Introduction to Automata Theory & Formal Languages

MCA II Semester

1. **Unit-I:**

Basic concepts of Automata Theory: Alphabets, Strings and Languages, Deterministic Finite Automata (DFA) and Nondeterministic Finite Automata (NFA) – Definition, Representation using Transition Tables and State Diagrams, Language of DFA and NFA. NFA with ε-transitions, Language of NFA with ε-transitions, Equivalence of NFA and DFA

38.

2. **Unit – II:**
3. **Regular Expressions and Languages:** Introduction, Definition of regular expression, Kleen's Theorem, Equivalence of regular expression and Finite Automata, Pumping Lemma for regular Languages, Closure properties of Regular Languages, Decision properties of Regular Languages, Finite Automata with Output: Moore and Mealy Machine, Equivalence of Moore and Mealy Machines.
4.
5. **Unit – III:**
6. **Non-Regular Grammars:** Definition of Grammar, Classification of Grammars, Chomosky's Hierarchy. Context Free Grammars (CFG) and Context Free Languages (CFL) - Definition, Examples, Derivation trees, Ambiguous Grammars, Simplification of Grammars, Normal forms of CFGs: CNF and GNF, Closure properties of CFLs, Decision Properties of CFLs, Pumping lemma for CFLs. Push Down Automata (PDA): Definition and Description, Language of PDA and its applications.
7.
8. **Unit – IV:**
9. **Turing Machines:** Introduction, Basic Features of a Turing Machine, Language of a Turing Machine, Variants of Turing Machine: Multitapes, Nondeterministic Turing Machine, Universal Turing Machine. Turing Machine as Computer of Integer functions, Halting problem of Turing Machine, Church-Turing Thesis
10.
11. **Unit – V:**
12. **Undecidability:** Introduction, Undecidable problems about Turing Machines, Rice's Theorem, Post's Correspondence problem (PCP) and Modified PCP. Tractable and Intractable Problems: P and NP, NPComplete Problems, Introduction to recursive function theory
13.
14.
15. **Text Books:**
16. 1. Introduction to Automata theory, Languages and Computation, J.E.Hopcraft, R.Motwani, and Ullman. 2nd edition, Pearson Education Asia
17. 2. Introduction to languages and the theory of computation, J Martin, 3rd Edition, Tata McGraw Hill
18. 3. Elements and Theory of Computation, C Papadimitrou and C. L. Lewis, PHI
19. 4. Mathematical Foundation of Computer Science, Y.N.Singh, New Age International

Regular Expression and Regular Set

* **Regular Expression:-** Regular expressions are useful ~~for~~ for representing certain set of strings in an algebric fashion. R.E. describe the languages accepted by finite state automata.

* Recurssive Defination of R.E. over Σ is as follows:

(1) Any terminal symbol (ie an element of Σ, $\wedge$ and ϕ are regular expressions)

(2) Union of two regular expressions R_1 and R_2, written as $R_1 + R_2$, is also a R.E.

(3) Concatenation of two regular expressions R_1 and R_2, written as $R_1 R_2$ is also a R.E.

(4) The iteration (or closure) of a Regular expression R, written as R^*, is also an R.E.

(5) If R is a R.E., then (R) is also a R.E.

(6) The regular expressions over Σ are precisely those obtained recursively by application of the rules 1-5.

* **Regular Expressions** - It is a way of representing Regular Language. Basically, R.E. is an expression of strings and operators, like

(i) $*$ (Kleene Closure) $[a^*]$

(ii) $+$ (Positive Closure) $[a^+]$

(iii) $\cdot$ (Concatenation) $[a \cdot b]$

(iv) $+$ (Union) $[a+b]$

Note - (1.) R.E. is said to be valid iff it can be derived from the primitive R.E. by a finite number of application of the rule r_1^*, r_1^+, $r_1 \cdot r_2$, $r_1 + r_2$

(2.) If Σ is a given alphabet then, ϕ, ϵ / λ, $a \in \Sigma$ are the primitive regular expression.

*** Kleene Star / closure / operator -**

It is ~~[illegible]~~ undetermined power, represent infinite no of terms those can be made including empty string. It is denoted by *.

*** Kleene Plus / Positive / Positive closure -**

It is undetermined power, represent infinite no of terms that can be made except empty string. It is denoted by +.

* **Power of Alphabet** - determines that the strings made from alphabet will be of length equal to power of alphabet.

- $\Sigma = \{a, b\}$ $\quad \Sigma^2 = \{a, b\}^2 \quad \Sigma^2 = \{aa, ab, ba, bb\}$

$\Sigma^* = \{a, b\}^* \quad \Sigma^* = \{\wedge, a, b, aa, ab, ba, bb, aaa, bbb, ---\}$

$\Sigma^+ = \{a, b\}^+ \quad \Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, bbb, ---\}$

* **Power of String** - determines the length of string.

$ba^2b = baab$

$ba^*b = bb$ or bab or $baab$ or $baaaaab$

$ba^+b = bab$ or $baab$ or $baaaaab$

* **KLEENE CLOSURE** - Given an alphabet Σ, we wish to define a language in which any string of letters from Σ is a word, even the null string. This language is called the closure of the alphabet. It is denoted by writing a star (an asterisk) on the alphabet as a super-script, Σ^*, this notation is sometimes known as Kleene Star.

e.g; If $\Sigma = \{x\}$ then $\Sigma^* = \{\Lambda, x, xx, xxx, \ldots\}$

If $\Sigma = \{0, 1\}$ then $\Sigma^* = \{\Lambda, 0, 1, 00, 01, 10, 11, \ldots\}$

* **KLEENE PLUS** - It is denoted by Σ^+. It means that input alphabet Σ excluding null string (Λ).

i.e., $\Sigma^+ = \Sigma^* - \{\Lambda\}$

Q.(J.T. 108) Construct a transition system ...

If $|S| = 0$, it is called an empty string (denoted by λ or ϵ).

* **Kleene Star:-** The Kleene Star, Σ^*, is a unary operator on a set of symbols or strings, Σ, that gives the infinite set of all possible strings of all possible lengths over Σ including λ.

$\Sigma^* = \Sigma_0 \cup \Sigma_1 \cup \Sigma_2 \cup \ldots \Sigma_p$, where Σ_p is the set of all possible strings of length p.

e.g.:- If $\Sigma = \{a, b\}$, $\Sigma^* = \{\lambda, a, b, aa, bb, ab, ba, \ldots\}$

* **Kleene Closure/Plus:-** The set of Σ^+ is the infinite set of all possible strings of all possible lengths over Σ excluding λ.

$\Sigma^+ = \Sigma_1 \cup \Sigma_2 \cup \Sigma_3 \cup \ldots$

$\Sigma^+ = \Sigma^* - \{\lambda\}$

e.g.:- If $\Sigma = \{a, b\}$, $\Sigma^+ = \{a, b, aa, bb, ab, ba, \ldots\}$

* **Language:-** A language is a subset of Σ^* for some alphabet Σ. It can be finite or infinite.

e.g.:- If the language takes all possible strings of length 2 over $\Sigma = \{a, b\}$, then
$L = \{ab, bb, ba, aa\}$

x ——— x ——— x

* **Deterministic Finite Automata (DFA):-** In DFA, for each input symbol, one can determine the state to which the machine will move.

