UNIT II

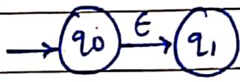
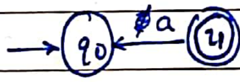
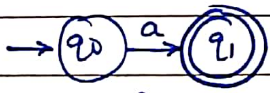
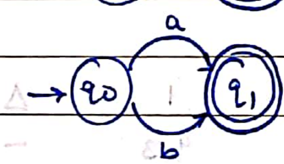
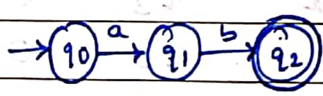
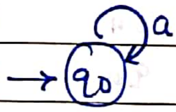
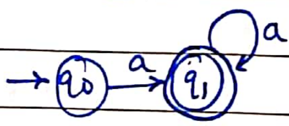
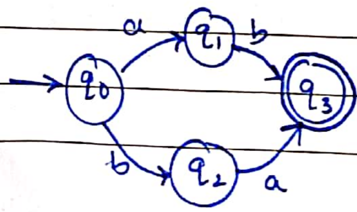
Regural Expression (RE)

Language → Finite Automata (F.A) $\xrightarrow{\text{expression}}$ RE

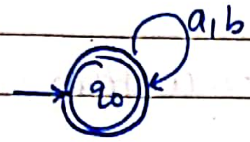
* Regular expression

The language accepted by F.A can easily be described by expression is called regular expression.

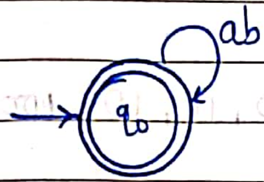
*

		F.A	RL	RE
(I)	Null string (ε)		$L(R) = \{\epsilon\}$	$R = \epsilon$
(II)	Empty string (∅)		$L(R) = \{\phi\}$	$R = \phi$
(III)			$L(R) = \{a\}$	$R = a$
(IV)	union/OR (+)		$L(R) = \{a, b\}$	$R = a+b$
(V)	concatenation (•)		$L(R) = \{ab\}$	$R = ab$
(VI)	closure (*)		$L(R) = \{\epsilon, a, aa, \ldots\}$	$R = a^*$
(VII)	+ve closure		$L(R) = \{a, aa, aaa \ldots\}$	$R = aa^*$ or a^+
(VIII)			$L(R) = \{ab, ba\}$	$R = ab+ba$

FABER-CASTELL
Date
Page No.

1X 

$L(R) = \{\epsilon, a, b, ab, aa, bb, aab \ldots\}$ $R = (a+b)^*$

X 

$L(R) = \{\epsilon, ab, abab \ldots\}$ $R = (ab)^*$

Q* Write R.E. for the Language accepting all Strings that can take any no. 'a' & 'b'.

Ans:- R.E = $(a+b)^*$

$L(R) = \{\epsilon, a, b, ab, ba, aa, bb, aba, abb, \ldots\}$

* Starting with ~~one~~ '1' & ending with 0.

Ans:- $L(R) = \{10, 110, 100, 1010010 \ldots\}$

RE = $1(1+0)^*0$

* Starting with 'a' but not having consecutive 'b'

Ans:- $L(R) = \{a, ab, aba, aab, aaba, aaa, \ldots\}$

RE = $(a+ab)^*$

Date
Page No.

* All the strings do not contain the substring '01'

Ans:- $L(R) = \{0, 1, 00, 11, 10, 100, 111000 \ldots\}$

$R.E = 1^* 0^*$

* At least 2 consecutive '0'

Ans:- $L(R) = \{aa, aab, baa, aaab, babaaba \ldots\}$

$R.E = (a+b)^* aa (a+b)^*$.

insem2022
Q3a- * The set of strings over alphabet $\Sigma = \{0,1\}$ that have atleas one '1'.

