Unit III - Regular Expression

27-3-23

- The language accepted by the finite automata are described or represented by simple expression called Regular Expression.

- Formal Definition:
  The class of Regular Expressions over 'Σ' is defined recursively as follows:

(i) The letters 'ϕ' (null set) & 'ϵ' (empty ~~tra~~ string of length 0) are regular expressions over 'Σ'

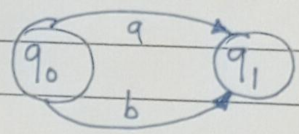
(ii) Every letter 'a' $\subset \Sigma$ is a Regular expression over Σ.

(iii) If 'R_1' & 'R_2' are regular expressions over Σ then $(R_1 + R_2)$, $(R_1 \cdot R_2)$ & $(R_1)^*$ where '+' indicates alternation (parallel path), '$\cdot$' indicates ~~conta~~ concatination (series) & '$*$' denotes iteration (closure)

(iv) The regular expressions are only those that are obtained using rules (i), (ii), (iii)

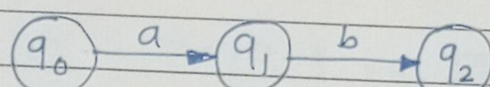
For eg.

$a+b \Longrightarrow$



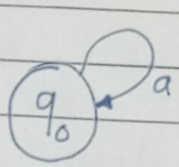
[Parallel path]

$a \cdot b \Longrightarrow$



[series]

$a^* \Longrightarrow$



In the above transition graphs '(a+b)' stands for either 'a' or 'b' & 'a·b' stands for 'a' followed by 'b' while 'a*' stands for any number of occurence of 'a'.

If 'r' is a regular expression then the language represented by r is L(r).

(i) if $r = (a+b)$ then $L(r) = \{a, b\}$
(ii) if $r = (a \cdot b)$ then $L(r)$ $\{ab\}$
(iii) if $r = (a)^*$ then $L(r)$ $\{\epsilon, a, aa, ---\}$

Q.1] If $r = ab^*a$ then describe $L(r)$.

→ $L(r) = \{a\epsilon a, aba, abba, abbba, ---\}$

Q.2] If $L(r) = \{\epsilon, x, xx, xxx, xxxx, xxxxx\}$ then what ~~is~~ is r?

→ $r = x^*$ where $* = 5$

$\therefore$ $\underline{r = (\epsilon + x^5)}$

Q.3] If $L(r) = \{a, c, ab, cb, abb, cbb, abbb, cbbb, ---\}$ then $r = ?$

→ ~~$r = (a+b)\, b^*$~~ $\underline{r = (a+c)\, b^*}$

Q.4] If $L(r)$ = set of all strings over Σ $\Sigma = \{0, 1\}$ ending with 011 then what is r?

→ ~~$r = 0^* 1^* 011$~~ $\underline{r = (0+1)^* 011}$

Page No.
Date

Q.5] Represent the language over $\Sigma = \{a, b\}$ with all strings starting & ending with 'a' & any number of b's ~~Find r~~ in between them. Find r.

→ ~~$r = ab^*(a+b)^* ab^*$~~

~~$r = ab^* (a+b)^* ab^*$~~

~~$\therefore L(r) = \{aa, ababab, abbaabbabb, \ldots\}$~~

~~$r = ab^*(a+b)^* a$~~

→ $r = \underline{ab^* a}$

$L(r) = \{aa, aba, abba, abbba, \ldots\}$

Q.6] Represent set of all words over $\Sigma = \{a, b\}$ containing atleast one 'a' using regular expression.

→ ~~$r = aa^* b$~~   ~~$r = a(a+b)^*$~~

~~$L(r) = \{ab, aab, aaab, aaaab, \ldots\}$~~

~~$r = a(a+b)^*$~~

~~$L(r) = \{a, aa, ab, aba, abb, aaa, abbaa, \ldots\}$~~

Ans → $\underline{(a+b)^* a (a+b)^*}$

Language can have one a & it may have any no. of trailing a's & b's & also can have any no. of leading a's & b's

Q.7] Show that $(a+b)^* = (a+b)^* + (a+b)^*$

Sol^n → Let $r_1 = (a+b)^*$

$L(r_1) = \{\epsilon, a, b, aa, bb, ab, ba, ---\}$ — ①

Let $r_2 = (a+b)^* + (a+b)^*$

$u = (a+b)^*$

& $v = (a+b)^*$

$\therefore$ ~~U.T~~ $L(u) = \{\epsilon, a, b, aa, bb, ab, ba, ---\}$ — ②

$L(v) = \{\epsilon, a, b, aa, bb, ab, ba, ---\}$ — ③

As $r_2 = u + v$

$= L(u) \cup L(v)$

$= \{\epsilon, a, b, aa, bb, ab, ba\} \cup \{\epsilon, a, b, aa, bb, ab, ba\}$ from ②, ③

$L(r_2) = \{\epsilon, a, b, aa, bb, ab, ba, ---\}$ — ④

As per eq^n ① & ④

$L(r_1) = L(r_2)$

$\therefore \boxed{(a+b)^* = (a+b)^* + (a+b)^*}$

Hence proved.

Q.8] Show that $(a\cdot b)^* \neq (a)^* \cdot (b)^*$

Sol^n → Let $r_1 = (a\cdot b)^*$; $r_2 = a^* \cdot b^*$

~~$\therefore L(r_1) = \{ababab$~~

$L(r_1) = \{\epsilon, ab, abab, ababab, ---\}$ — ①

For $L(r_2) = a^* \cdot b^*$

Let $u = a^*$, $v = b^*$

$\therefore L(u) = \{\epsilon, a, aa, aaa, ---\}$ — ②

& $L(v) = \{\epsilon, b, bb, bbb, ---\}$ — ③

$L(r_2) = L(u) \cup L(v)$

$= \{\epsilon, a, aa, aaa, ..\} \cup \{\epsilon, b, bb, bbb ..\}$ ... from ② & ③

$L(r_2) = \{\epsilon, a, b, aa, bb, aaa, bbb ----\}$ — ④

from eq^n ① & ④

$L(r_1) \neq L(r_2)$

$\therefore \boxed{(a\cdot b)^* \neq a^* \cdot b^*}$

HENCE PROVED.

HW Q.9] Show that $(a^* \cdot b^*)^* = (a+b)^*$

Sol^n → Let $r_1 = (a^* \cdot b^*)^*$; $r_2 = a^* \cdot b^*$

∴ $L(r_1') = a^* \cdot b^*$

Let $u = a^*$; $v = b^*$

∴ $L(u) = \{\epsilon, a, aa, aaa, ---\}$

$L(v) = \{\epsilon, b, bb, bbb, ---\}$

$L(r_1') = L(u) \cup L(v)$

$= \{\epsilon, a, aa, aaa --\} \cup \{\epsilon, b, bb, bbb, --\}$

$L(r_1) = \{\epsilon, a, b, aa, bb, aaa, bbb, ---\}$.