Construction of Finite Automata Equivalent to a Regular Expression ①

Step (1.) :- Construct a transition graph or transition system equivalent to given regular expression using Λ-moves.

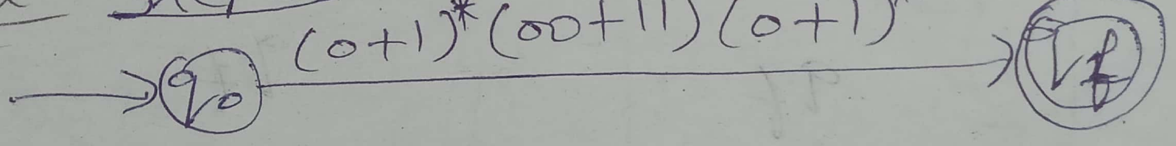
Step (2.) :- Construct the transition table for the transition graph obtained in step (1)

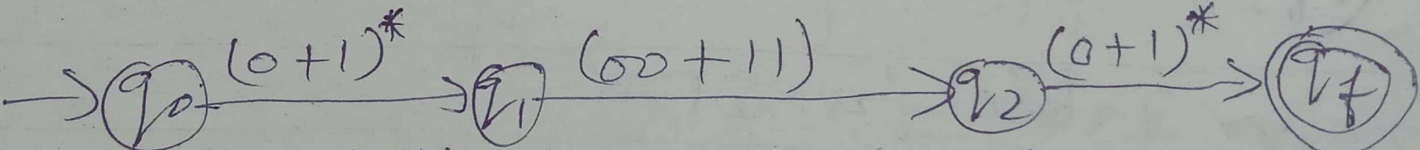
Now, construct the DFA.

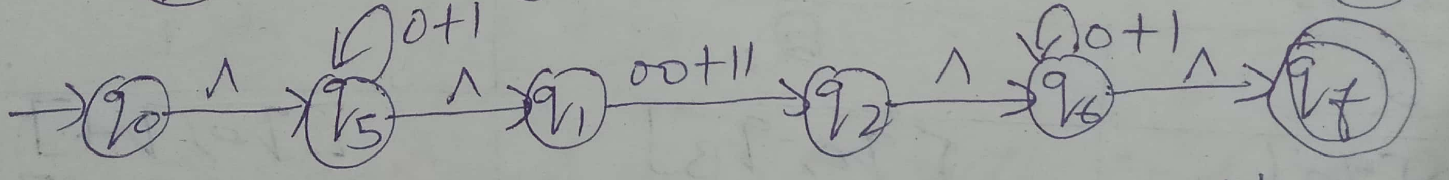
Ex. (4.13) KLP (91) Construct an FA equivalent to the regular expression.

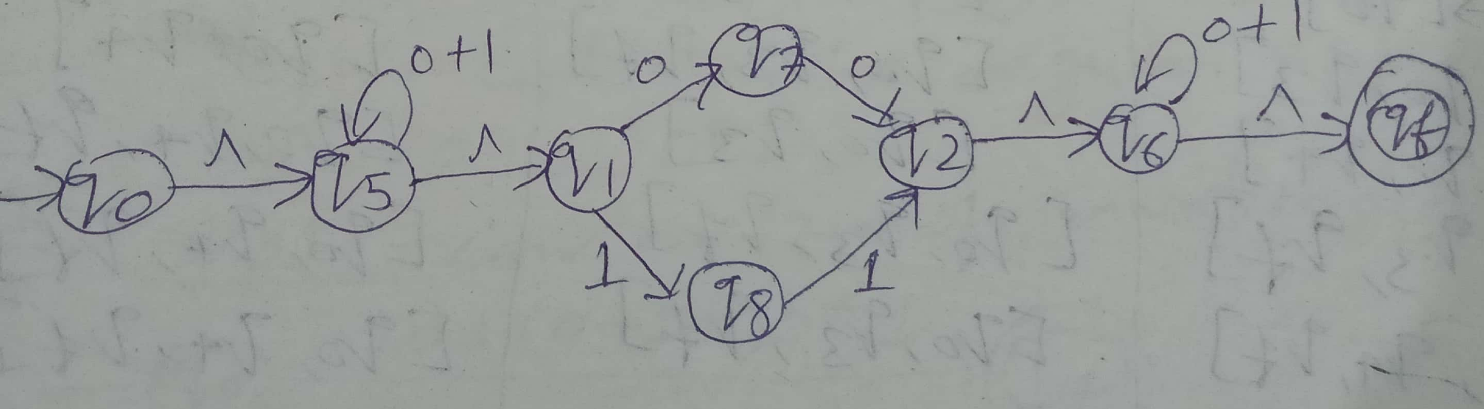
$$(0+1)^*(00+11)(0+1)^*$$

Sol^n. - Step (1.) -

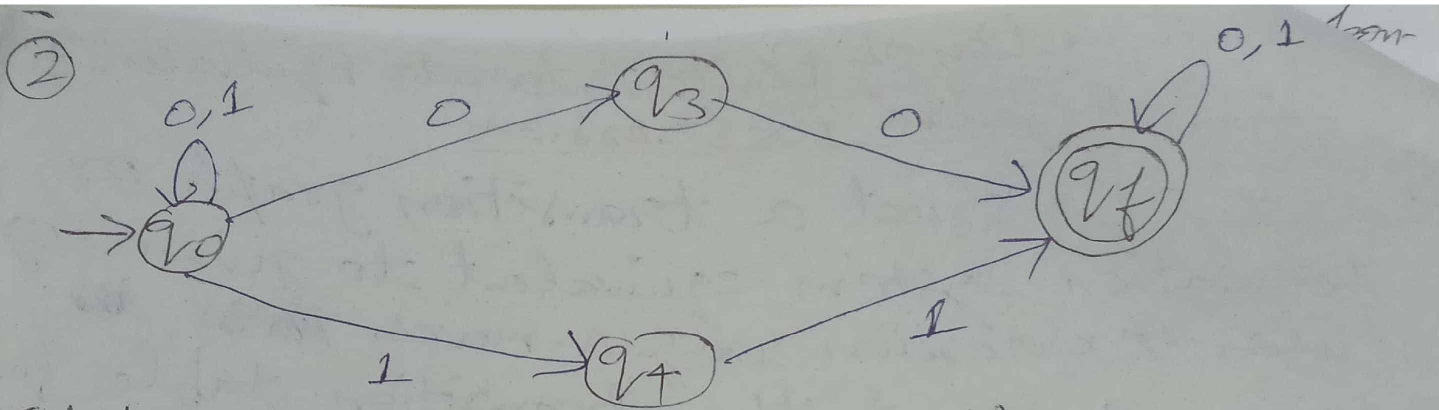








(2)



Step(2):- (Construction of DFA)