$L(R) = \{1, 10, 01, 110, 11, 011 \ldots\}$

$R.E = (1+0)^* 1 (1+0)^*$

* Atmost one '1' (Add ϵ)

$L = \{\epsilon, 0, 1, 00, 01, 10, 0001, 1000\}$

$RE = 0^* (1+\epsilon) 0^*$

FABER-CASTELL

Date

Page No.

* ending with '00' & begining with '1'

$$R.E = 1(0+1)^*00$$

14/8/23

* Construct R.E $\Sigma = \{a,b\}$ where $L_1 = \{$ length exactly 2

Ans:- $\Sigma = \{a,b\}$

$L_1 = \{aa, ab, ba, bb\}$

$$R.E = aa + ab + ba + bb.$$
$$= a(a+b) + b(a+b)$$
$$= (a+b)(a+b).$$

* L_2 = Length is atleast 2.

$L_2 = \{aa, ab, ba, aa, aaa, aba, bbb, bba \ldots\}$

$$R.E = (a+b)(a+b)(a+b)^*$$

* L_3 = Length is atmost 2.

$L_3 = \{\epsilon, a, b, ab, ba, aa, bb\}$

$$R.E = \epsilon + a + b + ab + ba + aa + bb.$$
$$= (a+b+\epsilon)(a+b+\epsilon)$$

Scanned with CamScanner

* L_4 = Even length string.

$L_4 = \{ \epsilon, aa, ab, ba, bb, aaaa, abba, baba, bbbb \ldots \}$

R.E = $[(a+b)(a+b)]^*$

* L_5 = Odd length string.

$L_5 = \{ a, b, aaa, aba, bbb \ldots \}$

R.E = $[(a+b)(a+b)]^* \cdot (a+b)$

* L_6 = length of string is divisible by 3.

$L_6 = \{ \epsilon, aaa, aba, bbb, aaaaaa \ldots \}$

R.E = $[(a+b)(a+b)(a+b)]^*$

* L_7 = length is divisible by 3 & 2 reminder

$L_7 = \{ aa, ab, ba, bb, aaaaa, ababb \ldots \}$

R.E = $[(a+b)(a+b)(a+b)]^* (a+b)(a+b)$

* Find R.E over $\Sigma = \{a, b\}$ where L_1 = no. of a is exactly 2.

$L_1 = \{aa, aab, baa, aba, ababb \ldots\}$.

$R.E = b^* a b^* a b^*$

* L_2 = No. of a's are atleast 2.

$L_2 = \{aa, aab, aaab \ldots\}$

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

* L_3 = No. of a' is atmost 2

$L_3 = \{\epsilon, a, b, bb, ab, aa, aab, aba \ldots\}$

$R.E = b^* (a+\epsilon) b^* (a+\epsilon) b^*$

* L_4 = No. of a is even
L_5 = No. of a's are odd.

$L_4 = \{\epsilon, b, aa, aab, aba, aabaa, aabb \ldots\}$

$R.E = (b^* a b^* a)^*$

$L_5 = \{ab, ba, aaab, aabba \ldots\}$.

$R.E = b^* a b^* (b^* a b^* a)^*$

Scanned with CamScanner

FABER-CASTELL

* Construct R.E over $\Sigma = \{0, 1\}$ where L_1 = set of strings start with ~~one~~ '1'.

$R.E = 1(1+0)^*$

* L_2 = ending with '1'

$R.E = (1+0)^* 1$

* L_3 = containing 1

$R.E = (1+0)^* 1 (1+0)^*$

* L_4 = Starting & ending with diff symbols.