$L(r_1) = (a^* \cdot b^*)^*$

$= \{\epsilon, a, b, aa, bb, aaa, bbb, ab, ba, aba, abb, baa, ---\}$ — ①

$\boxed{L(r_1) = [L(r_1')]^*}$

② & ③

$L(r_2) = (a+b)^*$

$L(r_2) = \{\epsilon, a, b, aa, bb, ab, ba, ---\}$ — ②

from eq^n ① & ②

$L(r_1) = L(r_2)$

i.e. $\boxed{(a^* \cdot b^*)^* = (a+b)^*}$

Hence proved.

Page No.
Date (K.L.P. Mishra) 130

* IDENTITIES FOR REGULAR EXPRESSIONS

$I_1 \rightarrow \phi + R = R$

$I_2 \rightarrow \phi R = R\phi = \phi$

· $I_3 \rightarrow \lambda R = R\lambda = R$

· $I_4 \rightarrow \lambda^* = \lambda$ & $\phi^* = \lambda$

$I_5 \rightarrow R + R = R$

$I_6 \rightarrow R^* . R^* = R^*$

$I_7 \rightarrow RR^* = R^* R$

$I_8 \rightarrow (R^*)^* = R^*$

· $I_9 \rightarrow \wedge + RR^* = R^* = \wedge + R^* R$

· $I_{10} \rightarrow (PQ)^* P = P(QP)^*$

· $I_{11} \rightarrow (P+Q)^* = (P^* Q^*)^* = (P^* + Q^*)$

· $I_{12} \rightarrow (P+Q)R = PR + QR$ &
$R(P+Q) = RP + RQ$

Q.1] Prove $(1+00^*1) + (1+00^*1)(0+10^*1)^*(0+10^*1)$
$= 0^*1(0+10^*1)^*$

Sol^n $\rightarrow$ LHS $= (1+00^*1) + (1+00^*1)(0+10^*1)^*(0+10^*1)$

$= (1+00^*1)[\wedge + (0+10^*1)(0+10^*1)]$ .... (I_{12})

$= (1+00^*1)(0+10^*1)^*$ .... where $R = 0+10^*1$ & $R^* = (0+10^*1)^*$ (I_9)

$= (\wedge + 00^*)1(0+10^*1)$

$= 0^*1(0+10^*1)^*$

Page No.
Date

Q.2] Prove that $(1+011)^* = \wedge + 1^*(011)^*(1^*(011)^*)^*$

Sol^n → LHS $= \wedge + 1^*(011)^*(1^*(011)^*)^*$

$\{ I_9 = \lambda + R \cdot R^* = R^* \; \ldots \; I_9 \}$

i.e. $= [1^*(011)^*]^*$ .... where $R = (1^*(011)^*)^*$

$$\left\{ \begin{array}{l} I_{11} = (P+Q)^* = P^* + Q^* = (P^* Q^*)^* \\ \text{i.e. } P^* = 1^*, \; Q^* = (011)^* \end{array} \right\}$$

$= (1+011)^*$

$=$ R.H.S.

Q.3]

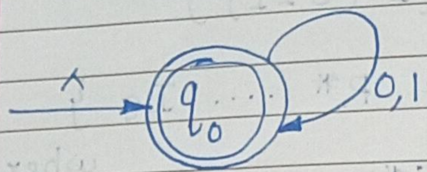
* Arden's Theorem

If P, Q & R are regular expressions then,

(i) If $R = P + RQ$ or $R = RQ + P$ then R can be simplified as $\boxed{R = PQ^*}$

(ii) If $R = P + QR$ or $R = QR + P$ then R can be simplified as $\boxed{R = Q^*P}$

Q.1) Construct Regular Expression corresponding to the state diagram given below.



Sol^n → From the above given state diagram,

$$q_0 = \wedge + q_0 0 + q_0 1$$

$$q_0 = \wedge + q_0 (0+1)$$

By Arden's theorem,

~~R+P~~ $R = P + RQ$.

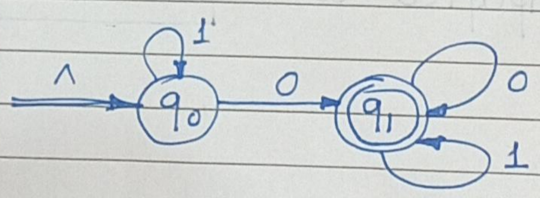
where $R = q_0$, $P = \wedge$, $Q = (0+1)$

$$\therefore \quad R = PQ^*$$

$$q_0 = \wedge (0+1)^*$$

i.e. $\boxed{q_0 = (0+1)^*}$ ... using I_3

Q.2] Construct a R.E. corresponding to the state diagram given below.



Page No.
Date

Sol^n → From the above given state diagram,

$$q_0 = \lambda + q_0 1 \quad \text{——} \quad (1)$$

$$q_1 = q_0 0 + q_1 0 + q_1 1$$
$$= q_0 0 + q_1 (0+1) \quad \text{——} \quad (2)$$

By Arden's theorem, $R = P + RQ \Rightarrow R = PQ^*$

(1) ⇒ ∴ for q_0 ⇒ $q_0 = \wedge(1)^* = 1^*$ ... from I_3 — (3)

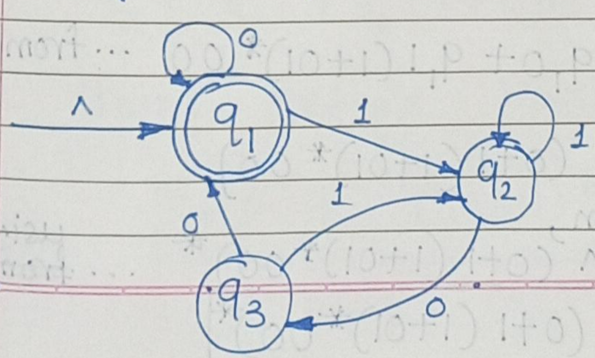
(2) ⇒ ~~for q_1 ⇒ $q_1 = \emptyset(0+1)^*$~~
$$q_1 = q_0 0 + q_1 (0+1)$$
from eq^n (3)
$$q_1 = 1^* 0 + q_1 (0+1)$$