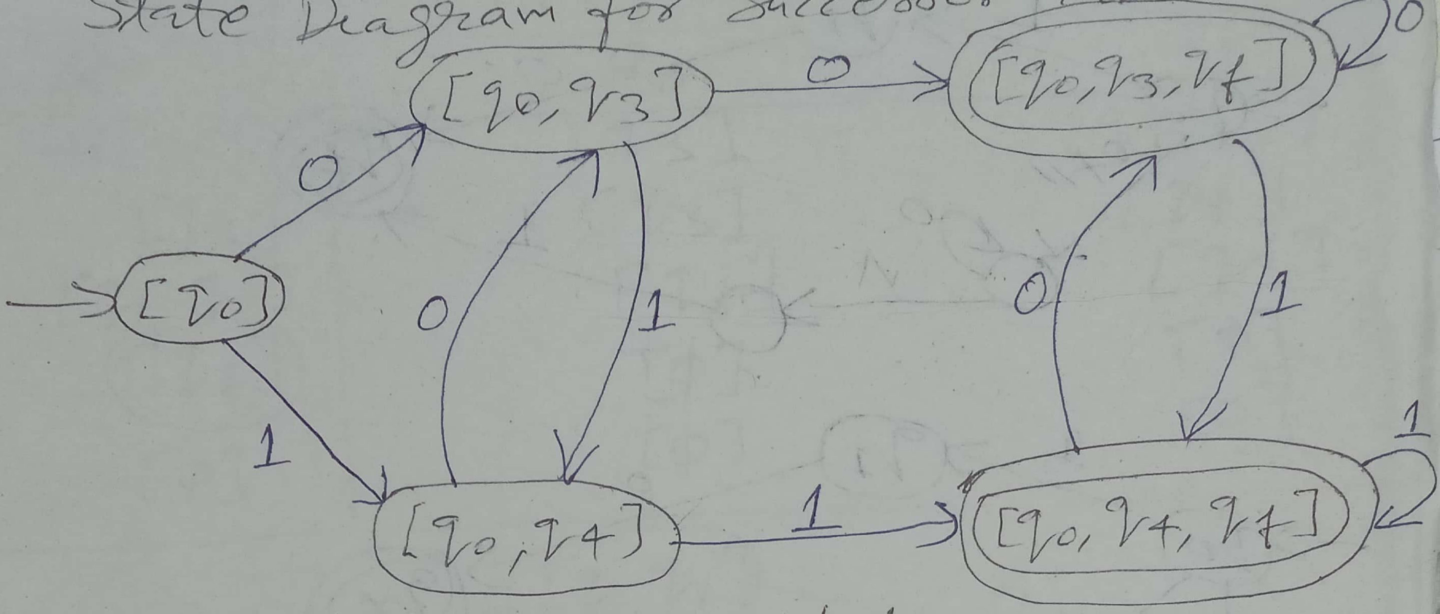
Transition Table for NFA

State/Σ	0	1
→q_0	q_0, q_3	q_0, q_4
q_3	q_f	—
q_4	—	q_f
q_f	q_f	q_f

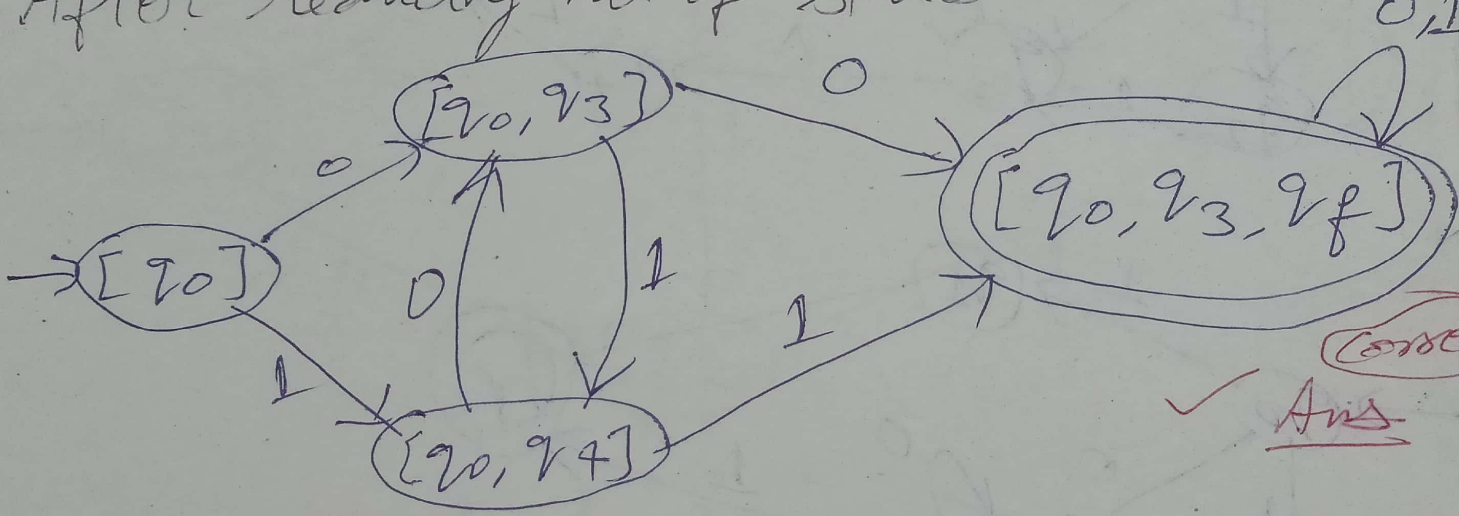
Successor Table for DFA

Q	Q_0	Q_1
→[q_0]	[q_0, q_3]	[q_0, q_4]
[q_0, q_3]	[q_0, q_3, q_f]	[q_0, q_4]
[q_0, q_4]	[q_0, q_3]	[q_0, q_4, q_f
[q_0, q_3, q_f]	[q_0, q_3, q_f]	[q_0, q_4, q_f
[q_0, q_4, q_f]	[q_0, q_3, q_f]	[q_0, q_4, q_f

