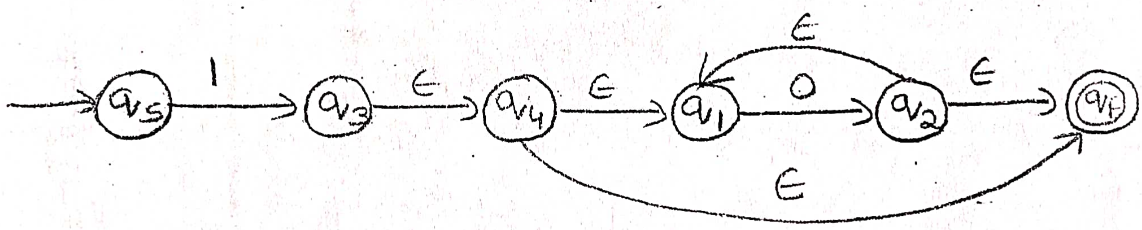


(2) Convert the r.E, $r = (10^*)^*$ into NFA with ∈

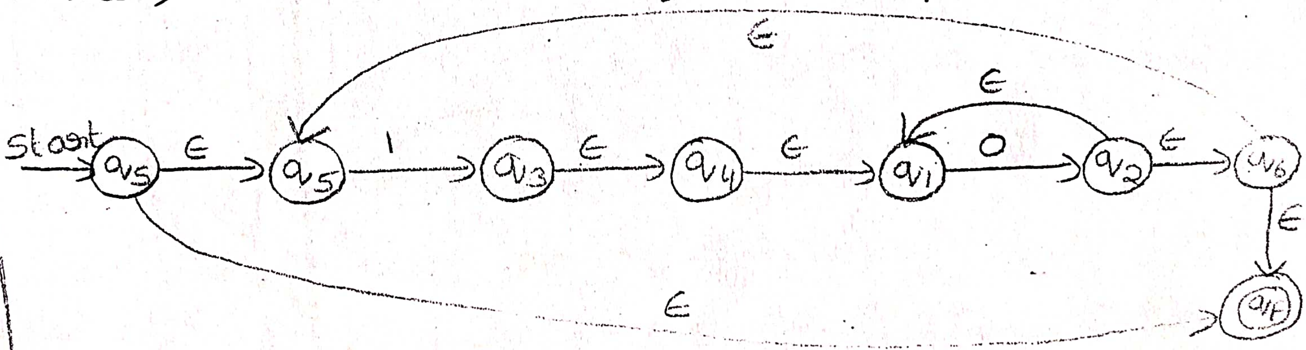
<u>Sol:-</u> Let us first construct r.E for 0^*



Now construct r.E for 10^* as $r_1 r_2$
r_1, r_2



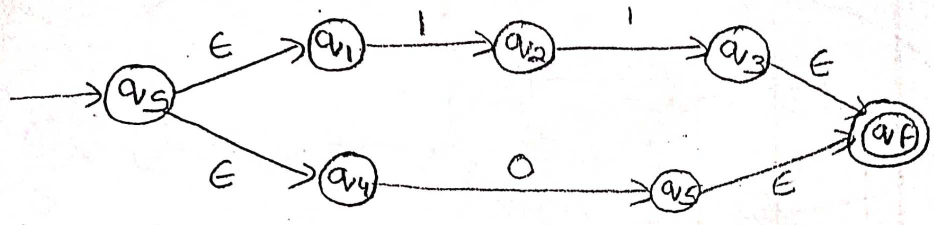
Atlast, consider $r = (10^*)^*$ as r_1^*



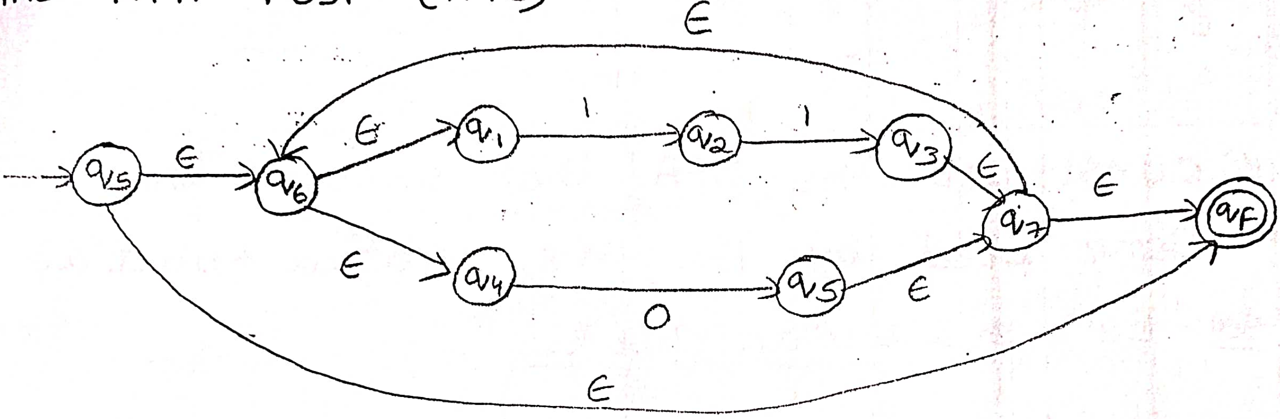
<u>Fig:-</u> The NFA for r.E $(10^*)^*$

Convert the r.e., $r = (11+0)^*(00+1)^*$ to ϵ-NFA (8)

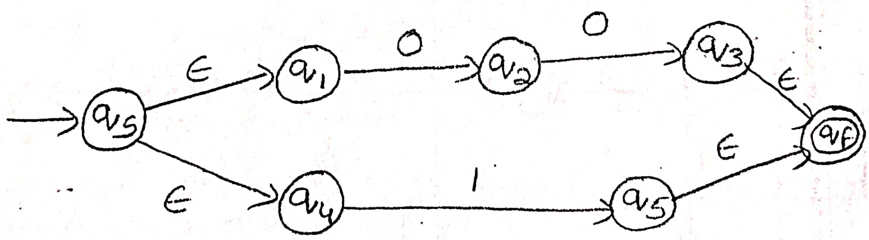
Sol: Let us first construct NFA for $(11+0)$