Using Arden's theorem,
$$R = P + RQ$$
$$\Rightarrow R = PQ^*$$
where $P = 1^* 0$, $R = q_1$, $Q = (0+1)$

$$\therefore \boxed{q_1 = (1^* 0)(0+1)^*}$$

KLP Mishra Pg.151

Q.3] Construct a RE corresponding to the state diagram given below:-



Page No.			
Date			

Sol^n → q_1 From the diagram given above,

$$q_1 = \Lambda + q_1 0 + q_3 0 \quad \text{— ①}$$

$$q_2 = q_1 1 + q_2 1 + q_3 1 \quad \text{— ②}$$

$$q_3 = q_2 0 \quad \text{— ③}$$

∴ Consider eq^n ②

$$q_2 = q_1 1 + q_2 1 + q_3 1 0$$

$$q_2 = q_1 1 + q_2 1 + q_2 0 1 \quad \ldots \text{where } q_3 = q_2 0$$

$$q_2 = q_1 1 + q_2 (1 + 01)$$

Using Arden's theorem,

$$\boxed{q_2 = q_1 1 \cdot (1 + 01)^*} \quad \text{— ④}$$

Now,
Consider eq^n ①

$$q_1 = \Lambda + q_1 0 + q_3 0 \quad \ldots \text{from ③}$$

$$q_1 = \Lambda + q_1 0 + q_2 0 0$$

$$q_1 = \Lambda + q_1 0 + q_1 1 (1 + 01)^* 00 \quad \ldots \text{from ④}$$

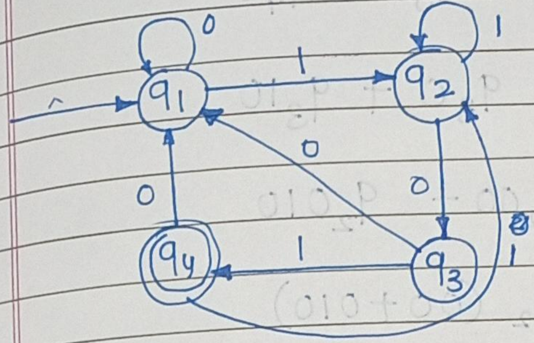
$$q_1 = \Lambda + q_1 (0 + 1 (1 + 01)^* 00)$$

By Arden's thrm,

$$q_1 = \Lambda (0 + 1 (1 + 01)^* 00)^* \quad \ldots \text{using } ~~\text{from}~~ \text{ I}_3$$

$$\boxed{q_1 = (0 + 1 (1 + 01)^* 00)^*}$$

Q.4) Find the Regular Expression corresponding to the figure given below:



$\Rightarrow$ From the diagram given above;

$$q_1 = \Lambda + q_1 0 + q_3 0 + q_4 0 \quad \text{—— (1)}$$

$$q_2 = q_1 1 + q_2 1 + q_4 1 \quad \text{—— (2)}$$

$$q_3 = q_2 0 \quad \text{—— (3)}$$

$$q_4 = q_3 1 \quad \text{—— (4)}$$

from (3), eq^n (4) becomes,

$$q_4 = \cancel{q_3}\, q_2 01 \quad \text{— (5)}$$

from (5)

$$q_2 = q_1 1 + q_2 1 + q_2 011$$

$$q_2 = q_1 1 + q_2 (1 + 011)$$

$\therefore$ Using Arden's theorem,

$$\boxed{q_2 = q_1 1 (1 + 011)^*} \quad \text{— (6)}$$

Consider eq^n 1

$$q_1 = \wedge + q_1 0 + q_3 0 + q_4 0$$

$$q_1 = \wedge + q_1 0 + q_3 0 + q_3 10$$

$$q_1 = \wedge + q_1 0 + q_2 00 + q_2 010$$

$$q_1 = \wedge + q_1 0 + q_2 (00 + 010)$$

$$q_1 = \wedge + q_1 0 + q_1 1 (1+011)^* (00+010) \quad \text{from } 6$$

$$q_1 = \wedge + q_1 (0 + 1 (1+011)^* (00+010))$$

$\therefore$ Using Arden's theorem,

$$R = P + RQ \Rightarrow R = PQ^*$$

$$q_1 = \wedge [(0+1(1+011)^* (00+010))]^*$$

But $\wedge P = P$

$$\therefore \boxed{q_1 = [0+1(1+011)^* (00+010)]^*} \quad -7$$

Put 7 in 6

$$\therefore q_2 = \{[0+1(1+011)^* (00+010)]^* 1 (1+011)^*\} \quad -8$$

Put 8 in 3

$$q_3 = q_2 0$$

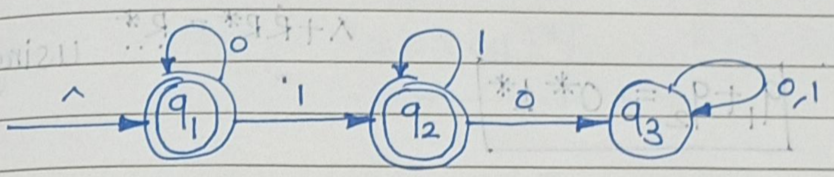
$$q_3 = \{[0+1(1+011)^* (00+010)]^* 1 (1+011)^*\} 0 \quad -9$$

Put (9) in (4) we get,

$q_4 = q_3 1$

$\therefore q_4 = \{[0+1(1+011)^*(00+010)]^* 1 (1+011)^*\} 01$

Q5] Describe in english the set accepted by Finite Automata whose transition diagram is given below:-



→ From the diagram given above,

$\cancel{q_1 = \Lambda q_1 0}$ $q_1 = \Lambda + q_1 0$ — (1)

$q_2 = q_1 1 + q_2 1$ — (2)

$q_3 = q_2 0 + q_3 0 + q_3 1$ — (3)

from (1)

$q_1 = \Lambda + q_1 0$

$\therefore$ Using Arden's theorem,

$R = P + RQ$

$\therefore R = P.Q^*$

$\Lambda . Q = Q$

$q_1 = \Lambda (0)^*$

$\therefore \boxed{q_1 = 0^*}$ — (4)

Put (4) in (2)

$q_2 = 0^* 1 + q_2 1$

Using Arden's theorem,

$\boxed{q_2 = 0^* 1 (1)^*}$ — (5)

As q_1 & q_2 are final state

∴ The final Regular Expression is as follows:

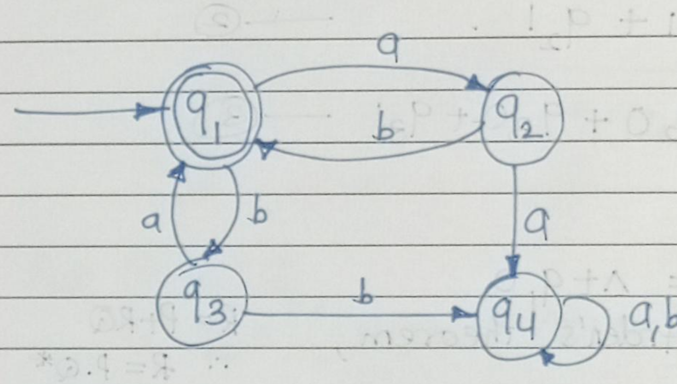
$$\boxed{q_1 + q_2 = 0^* + 0^* 1 (1)^*}$$

$$\cancel{q_1 + q} \quad q_1 + q_2 = 0^* \ (\Lambda + 1 1^*)$$

$\Lambda + RR^* = R^*$ using I_9

$$\boxed{q_1 + q_2 = 0^* 1^*}$$

Q.6] Construct a Regular Expression corresponding to the state diagram given below:-



⟹ From the diagram given above,

$q_1 = \Lambda + q_2 b + q_3 a$ —— ①

$q_2 = q_1 a$ —— ②

$q_3 = q_1 b$ —— ③

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

$q_1 = \wedge + q_1 ab + q_1 ba$ ... from ① & ②

$q_1 = \wedge + q_1 (ab+ba)$

Using Arden's theorem,

$R = P + RQ$ i.e. $R = PQ^*$

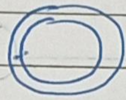
$\therefore q_1 = \wedge \cdot (ab+ba)^*$ ... by I_3