State Diagram for Successor Table



After reducing no. of states-



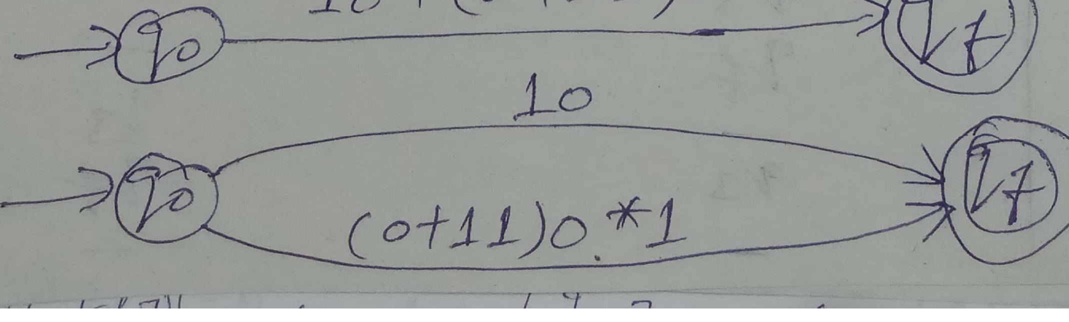
Correct

Ans

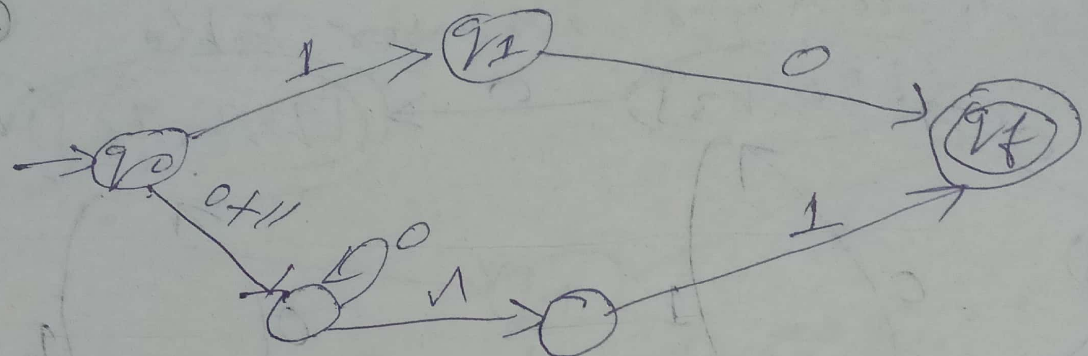
Ex (4.14) RLP(93) Construct a DFA with reduced state equivalent to the regular expression

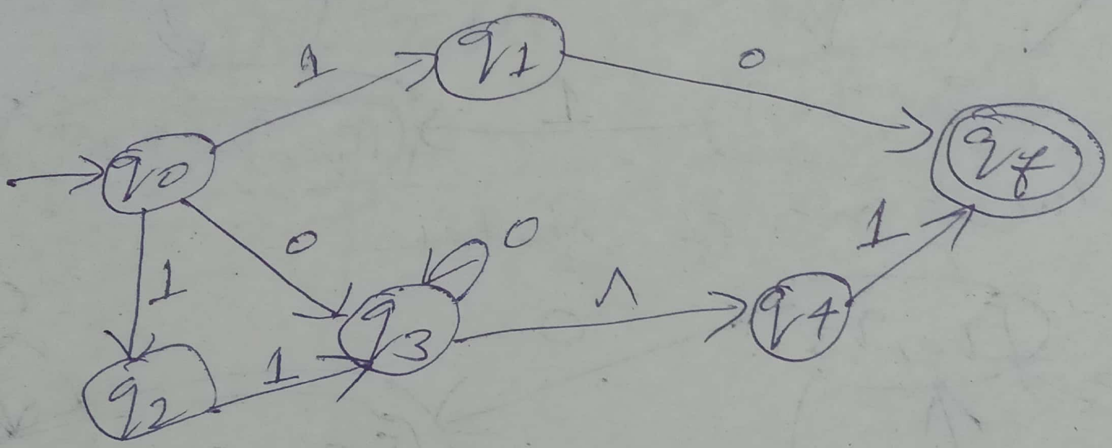
$10 + (0 + 11)0^*1$

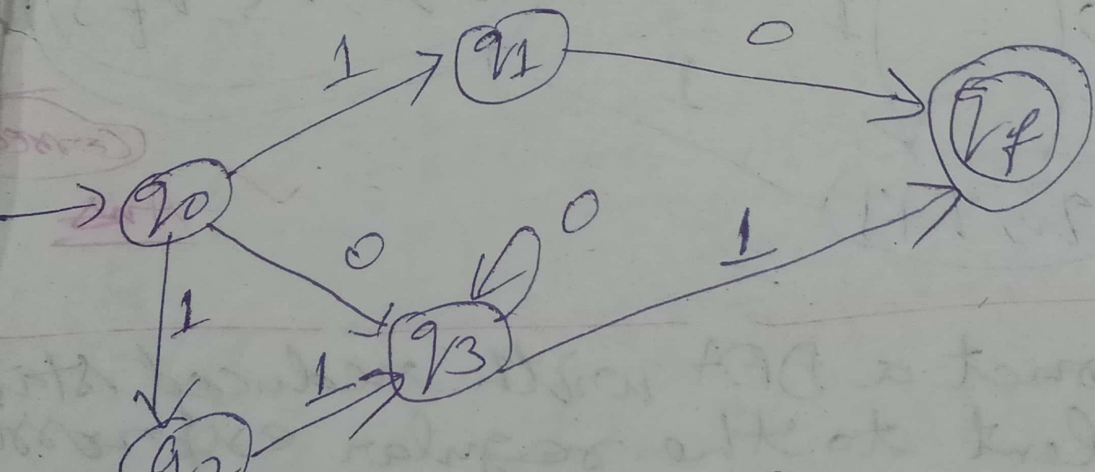
Soln - Step(1) - Construction of Transition Graph.



(2)
(4)