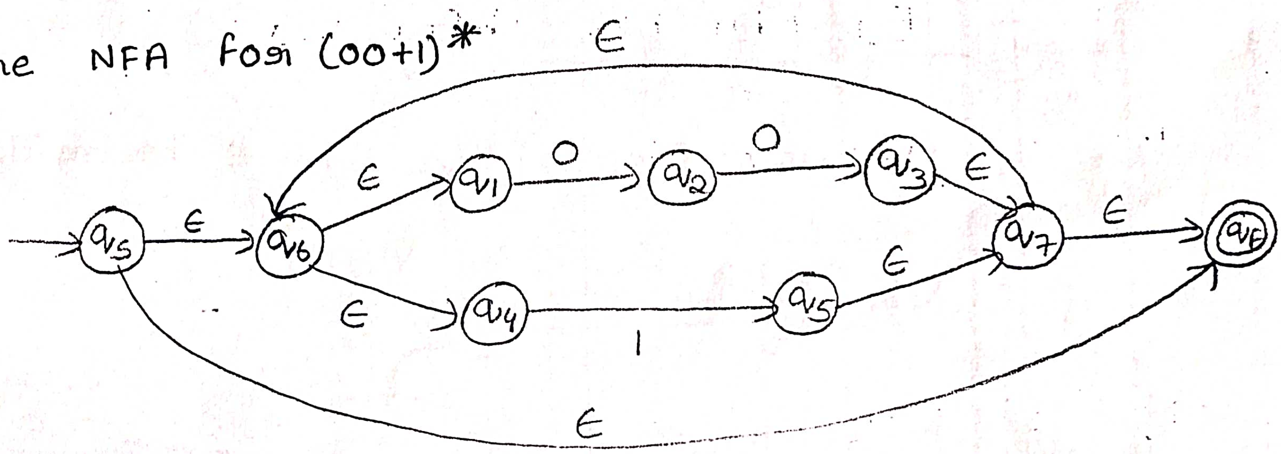
The NFA for $(11+0)^*$



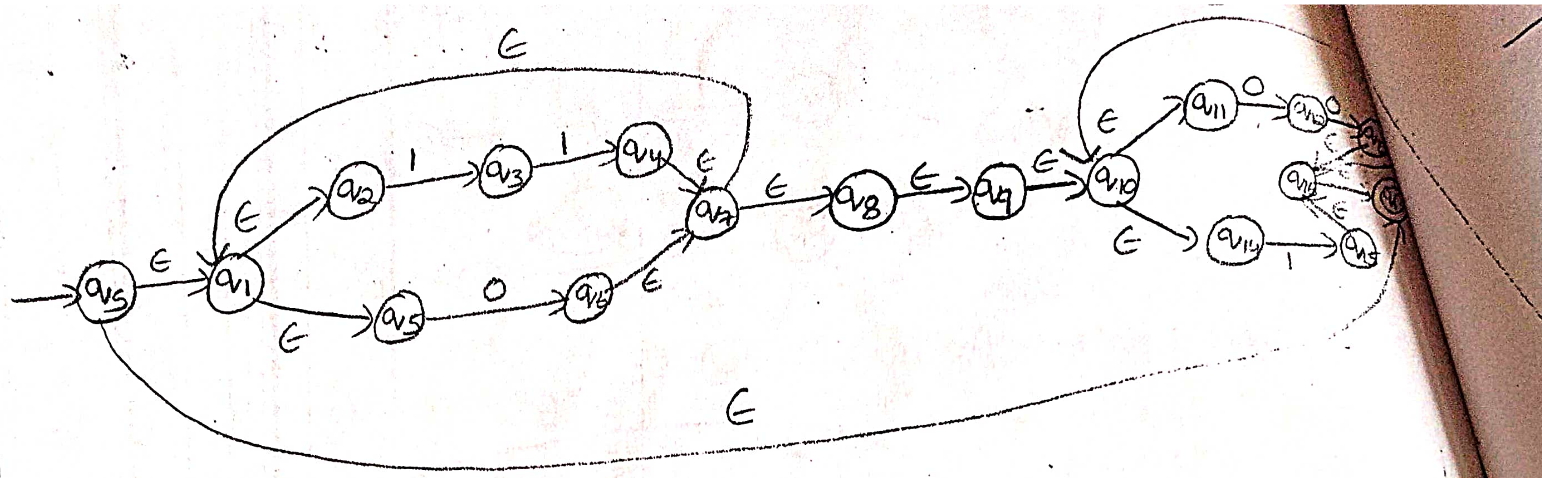
Again, consider $(00+1)$



The NFA for $(00+1)^*$



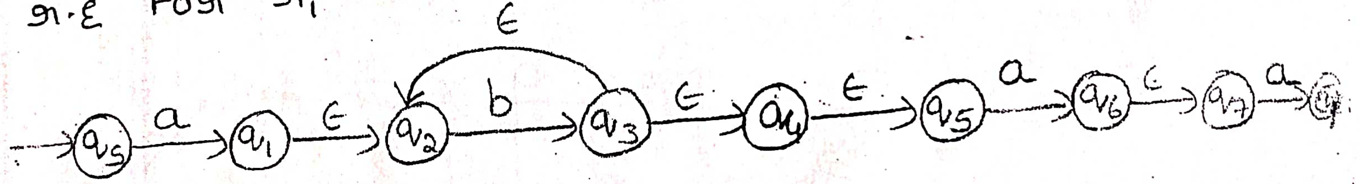
Now here, $r = \underbrace{(11+0)^*}_{r_1} \underbrace{(00+1)^*}_{r_2}$



(4) construct the NFA that accepts the language generated by the r.e, $ab^*aa + bba^*ab$

<u>Sol</u>:- Let, $r = \underbrace{ab^*aa}_{r_1} + \underbrace{bba^*ab}_{r_2}$

The r.e for r_1 is:-



By reduction, we have

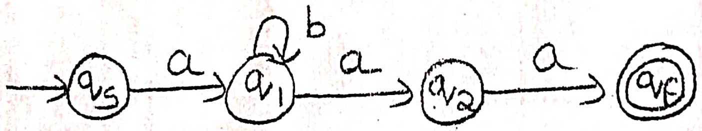
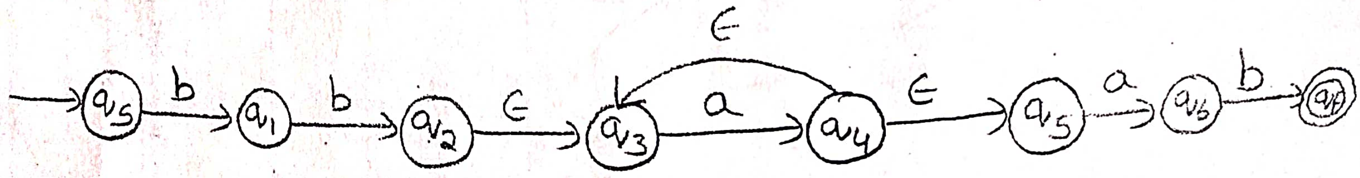


Fig:- NFA for ab^*aa w/o ϵ-transitions

The r.e for r_2 is:-



By reduction,

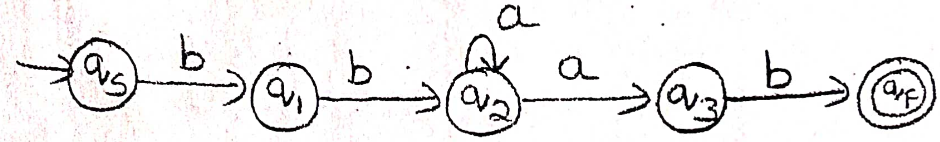


Fig:- NFA for bba^*ab w/o ϵ-transitions

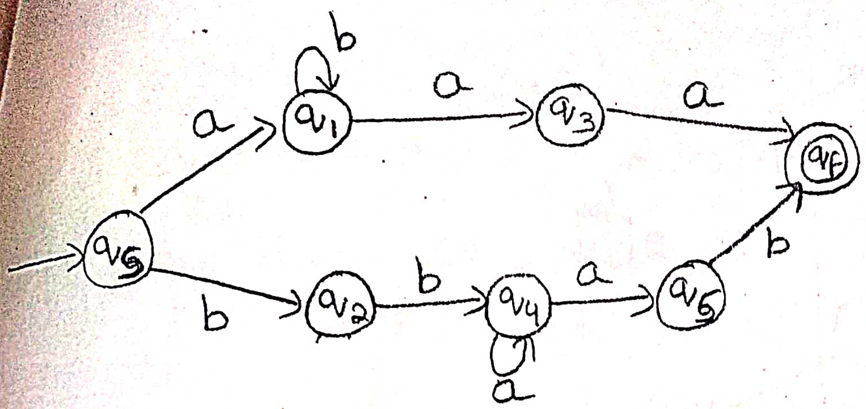
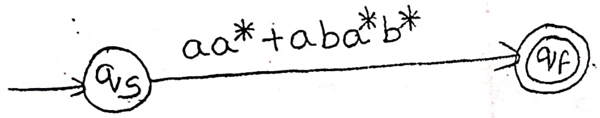


Fig:- NFA for $ab^*aa + bba^*ab$ w/o ε-transition

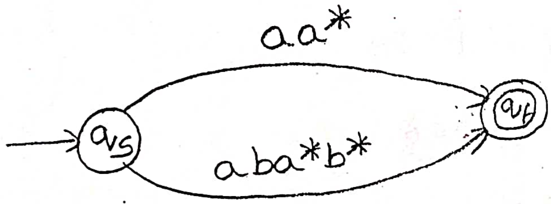
(5) Give the DFA for the r.e, $aa^* + aba^*b^*$

Sol:- Second way of construction:-

(1)

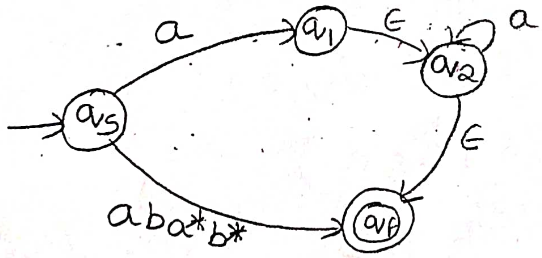


(2)