$R.E = 0(0+1)^* 1 + 1(0+1)^* 0$

* L_5 = Starting & ending with same symbol

$L_5 = \{0, 1, 11, 00, 010, 101, \epsilon \cdots\}$

$R.E = 0(0+1)^* 0 + 1(0+1)^* 1 + 0 + 1 + \epsilon$

Scanned with CamScanner

FABER-CASTELL
Date
Page No.

ababaabbba

Q* Difference betⁿ NFA & DFA
Q* Difference betⁿ moore & mealy machine
Q* Difference betⁿ NFA & ENFA

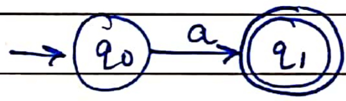
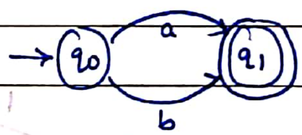
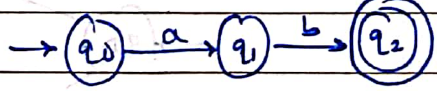
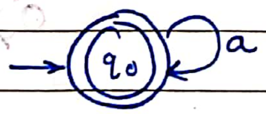
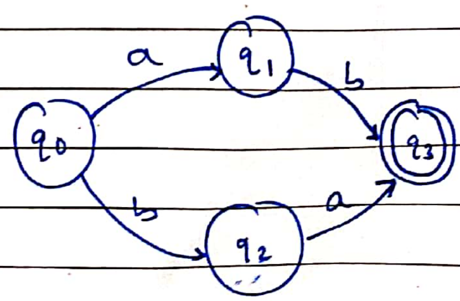
Q* Construct R.E ~~where~~ over $\Sigma = \{a, b\}$ where set of all strings in such way that 'aa' substring is not allowed. (two a's should not come together

Ans $L = \{\epsilon, a, b, ab, ba, bba, bb, aba, abba, \}$

$R.E = b^*(ab)^* b^* (ba)^* b^* + a$.

18/8/23

Q* Convert regular expression into F.A (F.A can be DFA, NFA, ENFA)

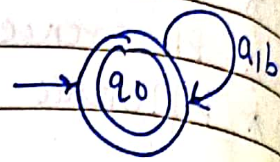
	Regular expression		F.A.
(I)	a	$L = \{a\}$	
(II)	$a+b$	$L = \{a, b\}$	
(III)	ab	$L = \{ab\}$	
(IV)	a^*	$L = \{\epsilon, a, aa \ldots\}$	
(V)	$ab+ba$	$L = \{ab, ba\}$	

Scanned with CamScanner

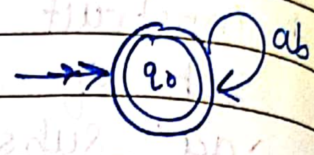
FABER-CASTELL
Date
Page No.

Regular expression $\qquad$ F.A

(VI) $(a+b)^*$ $\quad L=\{\epsilon, a, b, aa, ab, bbb \ldots\}$



(VII) $(ab)^*$ $\quad L=\{\epsilon, ab, abab \ldots\}$

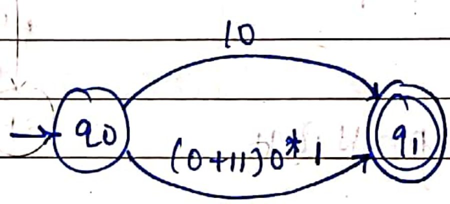


Q * Design F.A from given R.E = $10+(0+11)0^*1$

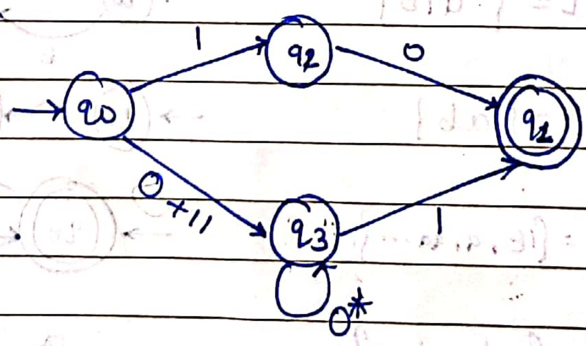
Ans:- $\underbrace{10}_{A} + \underbrace{(0+11)0^*1}_{B}$