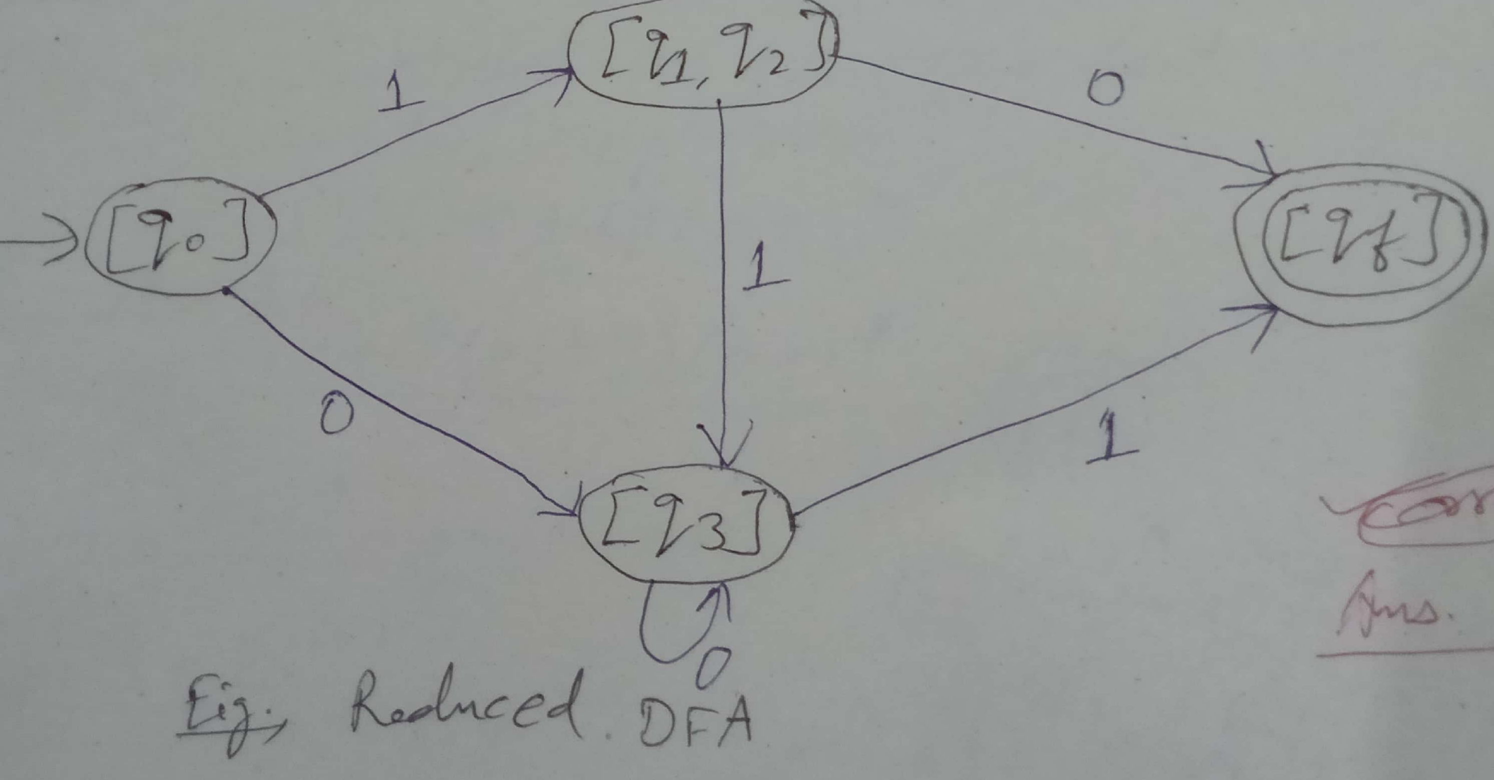


Transition Table

State/Σ	0	1
→q_0	q_3	q_1, q_2
q_1	q_f	—
q_2	—	q_3
q_3	q_3	q_f
q_f	—	—

Transition Table for DFA

Q	Q_0	Q_1
$\rightarrow [q_0]$	$[q_3]$	$[q_1, q_2]$
$[q_3]$	$[q_3]$	$[q_f]$
$[q_1, q_2]$	$[q_f]$	$[q_3]$
$[q_f]$	$[\phi]$	$[\phi]$
$[\phi]$	$[\phi]$	$[\phi]$

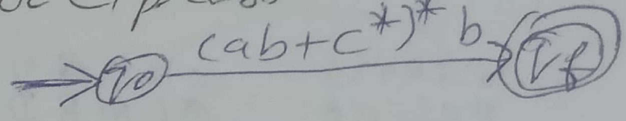


Correct Ans.

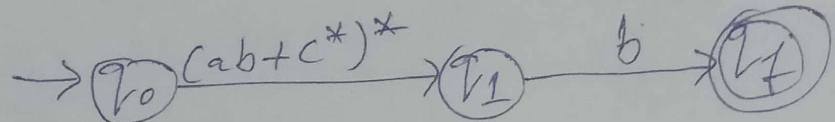
Fig: Reduced DFA

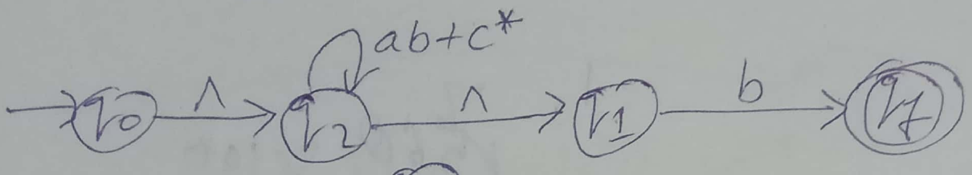
Q-(9.) KLP(108) Construct a transition system corresponding to the Regular expression

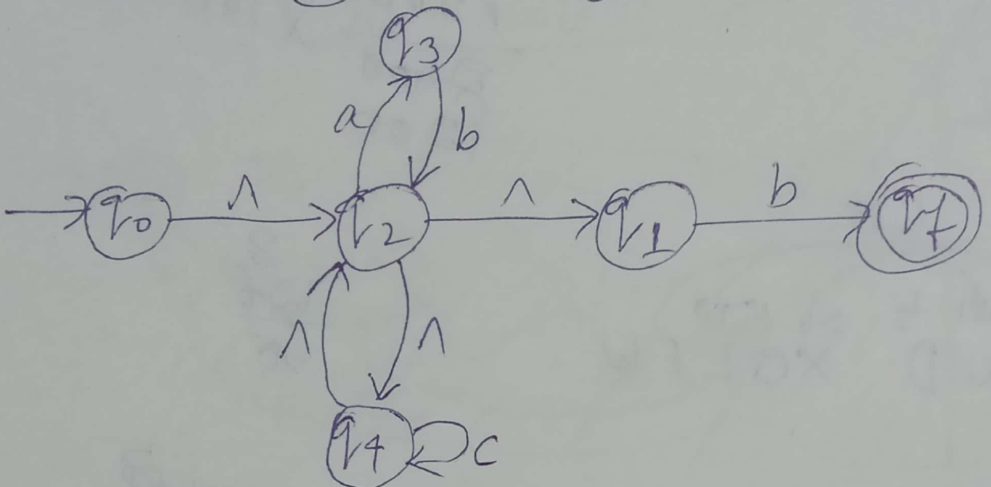
(i) $(ab+c^*)^* b$

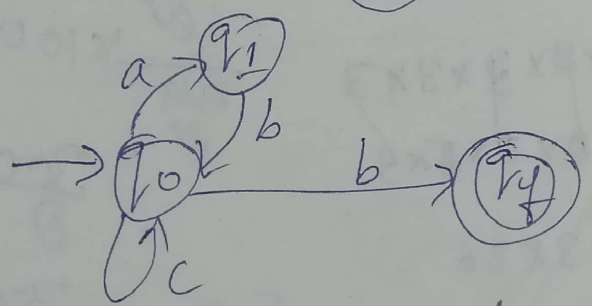


Sol^n -









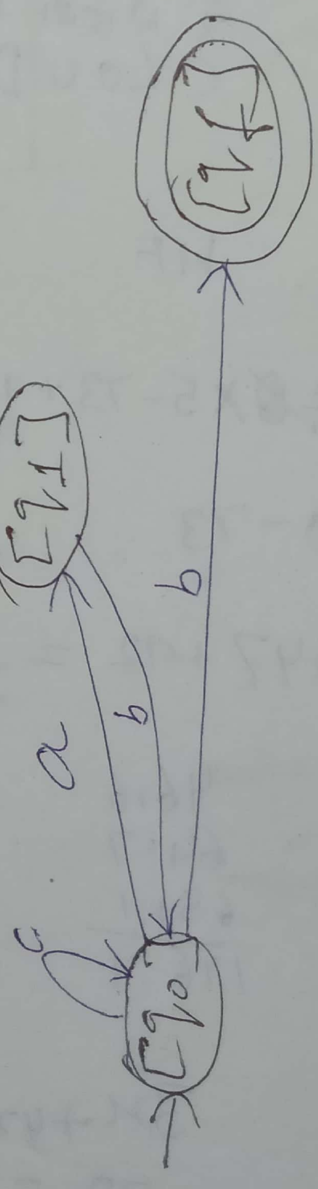
Ans.