$\therefore \boxed{q_1 = (ab+ba)^*}$

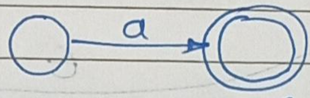
* Regular Expression to Finite Automata

1. RE to NFA

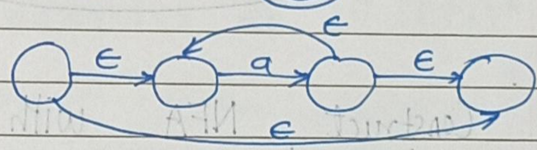
⟨i⟩ ε →



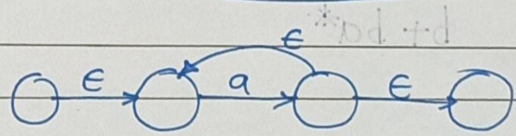
⟨ii⟩ a →



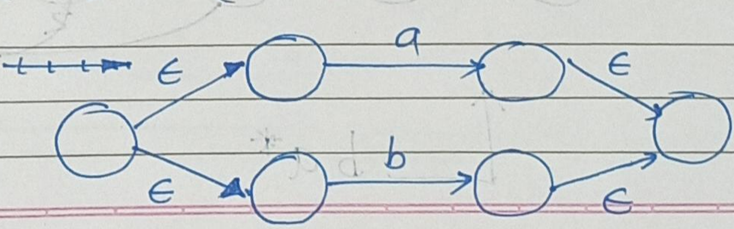
⟨iii⟩ a* →



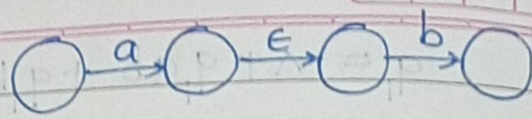
⟨iv⟩ a⁺ →



⟨v⟩ (a|b) or (a+b)



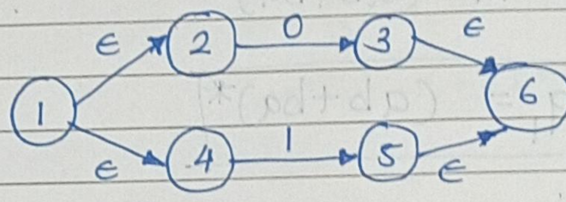
(vi) ab →



EXAMPLES

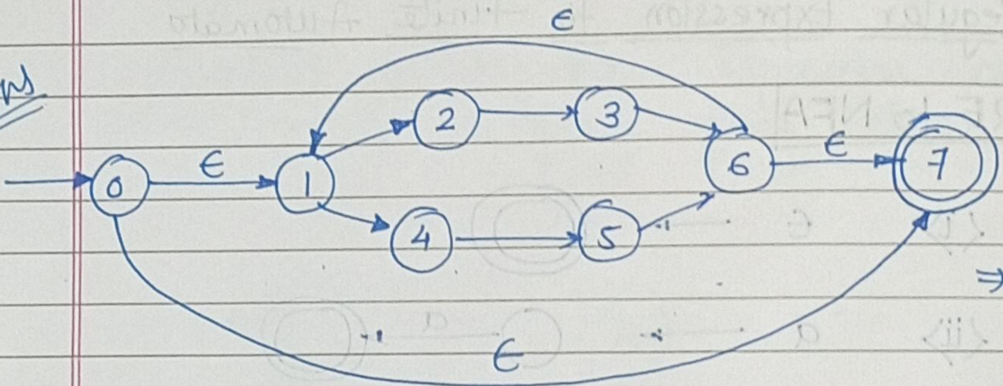
Q.1] Construct NFA with 'ϵ'-moves for regular expressions $(0+1)^*$

→



⇒ (0+1)