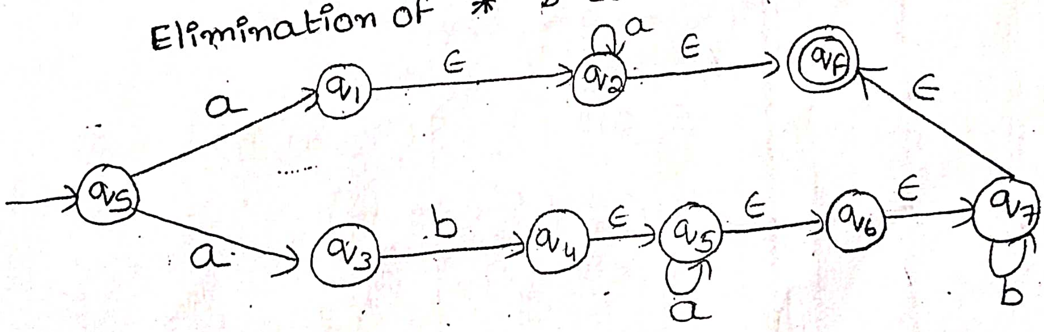
Elimination of '+'

(3)



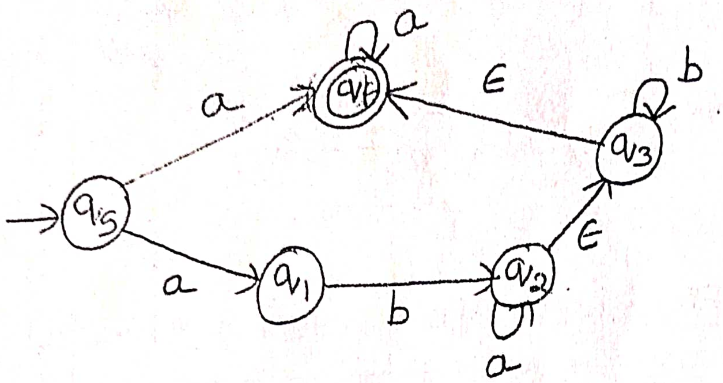
Elimination of * & concatenation

(4)



Elimination of *.

By reduction



NFA with ε-transitions

Inorder to convert it into NFA w/o ε-transitions

Find

$\epsilon\text{-clr}(q_s) = \{q_s\}$

$\epsilon\text{-clr}(q_1) = \{q_1\}$

$\epsilon\text{-clr}(q_2) = \{q_2, q_3, q_F\}$

$\epsilon\text{-clr}(q_3) = \{q_3, q_F\}$

$\epsilon\text{-clr}(q_f) = \{q_F\}$

Now, $\delta'(q_s, a) = \epsilon\text{-clr}(\delta'(\delta(q_s, \epsilon), a))$

$= \epsilon\text{-clr}(\delta'(q_s, a))$

$= \epsilon\text{-clr}(q_F) \cancel{\{q_F\}} \cup \epsilon\text{-clr}(q_1)$

$= \{q_1, q_F\}$

$\delta'(q_s, b) = \epsilon\text{-clr}(\delta'(\delta(q_s, \epsilon), b))$

$= \epsilon\text{-clr}(\delta'(q_s, b))$

$= \epsilon\text{-clr}(\emptyset) = \emptyset$

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

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

$= \epsilon\text{-clr}(\emptyset) = \emptyset$

$$\delta'(q_1, b) = \epsilon\text{-clo}(\delta'(\delta(q_1,\epsilon), b))$$
$$= \epsilon\text{-clo}(\delta'(q_1, b))$$
$$= \epsilon\text{-clo}(q_2) = \{q_2, q_3, q_f\}$$

$$\delta'(q_2, a) = \epsilon\text{-clo}(\delta'(\delta(q_2,\epsilon), a))$$
$$= \epsilon\text{-clo}(\delta'(\delta(q_2, q_f), a))$$
$$= \cancel{\epsilon\text{-clo}(\delta'(\emptyset)\cup\delta'} = \epsilon\text{-clo}(\delta'(q_2, a) \cup \delta(q_f, a))$$
$$= \epsilon\text{-clo}(\underset{q_2}{\emptyset} \cup q_f) = \{q_f\} \times$$
$$= \{q_2, q_3, q_f\}$$

$$\delta'(q_2, b) = \epsilon\text{-clo}(\delta'(\delta(q_2,\epsilon), b))$$
$$= \epsilon\text{-clo}(\delta'(q_2, q_3, q_4), b)$$
$$= \epsilon\text{-clo}(\delta(q_2, b) \cup \delta(q_3, b) \cup \delta(q_f, b))$$
$$= \epsilon\text{-clo}(\emptyset \cup q_3 \cup \emptyset) = \{q_3, q_f\}$$

$$\delta'(q_3, a) = \epsilon\text{-clo}(\delta'(\delta(q_3,\epsilon), a))$$
$$= \epsilon\text{-clo}(\delta'(q_3, q_f), a)$$
$$= \epsilon\text{-clo}(\delta(q_3, a) \cup \delta(q_f, a))$$
$$= \epsilon\text{-clo}(\emptyset \cup q_f) = \{q_f\}$$

$$\delta'(q_3, b) = \epsilon\text{-clo}(\delta'(\delta(q_3,\epsilon), b))$$
$$= \epsilon\text{-clo}(\delta'(q_3, q_f), b)$$
$$= \epsilon\text{-clo}(\delta(q_3, b) \cup \delta(q_f, b))$$
$$= \epsilon\text{-clo}(q_3 \cup \emptyset) = \{q_3, q_f\}$$

$$\delta'(q_f, a) = \epsilon\text{-clo}(\delta'(\delta(q_f,\epsilon), a))$$
$$= \epsilon\text{-clo}(\delta'(q_f, a))$$
$$= \epsilon\text{-clo}(q_f) = \{q_f\}$$

$$\delta'(q_f, b) = \epsilon\text{-clr}(\delta'(\delta(q_f, \epsilon), b))$$
$$= \epsilon\text{-clr}(\delta'(q_f, b))$$
$$= \epsilon\text{-clr}(\phi) = \phi$$