Transition Table for NFA

State	Inputs a	b	c
$\rightarrow q_0$	q_1	q_f	q_0
q_f	—	—	—
q_1	—	q_0	—

Successor Table for DFA

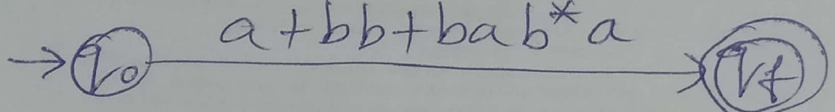
State	Inputs a	b	c
$\rightarrow [q_0]$	$[q_1]$	$[q_f]$	$[q_0]$
$[q_1]$	ϕ	$[q_0]$	ϕ
$[q_f]$	ϕ	ϕ	ϕ

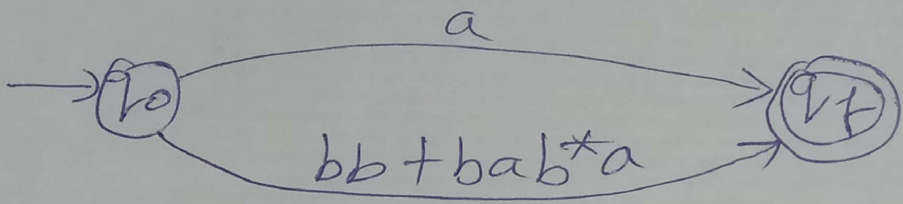


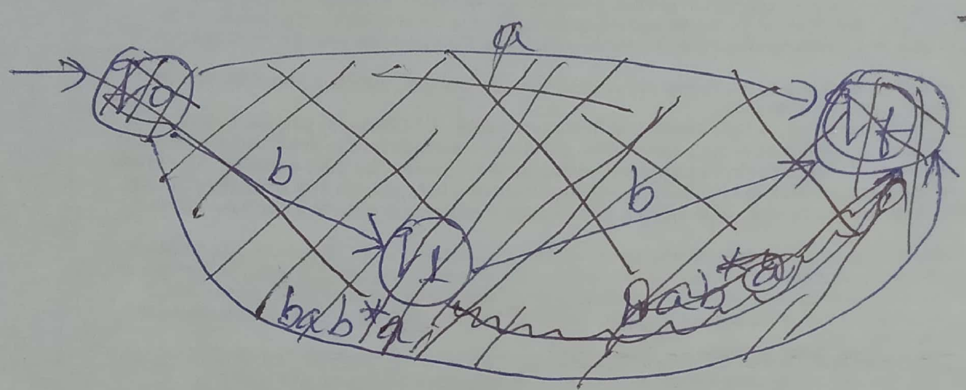
Q-(9.)
KLP (108) Construct a transition system cor-responding to the Regular Expression

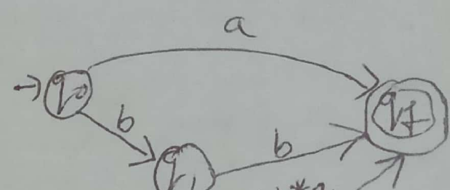
(ii) $a + bb + bab^*a$

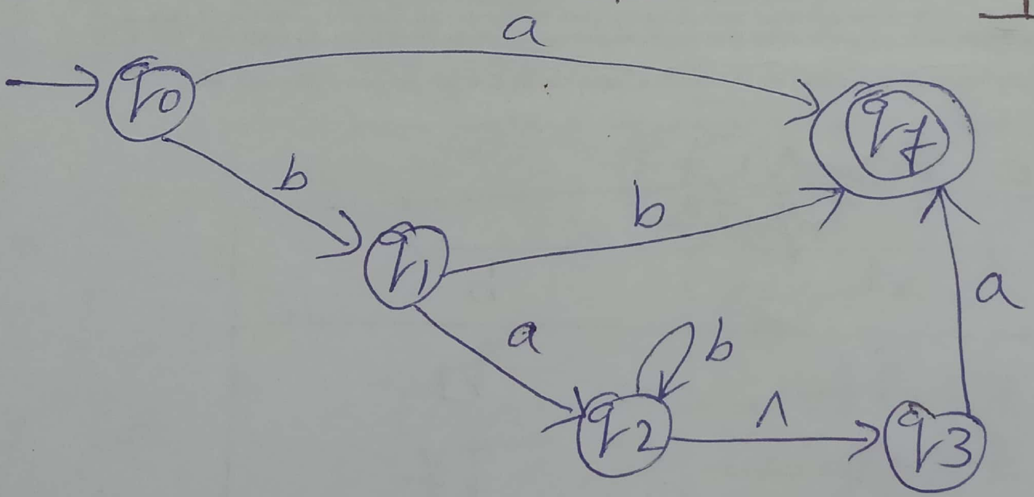
Sol.ⁿ-

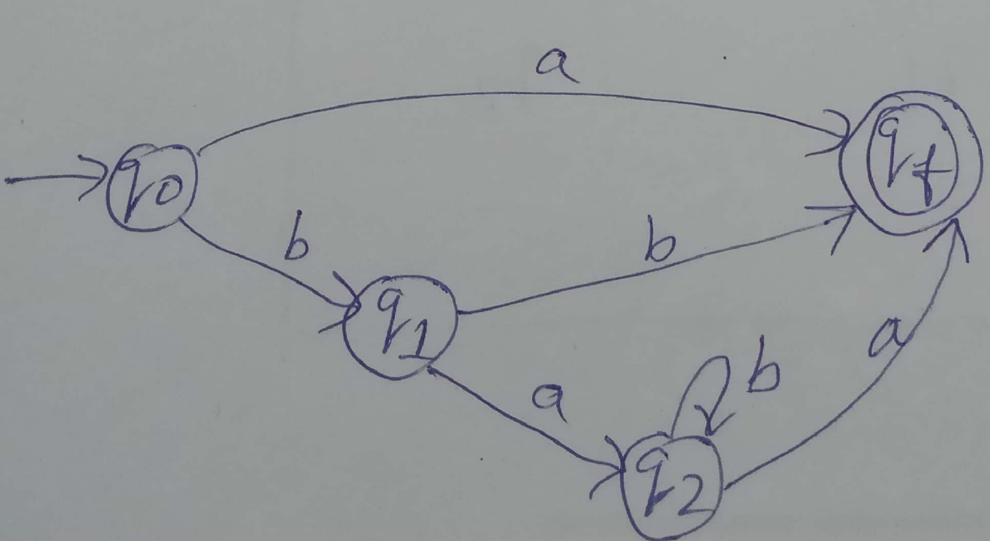












Successor Table for DFA

State	Inputs a	Inputs b
$\rightarrow[q_0]$	$[q_f]$	$[q_1]$
$[q_f]$	ϕ	ϕ
$[q_1]$	$[q_2]$	$[q_f]$
$[q_2]$	$[q_f]$	$[q_2]$