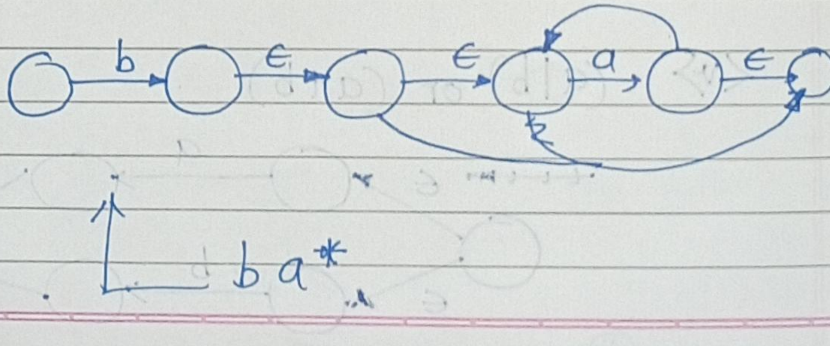
Ans



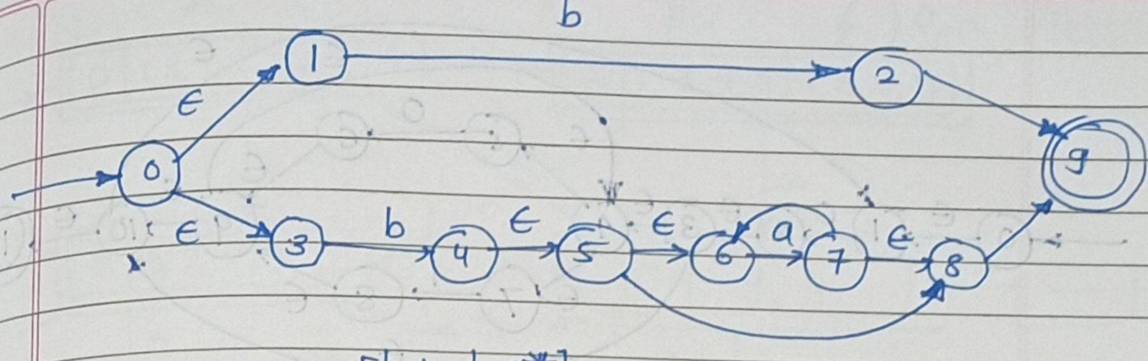
⇒ $(0+1)^*$

Q.2] Construct NFA with 'ϵ'-moves for RE $b+ba^*$

→

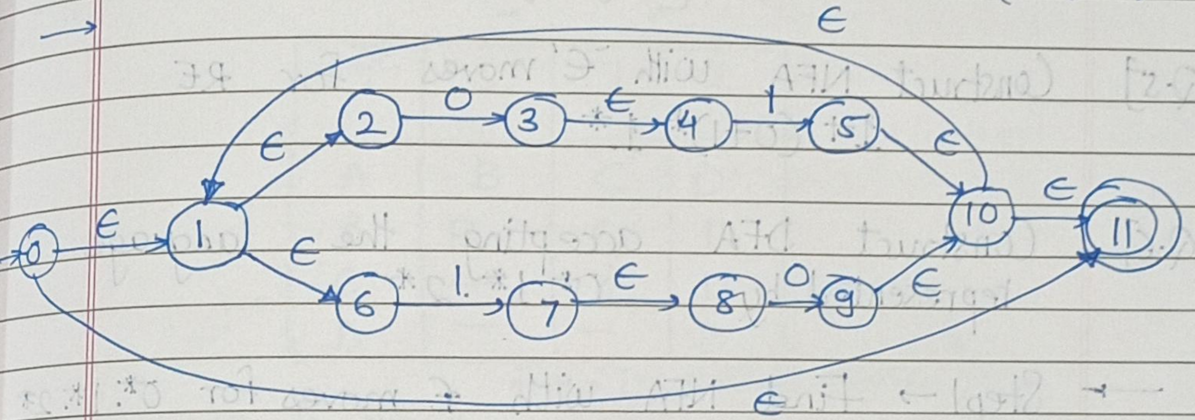


Page No.
Date

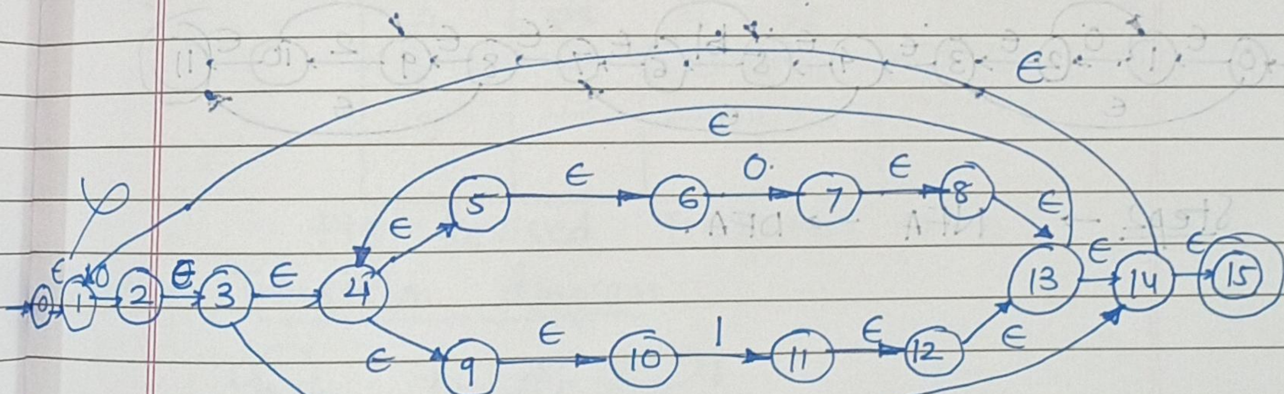


[b+ ba*]

Q.3] Construct NFA with ϵ-moves for RE (01+10)*



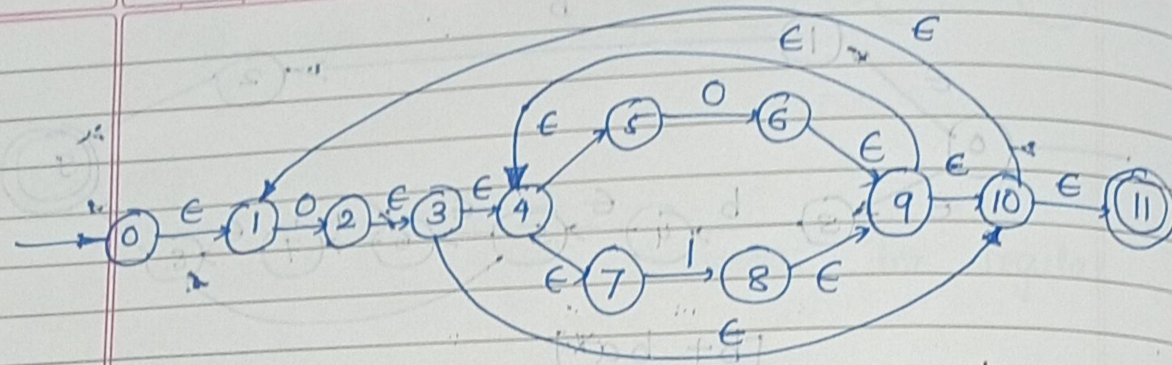
Q.4] Construct NFA with 'ϵ'-moves for RE (0 (0+1)*)+



(0+1)*

0(0+1)*

(0(0+1)*)+



$(0+1)^*$

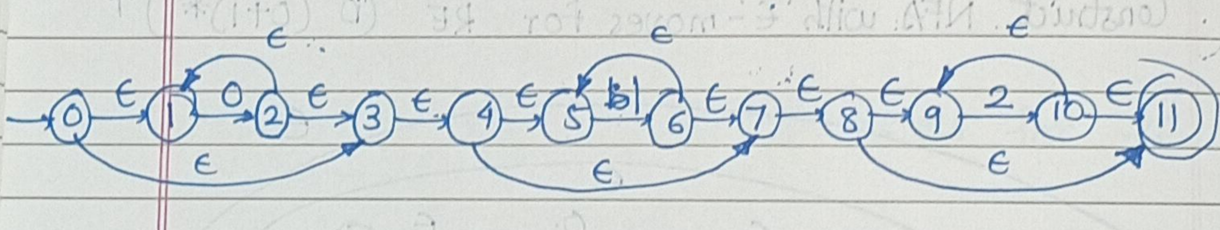
$0\cdot(0+1)^*$

$[0\cdot(0+1)^*]^*$

Q.5] Construct NFA with 'ε' moves for RE
$1 + (0+1)^* \cdot 1^*$

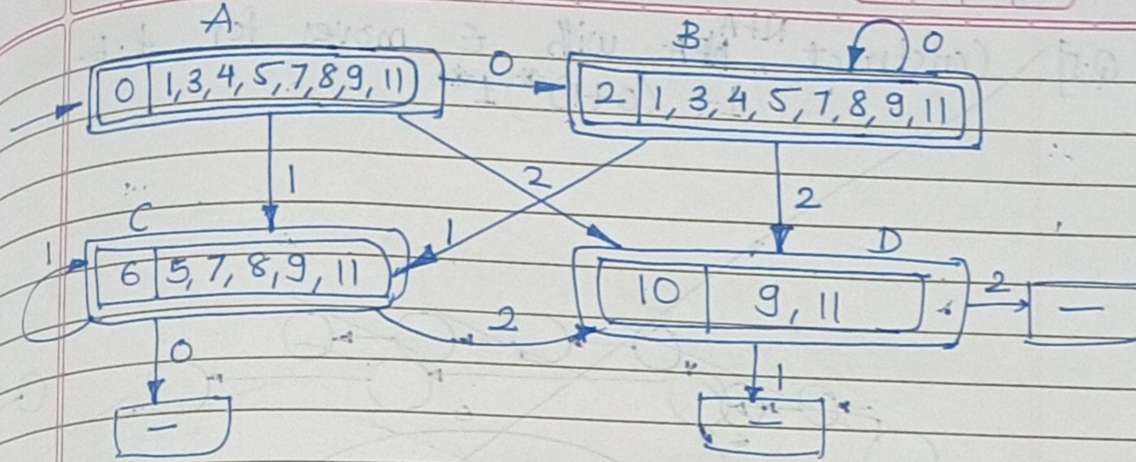
Q.6] Construct DFA accepting the language represented by $0^* \cdot 1^* \cdot 2^*$

→ Step1 → Find NFA with ε moves for $0^* \cdot 1^* \cdot 2^*$



Step2 → NFA → DFA.