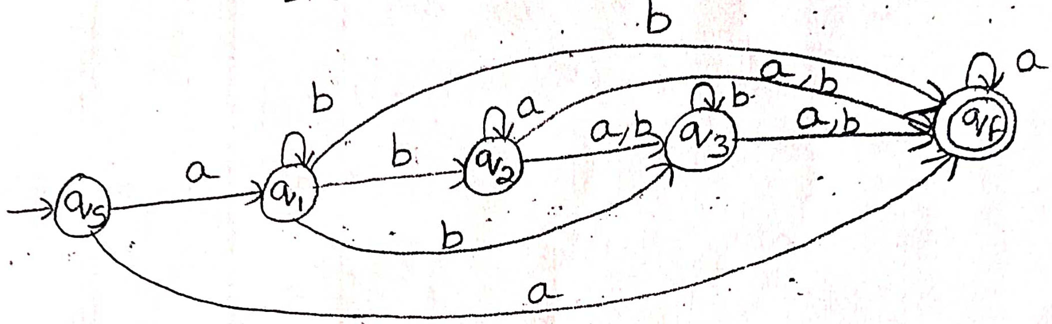


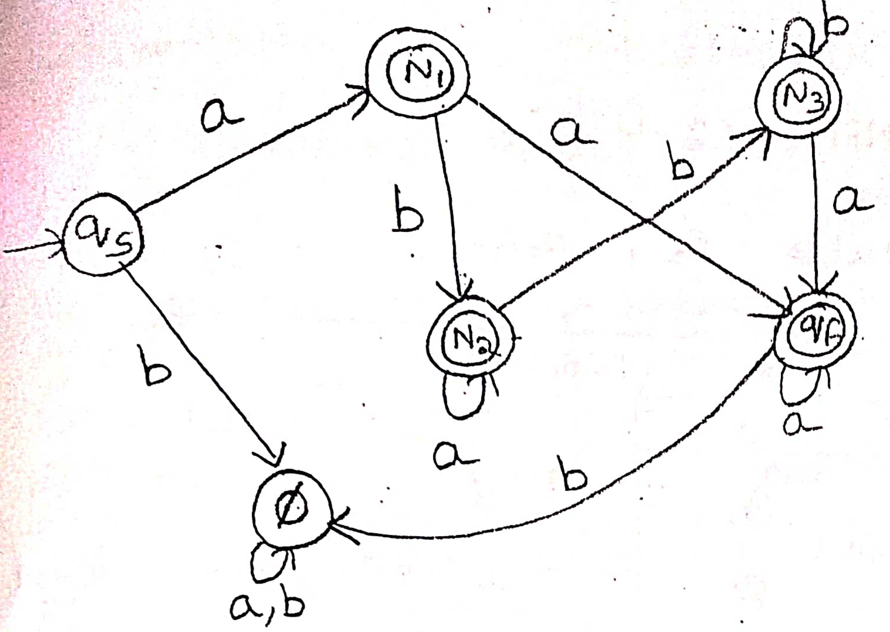
Fig:- NFA w/o ϵ-moves

Construction of DFA:-

states	inputs	
	a	b
$\rightarrow q_s$	$[q_1, q_f]$ N_1	ϕ
q_1	ϕ	$[q_2, q_3, q_f]$ N_2
q_2	$[q_2, q_3, q_f]$	$[q_3, q_f]$ N_3
q_3	$[q_f]$	$[q_3, q_f]$
q_f	$[q_f]$	ϕ

states	inputs	
	a	b
N_1	$[q_f]$	$[q_2, q_3, q_f]$ N_2
N_2	N_2	N_3
N_3	$[q_f]$	N_3

Basing on these transition table, we h
to construct DFA

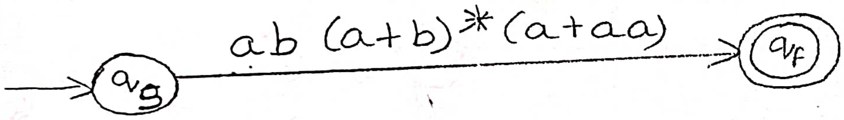


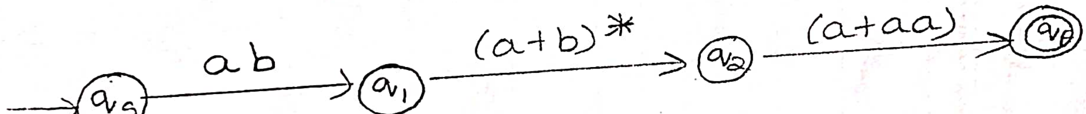
DFA

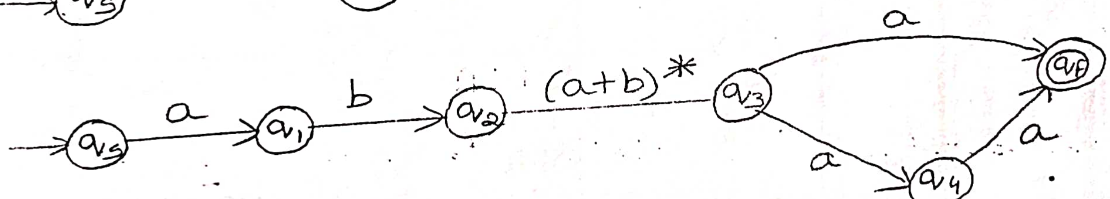
(6) Give the DFA for re ab (a+b)* (a+aa)

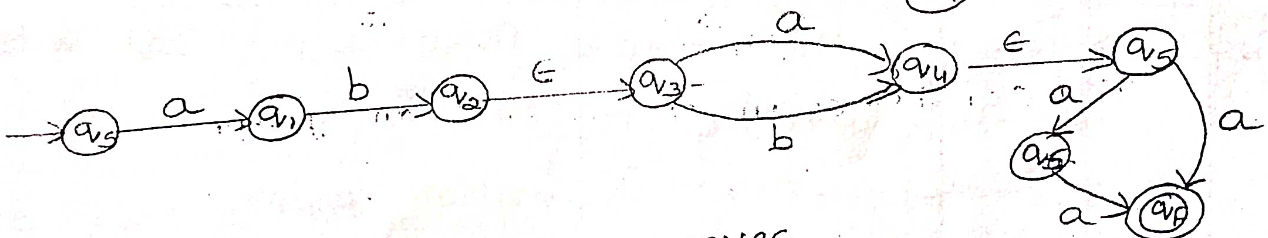
Sol:- The re is ab (a+b)*(a+aa), the case is

$r = r_1 r_2$

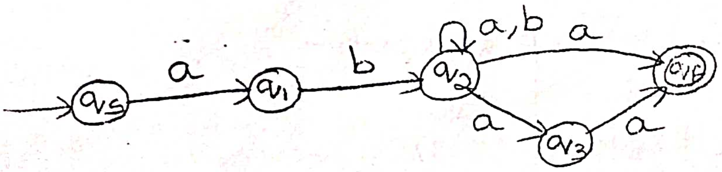
(1) 

(2) 

(3) 

(4) 

NFA with ε-moves

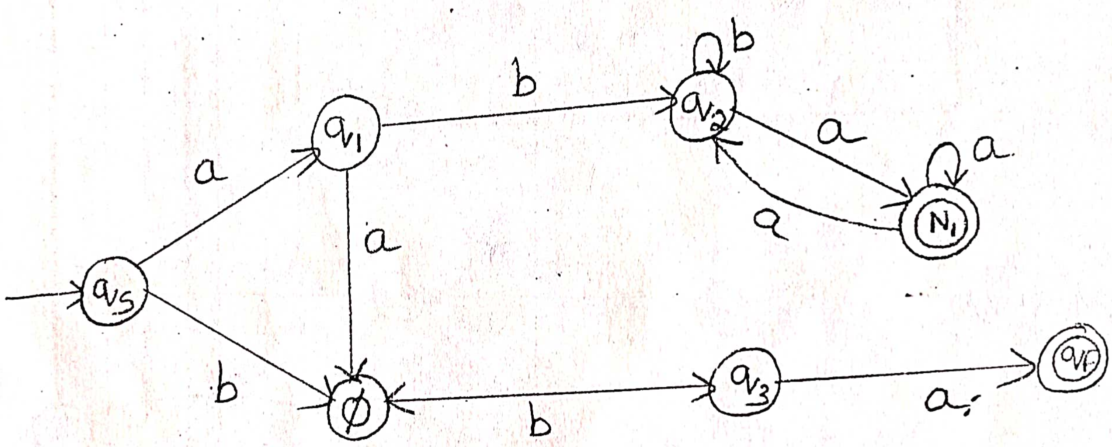
(5) 

NFA w/o ε-moves

Conversion of NFA to DFA:-

The transition table for NFA:-

States	Inputs a	Inputs b
$\rightarrow q_s$	q_1	ϕ
q_1	ϕ	q_2
q_2	$\underbrace{[q_2, q_3, q_F]}_{N_1}$	q_2
q_3	q_F	ϕ
q_F	ϕ	ϕ
N_1	N_1	q_2

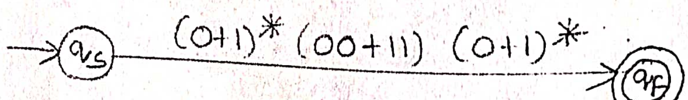


DFA

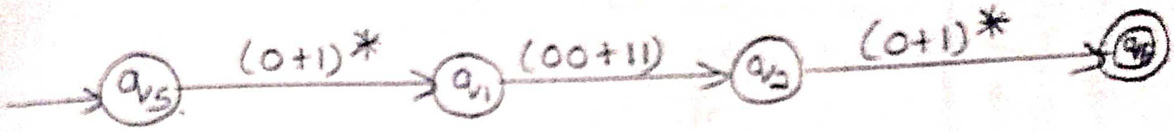
(7) Construct the Finite Automaton equivalent to the RE $(0+1)^*(00+11)(0+1)^*$