A + B

(1)



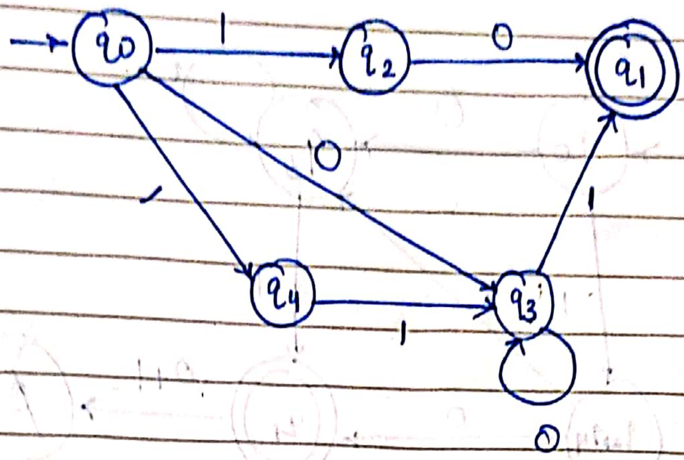
(2)



Scanned with CamScanner

FABER-CASTELL
Date ....
Page No. ....

3



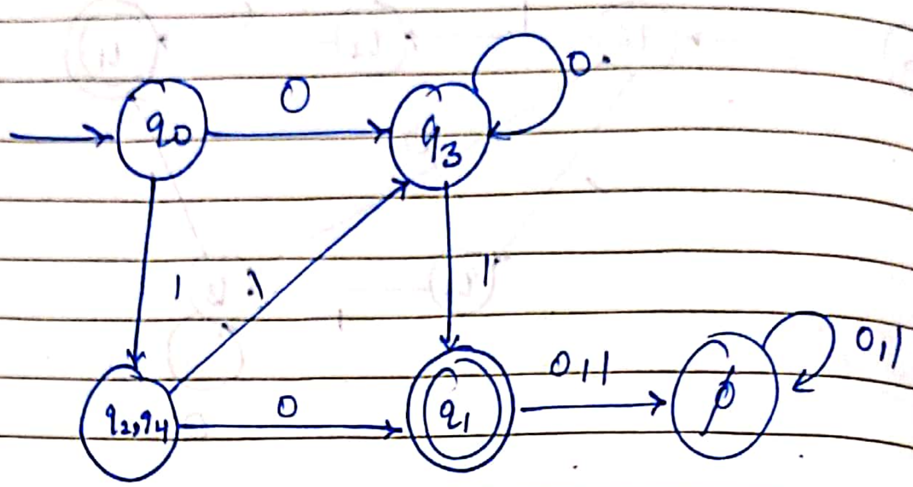
- NFA Transition table.

$q \backslash \Sigma$	0	1
$\rightarrow q_0$	q_3	$\{q_2, q_4\}$
q_2	q_1	ϕ
q_3	q_3	q_1
q_4	ϕ	q_3
$* q_1$	ϕ	ϕ

- DFA transition table

$q \backslash \Sigma$	0	1
$\rightarrow q_0$	q_3	$\{q_2, q_4\}$
q_3	q_3	q_1
$\{q_2, q_4\}$	q_1	q_3
q_1	ϕ	ϕ
ϕ	ϕ	ϕ

Scanned with CamScanner



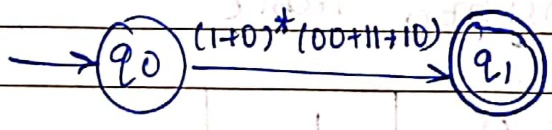
* Convert following R.E into F.A.