Page No.	
Date	



Step 3 → Transcription table.

δ \ Σ	0	1	2
A	B	C	D
B	B	C	D
C	—	C	D
D	—	—	D

Fig. Transition table

Replace B by A.

δ \ Σ	0	1	2
A	A	C	D
C	—	C	D
D	—	—	D

Fig. Revised Transition table.

Step 4 → Transition diagram.

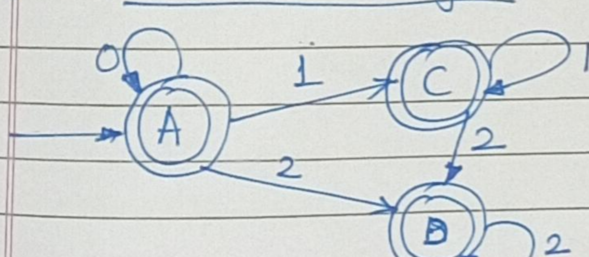
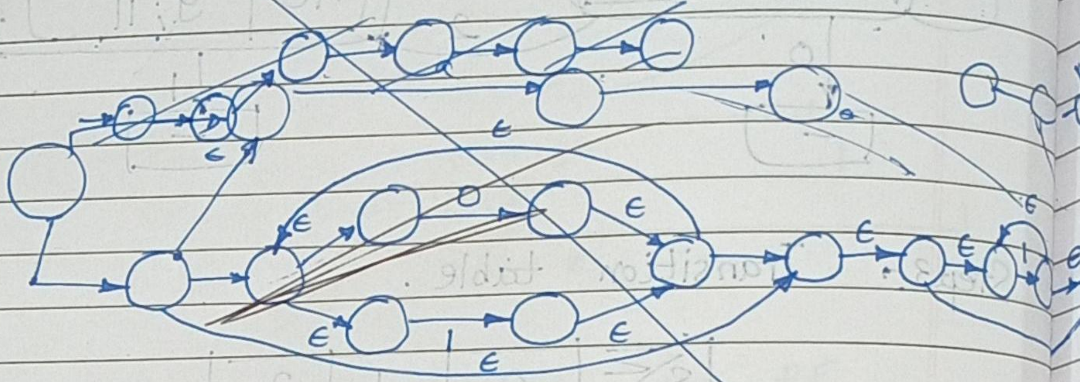
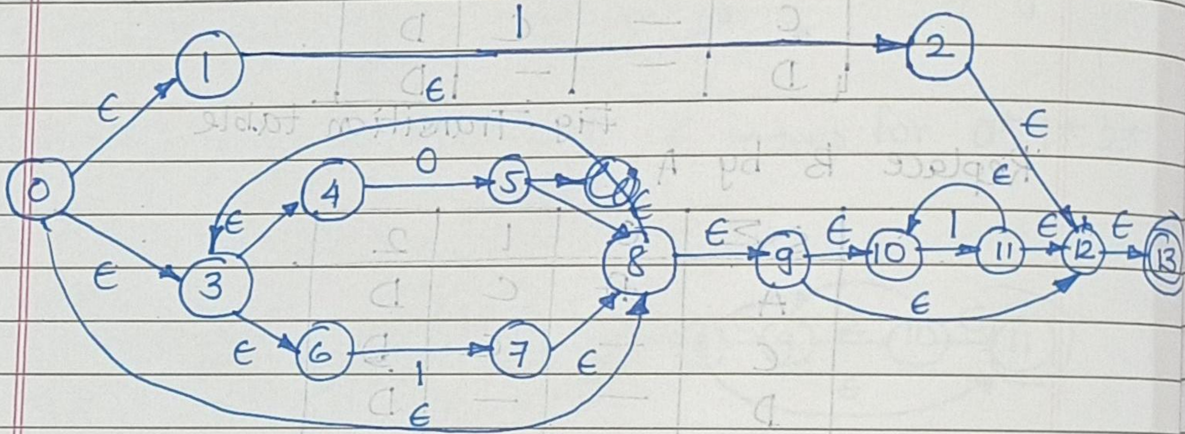


Fig. Transition Diagram.

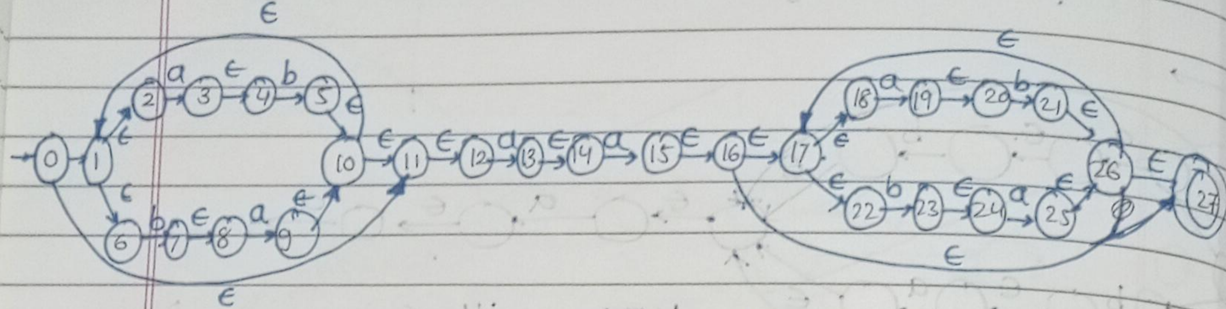
Q.5] Construct NFA with ε moves for R.E.
$1+(0+1)^* 1^*$