Sol:- (1) Construction of transition graph:-

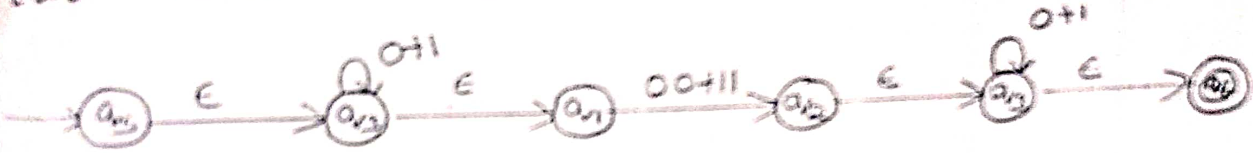
(a) First of all we construct the transition graph with ε-moves using Kleene's theorem



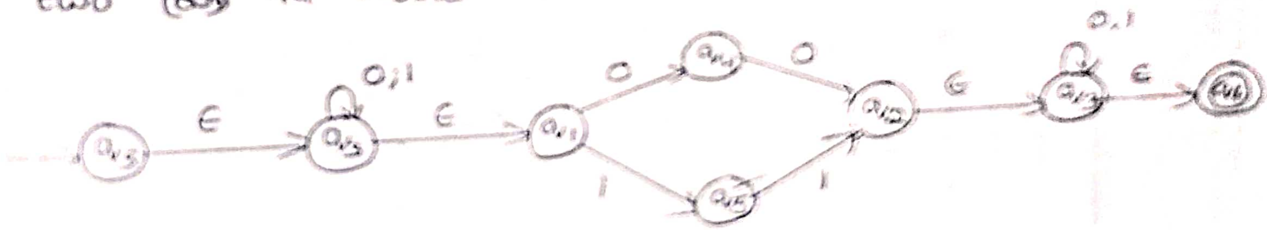
we eliminate the concatenations in the given RE (2)
by introducing new vertices q_1 & q_2



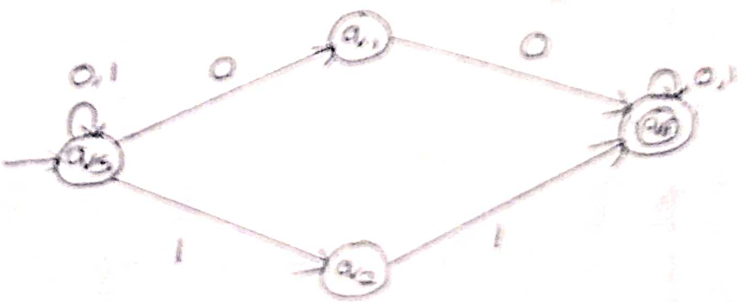
(c) we eliminate the * operations, by introducing two more new states q_3 & q_4 on ε-moves



(d) we eliminate + & concatenation by introducing two (say) more states



NFA with ε-moves



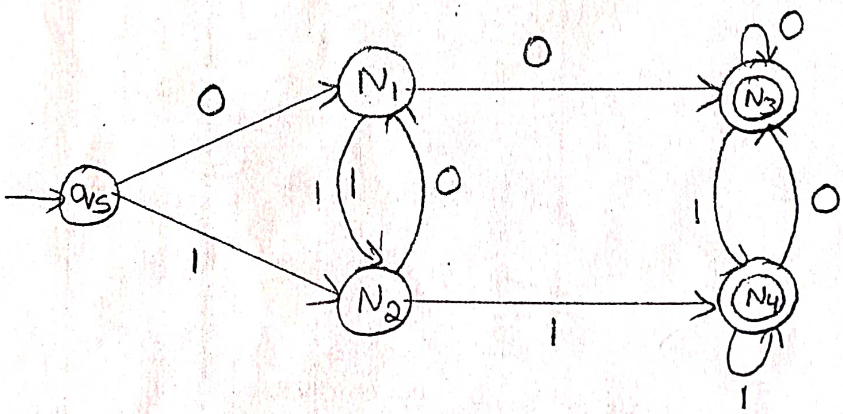
NFA w/o ε-moves

(2) Construction of DFA:-

(a) The transition table for NFA can be constructed as following:-

States	inputs	
	0	1
$\rightarrow q_s$	$\underbrace{[q_s, q_1]}_{N_1}$	$\underbrace{[q_s, q_2]}_{N_2}$
q_1	q_f	ϕ
q_2	ϕ	q_f
q_f	q_f	q_f

States	inputs	
	0	1
N_1	$\underbrace{[q_s, q_f, q_1]}_{N_3}$	$[q_2, q_s]$
N_2	$[q_s, q_1]$	$\underbrace{[q_s, q_2, q_f]}_{N_4}$
N_3	$[q_s, q_f, q_1]$	$[q_2, q_s, q_f]$
N_4	$[q_s, q_f, q_1]$	$[q_2, q_s, q_f]$

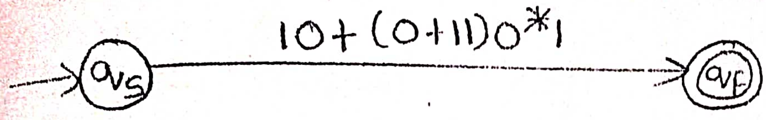


DFA

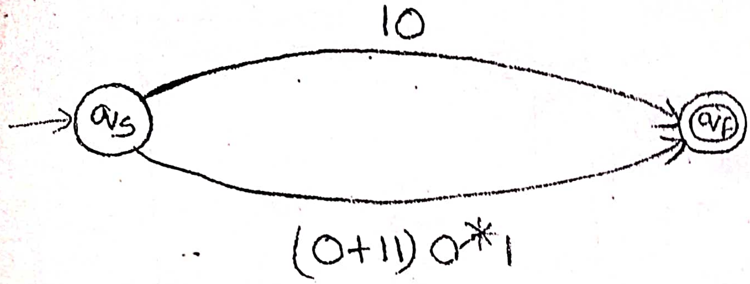
(8) construct a DFA with reduced states equivalent to the re $10 + (0+11)0^*1$

Sol:- construction of transition graph:-

(1)

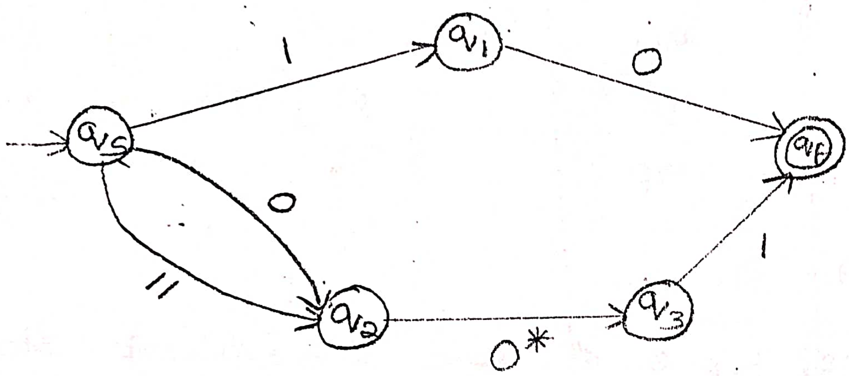


(2)



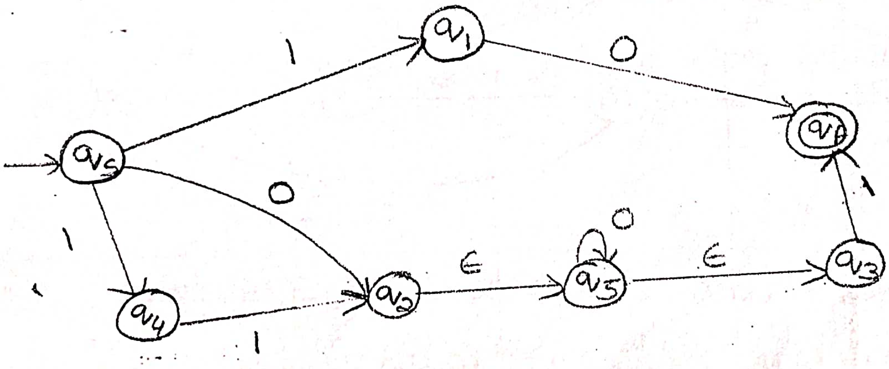
Elimination of '+'

(3)