$$(1+0)^*(00+11+10)$$

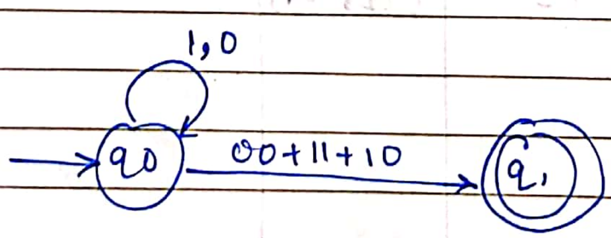
Ans:-

$$\underbrace{(1+0)^*}_{A}\ \underbrace{(00+11+10)}_{B}$$

①

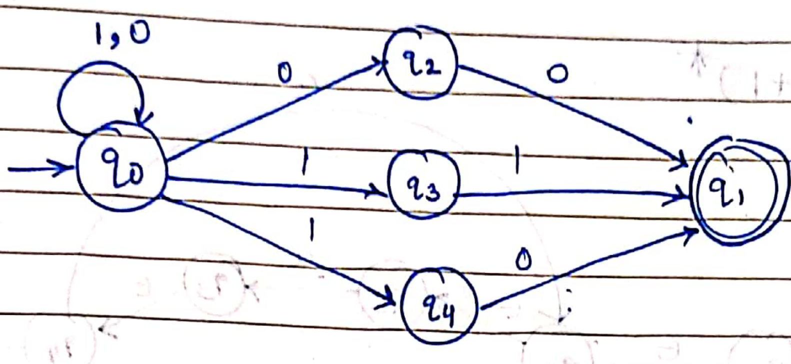


②



FABER-CASTELL
Date
Page No.

3

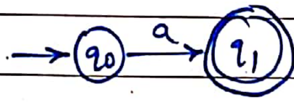


Q* $R = 0^*1 + 10$

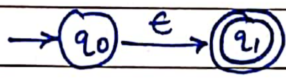
Q* $R = ba^*b$

* Regular expression to F.A with ϵ.