Transition Graph

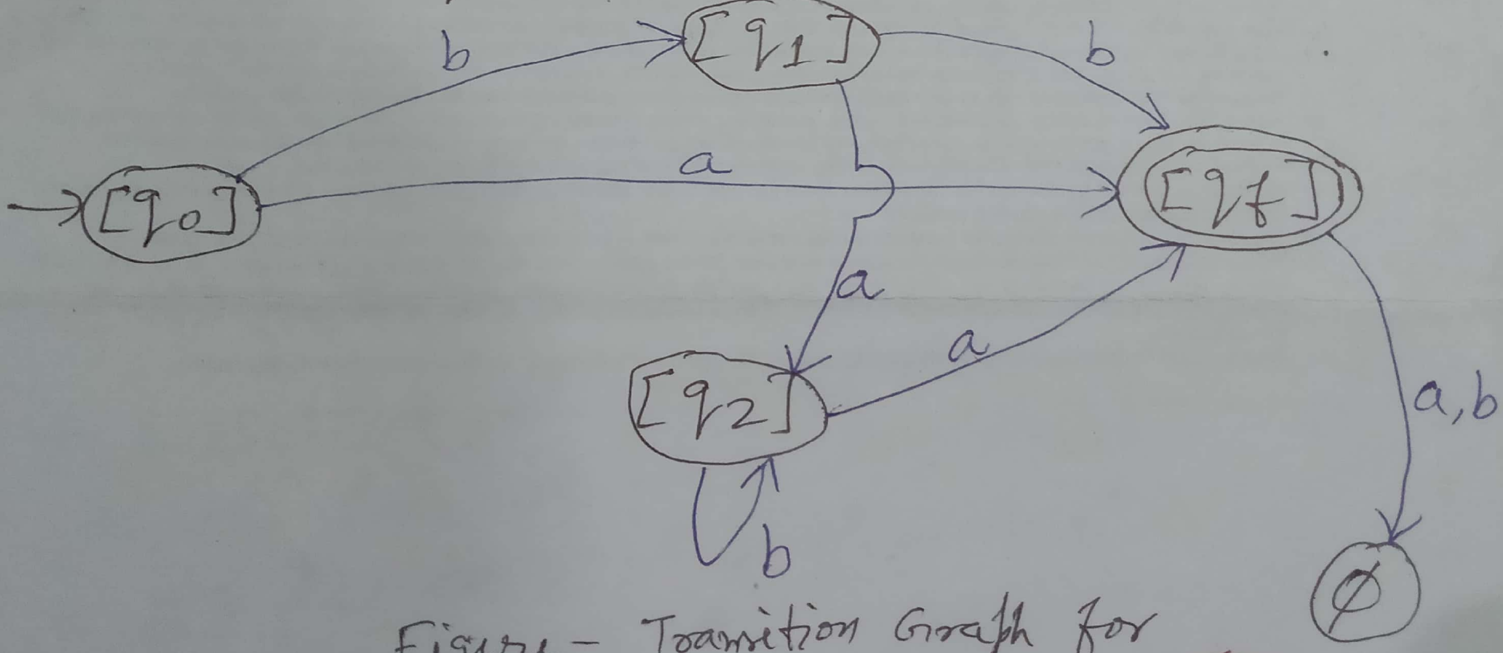


Figure – Transition Graph for Equivalent R.E. ✓ Ans.

CHAPTER-4

Q-(1.) Represent following sets by regular expressions:

(a.) $\{0, 1, 2\}$

$= 0 + 1 + 2$ ✓ Ans.

(b.) $\{1^{2n+1} \mid n > 0\}$

Let $n = 1$, $1^{2\times1+1} \Rightarrow 1^{2+1} \Rightarrow 1^3$

Let $n = 2$, $1^{2\times2+1} \Rightarrow 1^{4+1} \Rightarrow 1^5$

Let $n = 3$, $1^{2\times3+1} \Rightarrow 1^{6+1} \Rightarrow 1^7$

$= \boxed{1(11)^*}$ ✗ (wrong) ~~Ans.~~ $\boxed{111(11)^*}$ ✓ Ans.

(c.) $\{w \in \{a, b\}^* \mid w \text{ has only one } a\}$.

$= b^*ab^*$ ✓ Ans.

(d.) The set of all strings over $\{0, 1\}$ which has at most two zeros.

$= 01^* + 1^* + 001^*$

$= (0 + \Lambda + 00)1^*$ ✓ Ans.

(e.) $\{a^2, a^5, a^8, \ldots\}$

$= aa(aaa)^*$ ✓ Ans.

(f.) $\{a^n \mid n \text{ is divisible by 2 or 3 or } n = 5\}$.

$= (aa)^* + (aaa)^* + aaaaa$ ✓ Ans.

(g.) The set of all strings over $\{a, b\}$ begining & ending with a.

$= a(a+b)^*a$ ✓ Ans.

Pumping (Generating) Lemma ①

Ex-(1) Use P.L to prove that Language is not CFL

Solⁿ:- $L = \{a^n b^n \mid n \geq 1\}$, $n = 1, 2, \ldots$

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

$a^n b^n$	$n=1$
ab	
aabb	$n=2$
aaabbb	$n=3$

Let $w = aabb$

$w = xyz$

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

Case 1:-

$|xy| \leq n$

$3 \leq 4$ ✓ (True)

Case 2:-

$|y| > 0$

$2 > 0$ ✓ (True)

Case 3:-

$w = xy^iz$ $i=2$

xy^2z

$= a(ab)^2b$

$= aababb \notin L$

Correct Ans.

$i=1$ (we will not consider it being we are checking for -ve test)

xy^1z

$= \cancel{aabb}$

$aabb \in L$

(2)

Ex-(2.) Using Pumping Lemma to prove that language is not CFL.

$L = \{0^n 1^n 2^n \mid n \geq 1\}$

$0^n 1^n 2^n$	
0 1 2	$n=1$
001122	$n=2$
000111222	$n=3$

Solⁿ:- $L = \{012, 001122, 000111222, \ldots\}$

Let, $w = 001122$

$w = xyz$

$w = \underset{x}{0}\,\underset{y}{0112}\,\underset{z}{2}$

Case 1:- $|xy| \leq n$

$5 \leq 6$ ✓ (True.)

Case 2:- $|y| > 0$

$4 > 0$ ✓ (True.)

Case 3:- $w = xy^iz$

Let $i = 1$, $w = xy^1z$

$= 001122 \in L$ (We will not consider it.)

Correct.

Let $i = 2$, $w = xy^2z$

$= 0(0112)^2 2$

$= 0011201122 \notin L$

✓ Ans.

Ex.-(4.18)
KLP(101) Show that the set $L = \{a^{i^2} \mid i \geq 1\}$ is not regular.