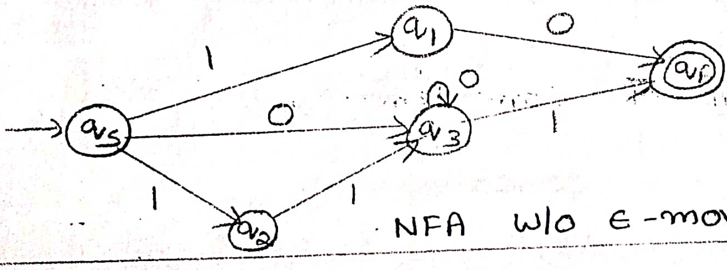
Elimination of concatenation & +

(4)



Elimination of concatenation & *

(5)



NFA w/o ε-moves

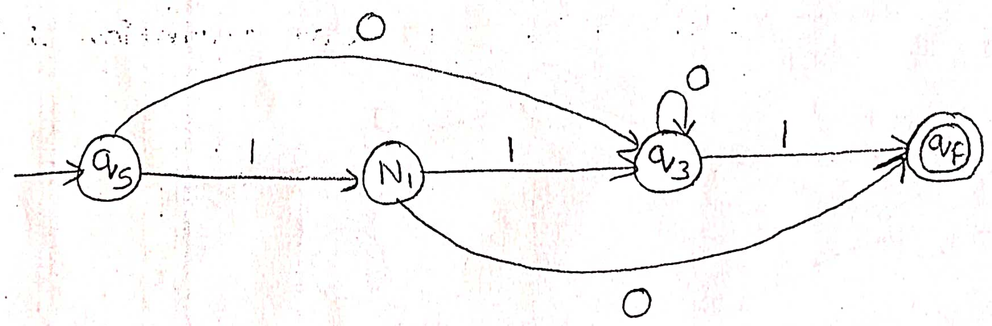
(2) Construction of DFA:-

States	Inputs 0	1
$\rightarrow q_s$	q_3	$\underbrace{[q_1, q_2]}_{N_1}$
q_1	q_f	ϕ
q_2	ϕ	q_3
q_3	q_3	q_f
q_f	ϕ	ϕ
N_1	q_f	q_3

(3) Reduced DFA:-

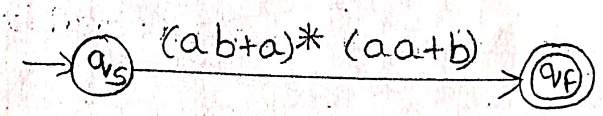
Here, q_f & ϕ are equivalent states. so,

we can remove the ϕ state

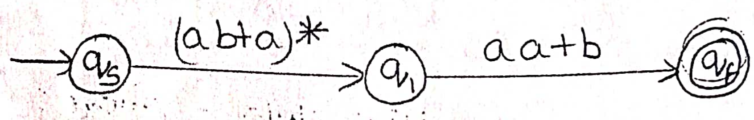


(4) Construct the transition systems equivalent to the re $(ab+a)^*(aa+b)$

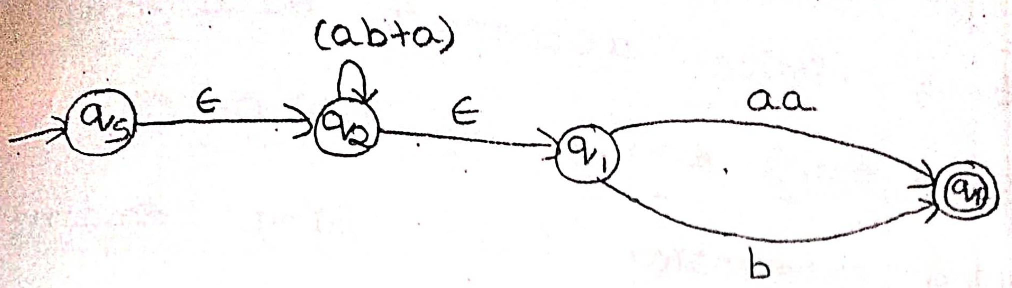
Sol:- Construction of transition graphs:-

(1) 

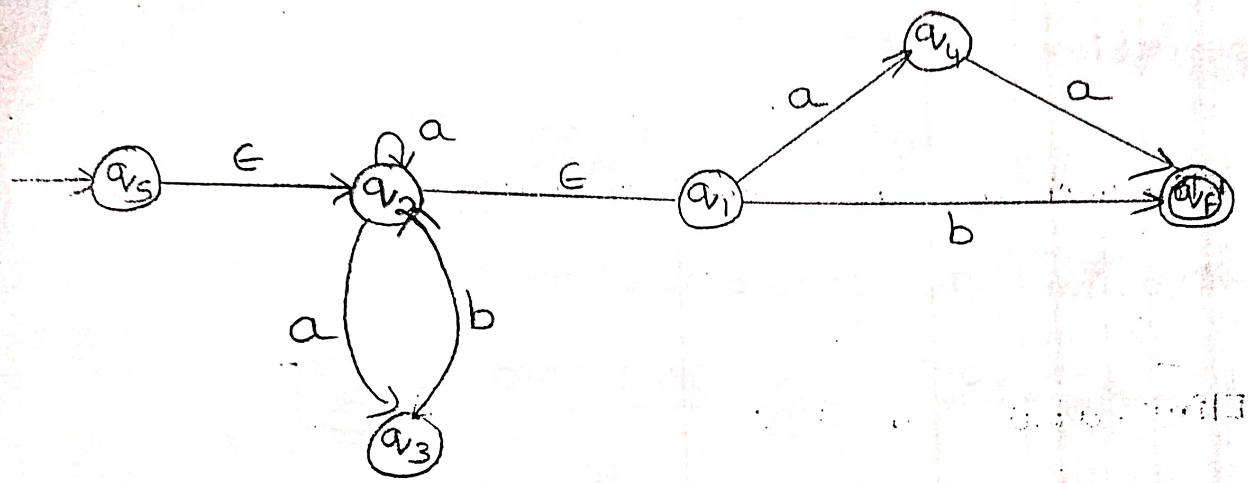
Elimination of concatenation:-

(2) 

Elimination of * & Concatenation:-



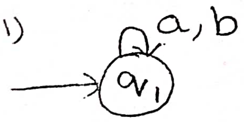
Elimination of + & Concatenation:-



Transition System

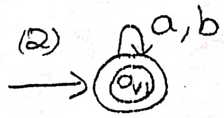
(10) Find the set of strings over $\Sigma = \{a, b\}$ recognized by the transition systems:-

(1)



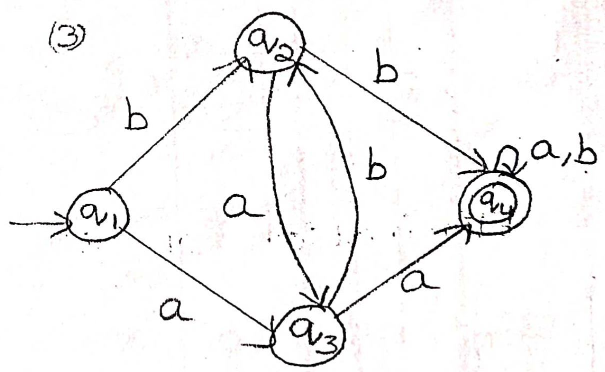
$\emptyset$

(2)



$(a+b)^*$

(3)

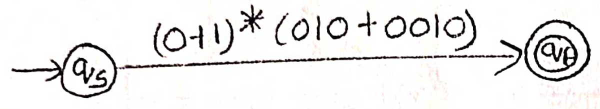


Two successive a's (or) two successive b's

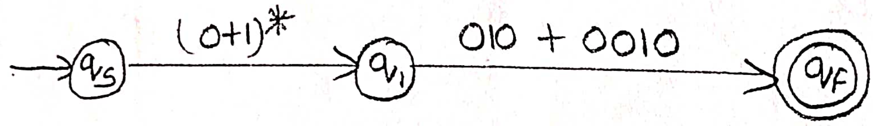
(11) Construct a finite automata accepting all strings over {0,1} ending 010 (or) 0010

Sol:- The regular expression for strings ending in 010 (or) 0010 is $(0+1)^*(010+0010)$