Q.5] Construct NFA with ε moves for RE
$1+(0+1)^* 1^*$



Q.7] Construct DFA accepting language represented by $(ab/ba)^*aa(ab/ba)^*$



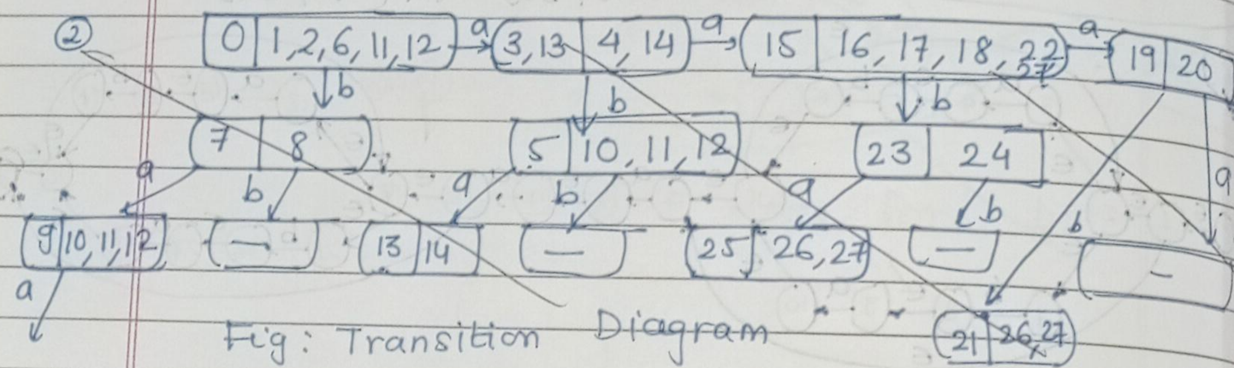
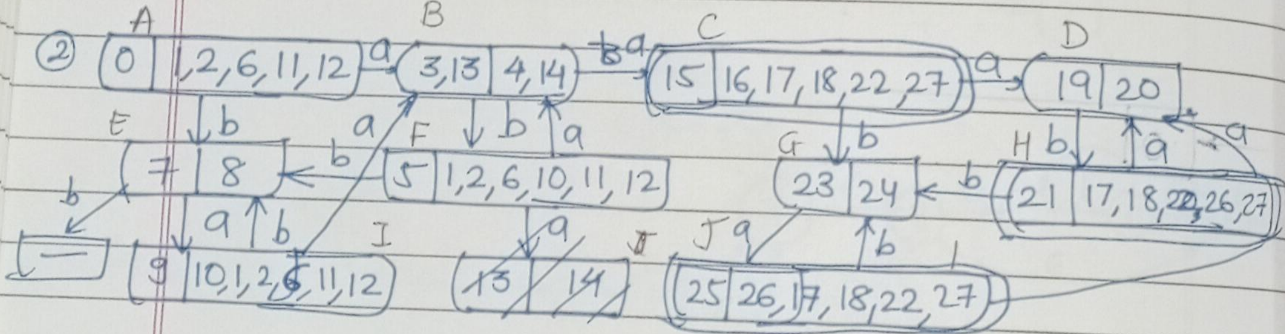


Fig: Transition Diagram



Q \ Σ	a	b
→ A	B	E
B	C	F
C	D	G
D	–	H
E	I	–
F	B	E
G	J	–
H	D	G
I	B	E
J	D	G

Fig. Transition table

Revised Transition Table.

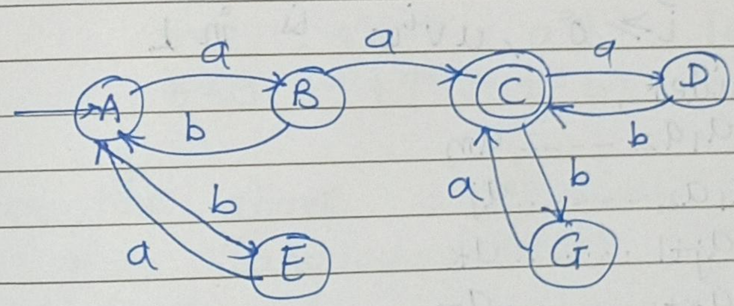
$Q \backslash \Sigma$	a	b
→ A, F, I	B	E
B	C	F
C, H, J	D	G
D	–	H
E	I	–
G	J	–

Replace H & J by C
&
Replace F & I by A

$Q \backslash \Sigma$	a	b
→ A	B	E
B	C	A
C	D	G
D	–	C
E	A	–
G	C	–

Fig. Revised Transition table.

(4) Transition diagram



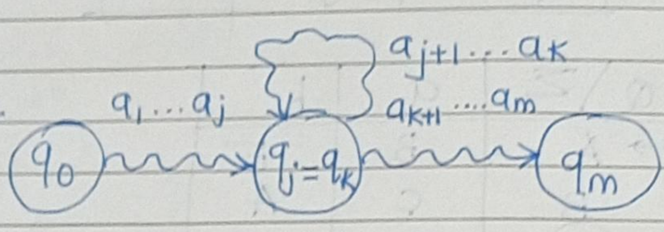
Q] State & prove Pumping Lemma for Regular Set / RE.

Page No.
Date

* Pumping Lemma for Regular Expression.

Statement: It states that ~~given~~ given any sufficiently long string accepted by an FSM we can find a substring near the beginning of the string that may be repeated (Pumped) as many times as we like & the resultant string will still be accepted by the same FSM.

Q] State



* Formal Statement of Pumping Lemma

Let L be a regular set, then there is a constant 'm' such that if 'z' is any word in 'L' & $|z| \geq n$ we may write $z = uvw$ in such a way that $|uv| \leq n$, $|v| \geq 1$ i.e. $1 \leq |v| \leq m$ & for all $i \geq 0$, uv^iw is in L

Consider,

$z = a_1 a_2 \dots\dots a_m$

$u = a_1 a_2 \dots\dots a_j$

$v = a_{j+1} \dots\dots a_k$

$w = a_{k+1} \dots\dots a_m$

Page No.
Date

Example

Q1] Prove that the following language is non-regular using pumping lemma.

$$\{a^n b^{n+1} \mid n > 0\}$$

Sol^n →

Steps: first find
1. The length of given language.
2. Assume the given language is regular.
3. Consider ~~the~~ one variable as pumping lemma constant.
4. Find the length of z by considering above pumping lemma constant
5. Calculate z by pumping lemma.
6. Consider $i = 2$ & find $|uv^iw|$
7. Consider the value of pumping lemma constant (from Step 3) as 1, 2, 3.

⟹ (1) Given: $n > 0$

for $n = 1$, $a^n b^{n+1} = a^1 b^{1+1} = a^1 b^2 \therefore$ length $= 3$

$n = 2$, $a^n b^{n+1} = a^2 b^3 \therefore$ length $= 5$

$n = 3$, $a^n b^{n+1} = a^3 b^4 \therefore$ length $= 7$

From the above observation we can find that the given language consists of string having Odd length.

(2) Assume that the given language,

$L = \{a^n b^{n+1} \mid n > 0\}$ is regular.

(3) Consider / Let 'l' be a constant of pumping lemma.

(4) Let $z = a^l b^{l+1}$
where,

$$\text{length of } z = |z| = l + l + 1$$
$$= \underline{\underline{2l+1}}$$

(5) By Pumping lemma constant we can write 'z' as $z = uvw$
where, $1 \leq |v| \leq l$ and
uv^iw for $i \geq 0$ is in 'L'.

(6) Let $i = 2$ and from pumping lemma we can write, $1 \leq |v| \leq l$

$$(2l+1) + 1 \leq |v| \leq l + (2l+1)$$
$$\cancel{2l+2}$$
$$(2l+2) \leq |v| \leq (3l+1)$$

(7) As $(2l+2) \leq |v| \leq (3l+1)$

(6) Let $i=2$ and from pumping lemma we can write, $1 \leq |v| \leq l$

$$(2l+1) + 1 \leq |uv \cdot v \cdot w| \leq l + (2l+1)$$

$$2l+2 \leq |uv^2w| \leq 3l+1$$

$\because \left\{ \begin{array}{c} z = uvw \\ |z| = 2l+1 \\ z = uv^iw \end{array} \right\}$

Page No.
Date

(7) $2l+2 \leq |uv^2w| \leq 3l+1$

$2l+1 < |uv^2w| < 3l+1+1$

$2l+1 < |uv^2w| < 3l+2$

$\because l=1$ put kiya
$2(1)+2 \leq |uv^2w| \leq 3(1)+1$
$4 \leq |uv^2w| \leq 4$
not possible

Let, $l=1$
$2(1)+1 \leq |uv^2w| \leq 3(1)+2$
$3 \leq |uv^2w| \leq 5$
i.e. $\underline{4}$
$\therefore$ length is even.