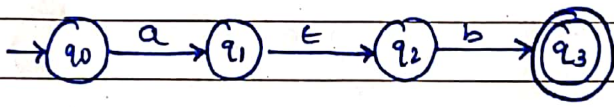
1) $R = a$



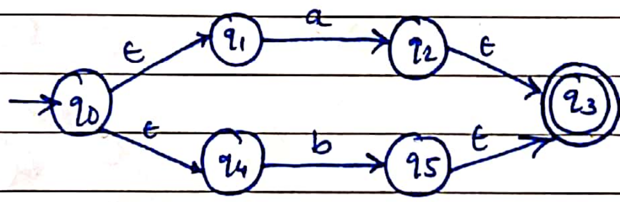
2) $R = \epsilon$



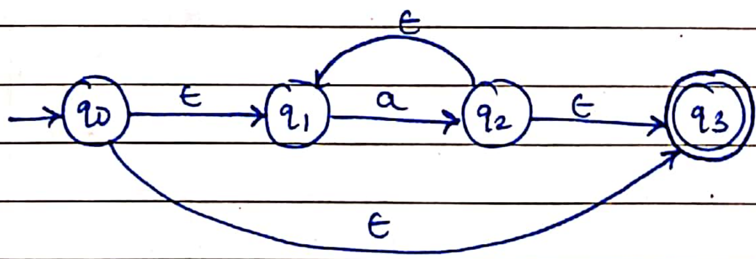
3) $R = ab$



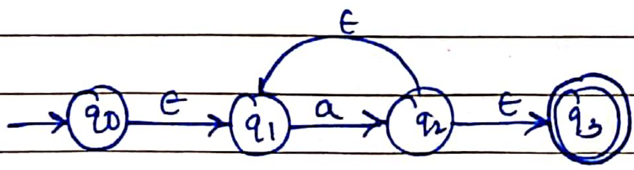
4) $R = a+b$



5) $R = a^*$

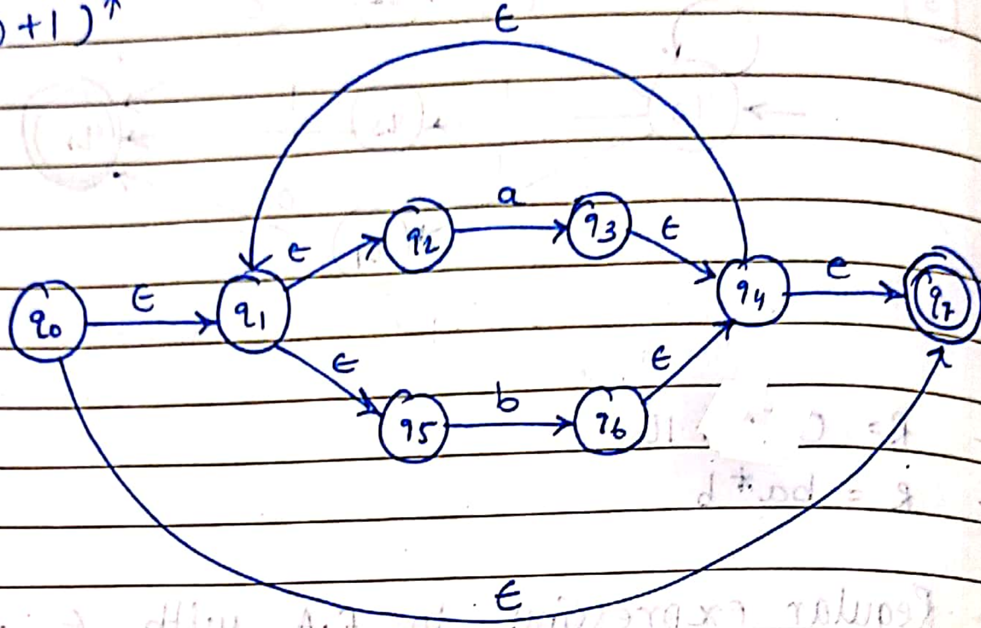


6) $R = a^+$

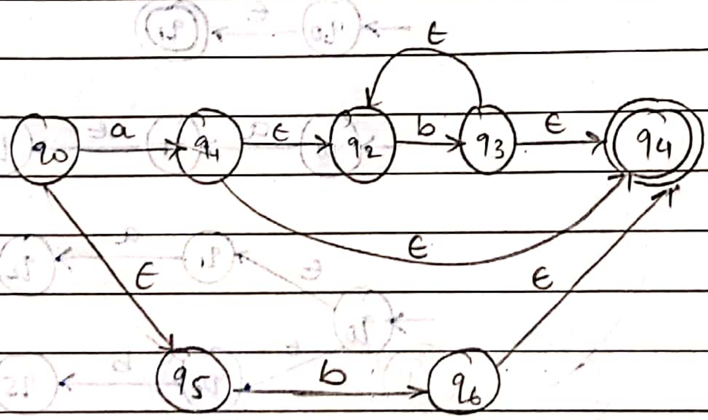


Scanned with CamScanner

Q* $(0+1)^*$



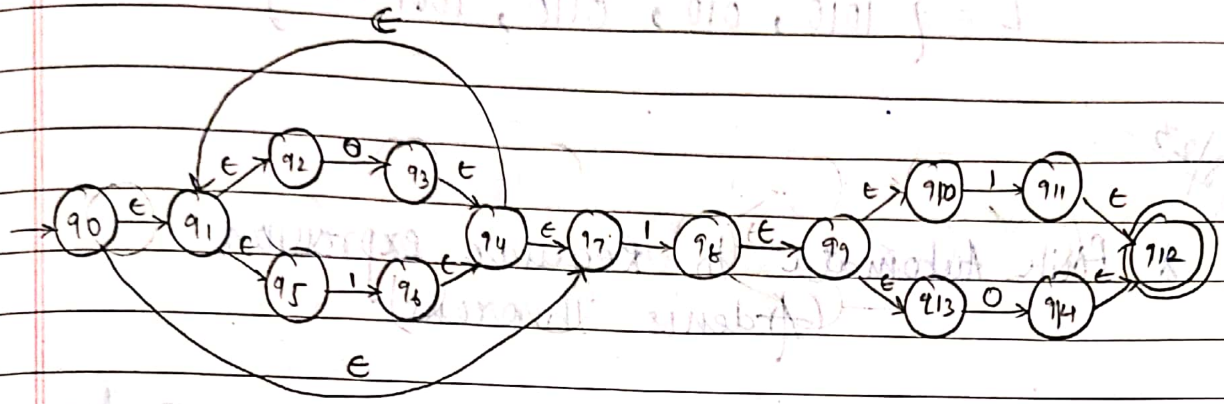
Q * $ab^* + b$



Scanned with CamScanner

insem 2012

Q* $R = (0+1)^* 1 (0+1)$



Q* Consider 2 R.E, $R = 0^* + 1^*$ & $S = 01^* + 10^* + 1^*0 + (0^*1)^*$

(I) find string corresponding to R but not to S

(II) corresponds to S & not to R.

(III) corresponds to both R & S.

(IV) corresponds to ~~neigh~~ neither S nor R.

Ans:- $L_R = \{\epsilon, 00, 0, 000, 0000 \ldots, 1, 11, 111, 1111 \ldots\}$

$L_S = \{\epsilon, 0, 01, 011, 0111, 01111 \ldots, 1, 10, 100, 1000, 10000 \ldots, 10, 110, 1110, 11110 \ldots, 11, 011, 0011\}$

→ (I)

$L = \{00, 000, 0000 \ldots\}$

(II)