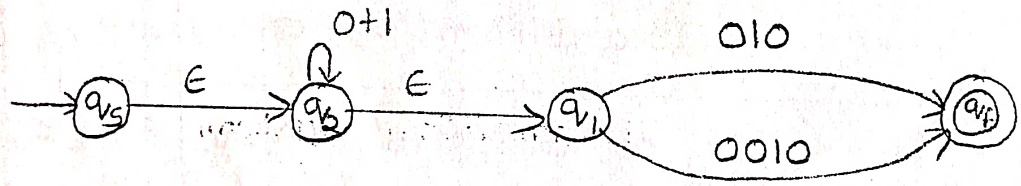
Construction of transition graph:-

(1) 

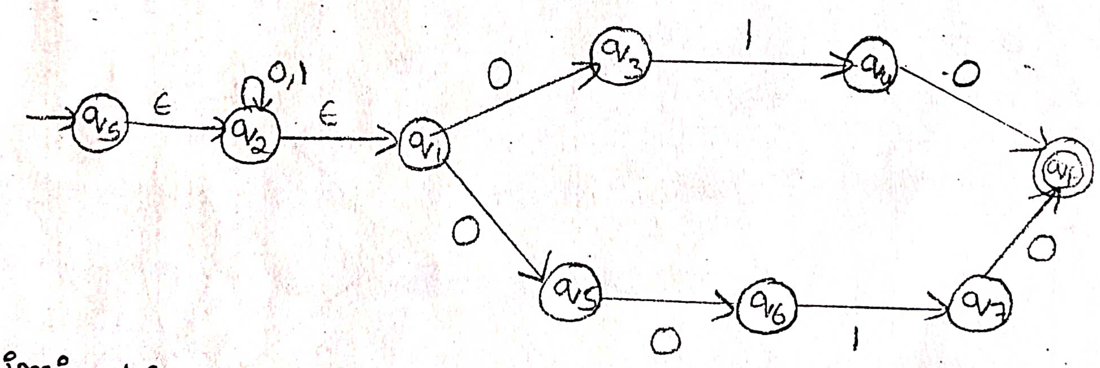
Elimination of concatenation:-

(2) 

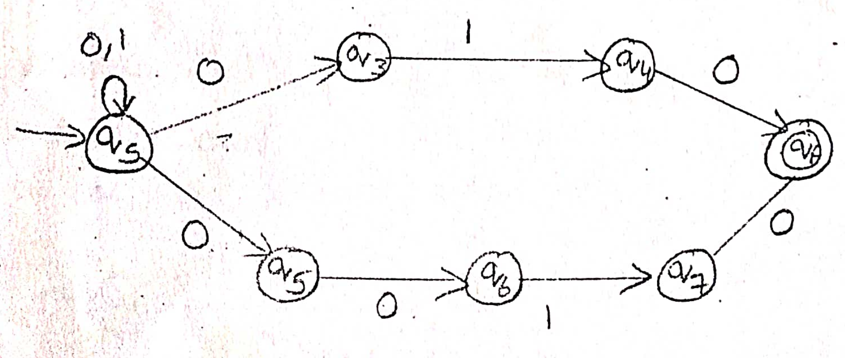
Elimination of * & +

(3) 

Elimination of concatenation:-

(4) 

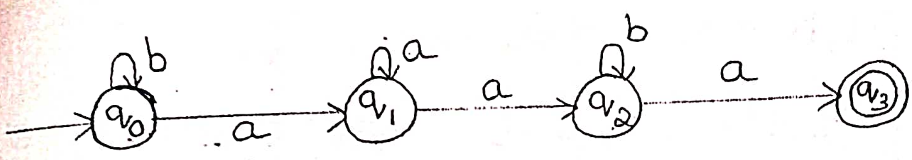
Elimination of *:-

(5) 

all

Conversion of Finite Automata to Regular Expression:- (15)

1) Find the re for the language accepted by the following automata?



As there are no ϵ-moves, we can obtain the equations as. The initial state must contain

(i) $q_0 = q_0 b + \epsilon$

(ii) $q_1 = q_0 a + q_1 a$

(iii) $q_2 = q_1 a + q_2 b$

(iv) $q_3 = q_2 a$

Step 2:- It is necessary to reduce the number of known by repeated substitution

Let us consider

$$q_0 = \epsilon + q_0 b$$

This is in the form of $R = Q + RP \Longleftrightarrow R = QP^*$

$$q_0 = \epsilon (b)^* = b^*$$

Placing q_0 in q_1- equation, we get

$$q_1 = q_0 a + q_1 a$$
$$= b^* a + q_1 a$$
$$= q_1 a + b^* a$$

This is in the form of $R = Q + RP \Longleftrightarrow QP^*$

$$q_1 = b^*aa^*$$

By placing q_1 - equation in q_2, we get

$$q_2 = q_1 a + q_2 b$$

$$= b^*aa^*a + q_2 b \quad [\because R = Q + RQ \Leftrightarrow R = QP^*]$$

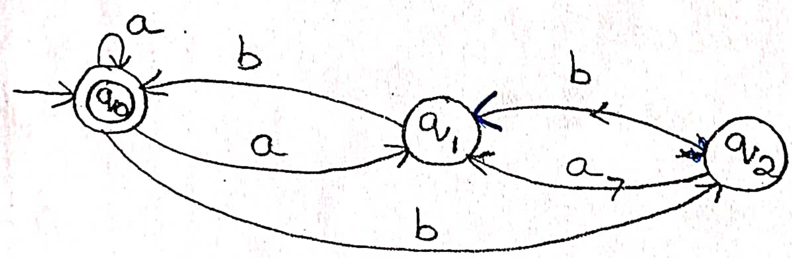
$$= (b^*aa^*a)\, b^*$$

Finally, reduce q_3 state the regular expression can be generated by this final state

$$q_3 = q_2 a$$

$$= b^*aa^*a\, b^*a$$

(2) Describe the regular expressions accepted by the following automaton



The Reachability equations are:-

$$q_0 = q_0 a + q_1 b + \epsilon \quad - (1)$$

$$q_1 = q_0 a + q_2 b \quad - (2)$$

$$q_2 = q_0 b + q_1 a \quad - (3)$$

Reduce the number of known states by repeated substitution & solve q_0 inorder to get the regular expression as it is the final state

Placing q_2 in q_1 - equation, we get

(16)

$$q_1 = q_0 a + q_2 b$$

$$= q_0 a + (q_0 b + q_1 a) b$$

$$= q_0 a + q_0 bb + q_1 ab$$

$$= q_0 (a+bb) + q_1 ab$$

This is the form of $R = Q + RP \Leftrightarrow R = QP^*$

$$q_1 = q_0 (a+bb)(ab)^*$$

Place q_1 in the q_0 - equation, we get

$$q_0 = q_0 a + q_1 b + \epsilon$$

$$= q_0 a + [q_0 (a+bb)(ab)^*] b + \epsilon$$

$$= q_0 a + q_0 b (a+bb)(ab)^* + \epsilon$$

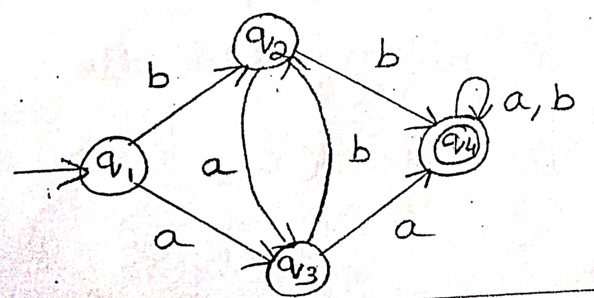
$$= q_0 (a + b(a+bb)(ab)^*) + \epsilon$$

$$= \underset{Q}{\underline{\epsilon}} + \underset{R}{\underline{q_0}} (a+b(a+bb)(ab)^*) \quad [\because R = QP^*]$$

$$= \epsilon \, [a+b(a+bb)(ab)^*]^*$$

$$= [a+b(a+bb)(ab)^*]^*$$

Find the regular expression for the following transition diagram:-



The Reachability equations are:-

$q_1 = \epsilon$ - ①

$q_2 = q_1 b + q_3 b$ - ②

$q_3 = q_1 a + q_2 a$ - ③

$q_4 = q_2 b + q_3 a + q_4 a + q_4 b$ - ④

Now, $q_2 = q_1 b + q_3 b$

$= (q_1 + q_3) b = (\epsilon + q_3) b = q_3 b$

$q_3 = q_1 a + q_2 a$

$= (q_1 + q_2) a = (\epsilon + q_2) a = q_2 a$

Now consider,

$q_4 = q_2 b + q_3 a + q_4 a + q_4 b$

$= (q_3 b) b + q_3 a + q_4 a + q_4 b$

$= q_3 (bb + a) + q_4 (a + b)$

$= (q_1 a + q_2 a) \cancel{+q_3} (bb + a) + q_4 (a + b)$

$= (q_1 + q_2) a (bb + a) + q_4 (a + b)$ ✓

$= \epsilon a (bb + a) + q_4 (a + b)$

$q_4 = a (bb + a) + q_4 (a + b)$

$= a (bb + a) (a + b)^*$

[∵ This in the form of $R = Q + RP \Leftrightarrow QP^*$]

Examples:-

Find out wheather these two FA's are equivalent or not?