Let, $l=2$
$2(2)+1 \leq |uv^2w| \leq 3(2)+2$
$5 \leq |uv^2w| \leq 8$
i.e. 6, 7
$\therefore$ not always odd

Let $l=3$
$2(3)+1 \leq |uv^2w| \leq 3(3)+2$
$7 \leq |uv^2w| \leq 11$
i.e. 8, 9, 10 $\therefore$ not always odd.

Thus, the length of uv^2w is not always odd which means, uv^2w is not in 'L'

$\therefore$ The given language,

$L = \{a^n b^{n+1} \mid n > 0\}$ is <u>non-regular</u>.

Q.2] Show that $L = \{0^i 1^i\}$ where $i \geq 1$ is not regular.

→ ① Given $i \geq 1$,
for $i = 1$, $0^i 1^i = 0^1 \cdot (1)^1 = 0^1 1^1$ length $= 2$
$i = 2$, $0^2 (1)^2 = 0^2 1^2$ → length $= 4$
$i = 3$, $0^3 (1)^3 = 0^3 1^3$ → length $= 6$

② From the above observation we can find that the given language consist of a string having <u>even</u> length.

③ Assume that
$L\{0^i 1^i\}$ is regular

④ Consider that 'l' be a constant of pumping lemma.
Let $z = 0^l 1^l$
where,
length of $z = |z|$
$= l + l$
$= 2l$

⑤ By pumping lemma constant, we can write 'z' as uvw
where, $1 \leq |v| \leq l$ &
uv^iw for $i \geq 1$ is in 'L'

⑥ Let $i = 2$ & from pumping lemma we can write, $1 \leq |v| \leq l$
$2l + 1 \leq |uv^2w| \leq 2l + l$

$\cancel{2l+l}$

$\cancel{2l+1 \leq |uv^2w| \leq 2l+l}$

$\cancel{2l \leq |uv^2w}$

(1) $2l+1 \leq |uv^2w| \leq 2l+l$

$2l < |uv^2w| < 2l+l+1$

Let $l = 1$

$2(1) < |uv^2w| < 2(1)+1+1$

$2 < |uv^2w| < 4$

i.e. 3

$\therefore$ length is odd.

Let $l = 2$

$2(2) < |uv^2w| < 2(2)+2+1$

$4 < |uv^2w| < 7$

i.e. 5, 6

$\therefore$ length is not always even

Let $l = 3$

$2(3) < |uv^2w| < 2(3)+3+1$

$6 < |uv^2w| < 10$

i.e. 7, 8, 9

$\therefore$ length is not always even.

Thus, the length of uv^2w is not always even which means, uv^2w is not in 'L'

$\therefore$ The given language,

$L = \{0^i 1^i\}, i \geq 1$ is <u>not regular</u>.

$(a^i)^2 \qquad a^{(i)^2}$

Q.3] Show that $L = \{(a^i)^2 \mid i \geq 1\}$ is not regular.

→ ① Given that $i \geq 1$

for $i=1$, $(a^i)^2 = (a^1)^2 = a^{1}$ ∴ length $= 1$

$i=2$, $(a^i)^2 = (a^2)^2 = a^4$ ∴ length $= 4$

$i=3$, $(a^i)^2 = (a^3)^2 = a^9$ ∴ length $= 9$

② From the above ~~ab~~ observation, we can find that the given language consist of a string having all perfect squares.

③ Assume that $L\{a^{(i)^2}\}$ is regular.

④ Consider that 'l' is a constant of pumping lemma

Let $z = a^{(l)^2}$

where,

length of $z = |z|$

$= l \cdot l$

$= l^2$

⑤ By pumping lemma constant, we can write 'z' as 'uvw'

where,

$1 \leq |v| \leq l$ &

uv^iw for $i \geq 1$ is in L

⑥ Let $i=2$ & from pumping lemma we can write,

$l^2 + 1 \leq |uv^2w| \leq l + l^2$

$$l^2 \leq |uv^2w| \leq l + l^2 + 1$$

(7) $l^2 < |uv^2w| < l + l^2 + 1$

Let $l = 1$

$1 < |uv^2w| < 3$ i.e. 2

$\therefore$ length is not a perfect square.

Let $l = 2$

$4 < |uv^2w| < (2+4+1)$

$4 < |uv^2w| < 7$

i.e. 5, 6

$\therefore$ length is not a perfect square

Let $l = 3$

$9 < |uv^2w| < (3+9+1)$

$9 < |uv^2w| < 13$

i.e. 10, 11, 12

$\therefore$ length is not a perfect square.

Thus, the length of uv^2w is not always a perfect square which means that uv^2w is not in 'L'

Hence the given language,

$L = \{a^{i^2}, i \geq 1\}$ is not regular.

$l^2+1 \leq v^2 \leq l^3$

$l^2 \leq v^2 \leq l^3+1$

Page No. | Date

Q.4] Show that $L = \{a^p \mid p$ ~~is~~ is prime$\}$ is not regular.

→ Given, $p = a^p$, p is prime.

$p = 2$, $a^p = a^2$ ∴ length = 2

for $p = 3$, $(a^p) = a^3$ ∴ length = 3

$p = 5$, $a^p = a^5$ ∴ length = 5

$p = 7$, $a^p = a^7$ ∴ length = 7

(2) From the above observation, we can find that the given language consist of a string having all prime numbers. / odd numbers.

(3) Assume that $L = \{a^p\}$ is regular

(4) Assume that 'l' is a constant of pumping lemma,

Let $z = a^p$

where, ~~$z = a$~~

length of $z = |z|$

$= l$

(5) By pumping lemma constant, we can write 'z' as 'uvw'

where

$1 \leq |uv^2w| \leq l$ &

uv^iw for $i \geq 1$ is in L

(6) Let $i = 2$ & from pumping lemma we can write,

$l + 1 \leq |uv^2w| \leq l + l$

$$l < |uv^2w| < 2l+1$$

(7) $l < |uv^2w| < 2l+1$

Let $l = 1$

$1 < |uv^2w| < 3$

i.e. 2

2 is ~~not~~ a prime number.

Let $l = 2$

$2 < |uv^2w| 5$

i.e. 3, 4

$\therefore$ length is not a prime no.

Let $l = 3$

$3 < |uv^2w| < 7$

i.e. 4, 5, 6

$\therefore$ length is not a prime no.

Thus, the length of uv^2w is not always a prime / odd no. which means that uv^2w is not in 'L'.

Hence the given ~~seq~~ language,

$L = \{a^p \mid p$ is a prime no.$\}$ is

not ~~a~~ regular.