$L = \{10, 100, 110, 1110, 0011, 011 \ldots\}$

(III)

$L = \{\epsilon, 0, 1, 11, 1111, 111 \ldots\}$

(IV)

$L = \{1010, 010, 0110, 1001 \dots\}$

24/8/23

* Finite Automata to Regular expression (Ardens Theorem)

Let p & q be 2 R.E over Σ & p does not contain null string (ϵ), then

$R = Q + RP$

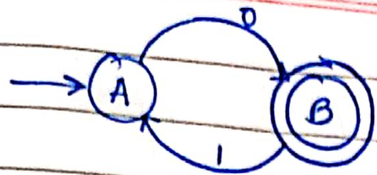
$$\left[R = Q + RP \approx R = QP^* \right]$$

$$R = Q + RP$$
$$R = Q + (Q + RP)P$$
$$R = Q + QP + RP^2$$
$$R = Q + QP + (Q + RP)P^2$$
$$R = Q + QP + QP^2 + RP^3$$
$$R = Q(\epsilon + P + P^2 + P^3 \dots)$$

$$\Rightarrow \left[R = QP^* \right]$$

FABER-CASTELL
Date
Page No.

Q*



Note :- Add ϵ to initial state

Ans

(1) $A = \epsilon + B \cdot 1$

$B = A \cdot 0$

(2) $B = A \cdot 0$

$B = (\epsilon + B \cdot 1) \cdot 0$

$B = 0 \cdot \epsilon + B \cdot (10)$

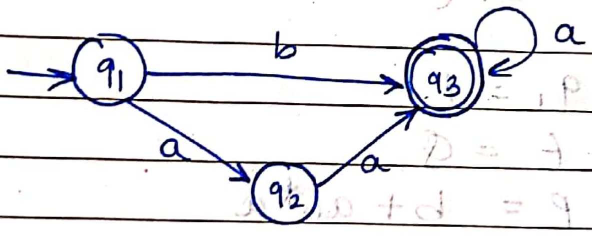
$B = 0 + B \cdot (10)$

comparing with $R = Q + RP$

$R = B$, $Q = 0$, $P = 10$

$\rightarrow$ $B = 0 \cdot (10)^*$ $[R = QP^*]$

Q*



$q_1 = \epsilon$ — (1)