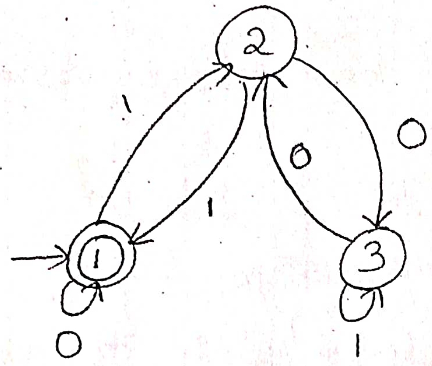


Fig:- ①

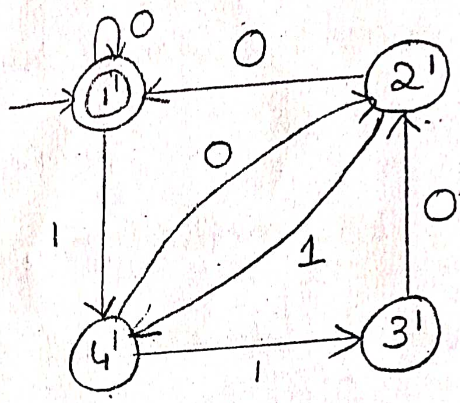


Fig:- ②

Here, The final states are : { 1, 1' }

Non-final states are: { 2, 2', 3, 3', 4' }

(V, V')	(V_0, V_0')	(V_1, V_1')
(1, 1')	(1, 1') Final states	(2, 4') Non-final states
(2, 4')	(3, 2') Non-final	(1, 3')

V_0 - state on i/p 0 in first FA

V_1' - state on i/p 1 in second FA

Construction steps:-

(1) Start with the initial state from both FA's check their transitions on both i/p's. i.e, on 0 - transitions they both are moving to final states & on 1- transition they are moving to non-final states

(2) Now consider 1^{st} row, 2^{nd} column element i.e (2, 4') check for both transitions. (13) on 0 i/p it is moving to final states & on 1 i/p it is moving to 1 FA to final & 2 FA to non-final. So, we terminate the process here & conclude that

The given automaton are not equivalent

(2) Test the 2 DFA's are equivalent (or) not?

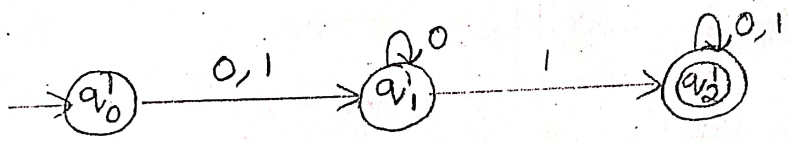


Fig:- (1)

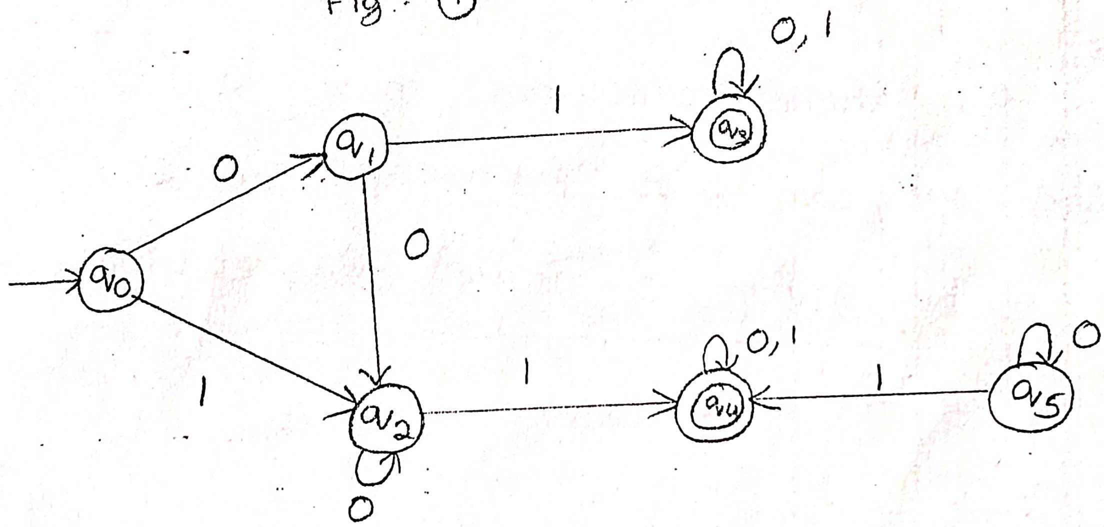


Fig:- (2)

Sol:- The final states are $\{q_2', q_3, q_4\}$

Non-final states are $\{q_0', q_1', q_0, q_1, q_2, q_5\}$

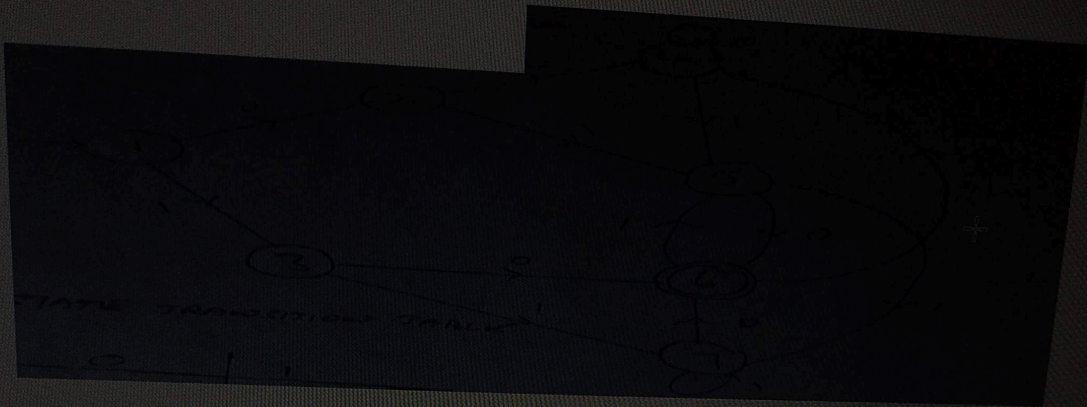
Now we should construct comparison table starting with initial states (q_0, q_0')

(v, v')	(v_0, v_0')	(v_1, v_1')
(q_0, q_0')	(q_1, q_1') Non-Final	(q_2, q_1') Non-Final
(q_1, q_1')	(q_2, q_1') Non-final	(q_3, q_2') Final
(q_2, q_1')	(q_2, q_1') Non-final	(q_4, q_2') final
(q_3, q_2')	(q_3, q_2') Final	(q_3, q_2') Final
(q_4, q_2')	(q_4, q_2') Final	(q_4, q_2') final

As there are no new states are produced, we can stop the construction & conclude

The given 2 DFA's are equivalent

Question: Find the minimum state FA for the given machine



Step 1: Construct Transition table

δ	0	1
→1	2	3
2	4	5
3	6	7
4	4	5
5	6	7