$q_2 = q_1 \cdot a$ — (2)

$q_3 = q_1 \cdot b + q_2 \cdot a + q_3 \cdot a$ — (3)

$\rightarrow$ $q_3 = q_1 \cdot b + q_2 \cdot a + q_3 \cdot a$

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

$q_3 = b + (\epsilon \cdot a) a + q_3 \cdot a$

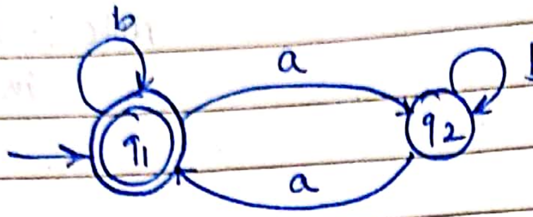
$q_3 = b + aa + q_3 \cdot a$

$q_3 = (b + aa) a^*$

$\left[\begin{array}{l} b + aa = Q \\ q_3 = R \\ a = P \end{array} \right.$

Scanned with CamScanner

Q *



$q_1 = \epsilon + q_1 . b + q_2 . a .$ — (1)

$q_2 = q_1 . a + q_2 . b .$ — (2)

compare eq (2) with $R = Q + RP$

$\rightarrow R = q_2 , \; Q = q_1 a , \; P = b .$

$\rightarrow q_2 = (q_1 a)(b)^*$ — (3) $[R = QP^*]$

$\Rightarrow q_1 = \epsilon + q_1 b + (q_1 a b^*) a$

$q_1 = \epsilon + q_1 (b + ab^* a)$

$\rightarrow$ ~~$q_1 = Q$~~ $q_1 = R$

$\epsilon = Q$

$P = b + ab^* a$

$\Rightarrow q_1 = \epsilon . (b + ab^* a)^*$ $[R = QP^*]$

$\therefore q_1 = (b + ab^* a)^*$

Ans

Date
Page No.

imp // * Pumping Lemma.

It is used to prove that given language is <u>not regular</u>

- use proof of contradiction.

* Steps.

(I) Assume 'L' is regular language, then L has pumping length (constant) → p such that any string ($w \in L$) | $|w| \geq p$

(II) 'w' is divided into three parts [here $\in$ = belongs to

$w = xyz$

then it must satisfy :-

(i) $xy^iz \in L \mid i \geq 0$

(ii) $|y| > 0$

(iii) $|xy| \leq p$

Date
Page No.

Q * Show that $L = \{a^n b^n \mid n \geq 0\}$ is not regular

Ans:- ① Assume L is regular language.

② $L = \{\epsilon, ab, aabb, aaabbb \ldots\}$

③ Assume pumping length $(p) = 2$.

$w = aabb$ | $|w| \geq p$, $4 \geq 2$

④ Devide w in ~~three~~ three parts.

$w = \underbrace{a}_{x}\underbrace{a}_{y}\underbrace{bb}_{z}$

$x = a$
$y = a$
$z = bb$

(i) $xy^iz \in L \mid i \geq 0$

$i = 2$.

[$aaabb \notin L$] → condition not satisfied

(ii) $|y| > 0$.

$1 > 0$

- satisfied.

Scanned with CamScanner

(iii) $|xy| \leq p$.

$2 \leq 2$ — Satisfy

As 1st condition is not satisfied ∴ our assumption is wrong ∴ given language is not regular .

insem 2022

Q* $L = \{0^m 1^m 0^{m+n} \mid m \geq 1 \ \& \ n \geq 1\}$

→ Assume L is regular language .

$L = \{0100, 011000 \cdots\}$.

~~W~~ consider pumping length (p) = 2 .

$W = 011000$

$x = 0$
$y = 11$
$z = 000$.

(1) $xy^iz \in L$

$i = 2$.

0111000