Sol^n:- $L = \{a, aaaa, aaaaaaaaa, ----\}$

$i = 1$
$a^{1^2} \Rightarrow a^1 \Rightarrow a$
$i = 2$
$a^{2^2} \Rightarrow a^4 \Rightarrow aaaa$
$i = 3$
$a^{3^2} \Rightarrow a^9 \Rightarrow aaaaaaaa$
$i = 4$
$a^{4^2} \Rightarrow a^{16} \Rightarrow aaaaaaaa$ $aaaaaaaa$

Let $w = aaaa$

$w = xyz$

$w = \underset{x}{a}\ \underset{y}{aa}\ \underset{z}{a}$

Case 1:- $|xy| \leq n$

$3 \leq 4$ holds True.

Case 2:- $|y| > 0$

$2 > 0$ holds True.

Case 3:- $w = xy^iz$

let $i = 1$

$= xy^1z$

$= a(aa)^1a$

$= aaaa \in L$

(We will not consider it because we are checking for negativity test.)

let $i = 2$

$= xy^2z$

$= a(aa)^2a$

$= aaaaaa \notin L$

Ans.

Ex.-(4.19)
KLP(102) Show that $L = \{a^p \mid p \text{ is a prime}\}$ not regular.

Sol^n:-
$L = \{a, aa, aaa, aaaaa, ---\}$

$p = 1$
$a^1 \Rightarrow a$
$p = 2$

Let $w = aaaaa$

$w = xyz$

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

$a^2 \Rightarrow aa$
$p = 3$ $a^3 \Rightarrow aaa$
$p = 5$ $a^5 \Rightarrow aaaaa$

Case 1:- $|xy| \le n$

$4 \le 5$ holds True.

Case 2:- $|y| > 0$

$3 > 0$ holds True.

Case 3:- $w = xy^i z$

let $i = 1$

$= xy^1 z$

$= a(aaa)^1 a$

$= aaaaa \in L$

(We will not consider it because we are checking for negativity test.)

let $i = 2$

$= xy^2 z$

$= a(aaa)^2 a$

$= aaaaaaa \notin L$

Ans.

Ex.-(4.20) KLP(102) Show that $L = \{0^i 1^i \mid i \ge 1\}$ is not regular

Sol^n:- $L = \{01, 0011, 000111, \ldots\}$

$i = 1$ $0^1 1^1 \Rightarrow 01$
$i = 2$ $0^2 1^2 \Rightarrow 0011$
$i = 3$ $0^3 1^3 \Rightarrow 000111$

Let $w = xyz$

$w = \underset{x}{0}\underbrace{0011}_{y}\underset{z}{1}$

Case 1:- $|xy| \le n$

$5 \le 6$ holds True.

Case 2:- $|y| > 0$

$4 > 0$ holds True.

Case 3:- $w = xy^i z$

let $i = 1$

$= xy^1 z$

$= 0(0011)^1 1$

$= 000111 \in L$

let $i = 2$

$= xy^2 z$

$= 0(0011)^2 1$

$= 0001100111 \notin L$

Ans.

(we'll not consider, as checking for -ve test)

Ex (4.22) KLP-(103) Is $L = \{a^{2n} \mid n \geq 1\}$ regular?

Sol^n:- $L = \{aa, aaaa, aaaaaa, \ldots\}$

$n = 1$ $a^{2\times 1} \Rightarrow a^2 \Rightarrow aa$
$n = 2$ $a^{2\times 2} \Rightarrow a^4 \Rightarrow aaaa$
$n = 3$ $a^{2\times 3} \Rightarrow a^6 \Rightarrow aaaaaa$
$n = 4$ $a^{2\times 4} \Rightarrow a^8 \Rightarrow aaaaaaaa$

let $w = xyz$

$w = \underset{x}{a}\,\underbrace{aaaa}_{y}\,\underset{z}{a}$

Case 1:- $|xy| \leq n$

$5 \leq 6$ holds True.

Case 2:- $|y| > 0$

$4 > 0$ holds True.

Case 3:- $w = xy^iz$

let $i = 1$

$= xy^1z$

$= a(aaaa)^1a$

$= aaaaaa \in L$

~~(We'll not consider it because we are checking for -ve test)~~

let $i = 2$

$\Rightarrow xy^2z$

$= a(aaaa)^2a$

$= a\,\underline{aaaa}\,\underline{aaaa}\,a \in L$

~~(we will not consider it because we are checking for -ve test)~~

let $i = 3$

$= xy^3z$

$= a(aaaa)^3a$

$= a\,\underline{aaaa}\,\underline{aaaa}\,\underline{aaaa}\,a \in L$ ✓ Ans.

So, it is proved that the Language L is regular

Q-(17.) KLP(110) Using Pumping Lemma show ... sets are not regular:

(a.) $\{a^n b^{2n} \mid n > 0\}$

Sol^n - $L = \{abb, aabbbb, aaabbbbbb, \ldots\}$

$n=1$ $a^1 b^{2\times1} \Rightarrow a^1 \cdot b^2 \Rightarrow abb$
$n=2$ $a^2 b^{2\times2} \Rightarrow a^2 b^4 \Rightarrow aabbbb$
$n=3$ $a^3 b^{2\times3} \Rightarrow a^3 b^6 \Rightarrow aaabbbbbb$
$n=4$ $a^4 b^{2\times4} \Rightarrow a^4 b^8$ $\Rightarrow aaaabbbbbbbb$

Let $w = xyz$

$w = \underset{x}{a}\underbrace{abbb}_{y}\underset{z}{b}$

Case 1:- $|xy| \leq n$

$5 \leq 6$ holds True.

Case 2:- $|y| > 0$

$4 > 0$ holds True.

Case 3:- $w = xy^iz$

Let $i = 1$

$= xy^1z$

$= a(abbb)^1 b$

$= aabbbb \in L$

(we'll not consider it, as we are checking for -ve test)

Let $i = 2$

$= xy^2z$

$= a(abbb)^2 b$

$= a\underline{abbb}\,\underline{abbb}\,b \notin L$ Ans.

~~Q-(17.) KLP(110)~~ (b.) $\{a^n b^m \mid 0 < n < m\}$

Sol^n - $L = \{abb, aabbb, aaabbbb, \ldots\}$

$n > 0, m > n$ $n=1, m=2$ $a^1 b^2 \Rightarrow abb$
$n=2, m=3$ $a^2 b^3 \Rightarrow aabbb$
$n=3, m=4$ $a^3 b^4 \Rightarrow aaabbbb$
$n=4, m=5$ $a^4 b^5 \Rightarrow aaaabbbbb$

Let $w = xyz$

$w = \underset{x}{a}\underbrace{abb}_{y}\underset{z}{b}$

Case 1:- $|xy| \leq n$

$4 \leq 5$ holds True.

Case 2:- $|y| > 0$

$3 > 0$ holds True.

------ continued

continued

Q-(17.)
KLP(110) (b.)

Case 3:- $w = xy^i z$

Let $i = 1$
$= xy^1 z$
$= a(abb)^1 b$
$= aabbb \in L$

(we'll not consider it, because we are checking for -ve test.)

Let $i = 2$
$= xy^2 z$
$= a(abb)^2 b$
$= a\underline{abb}\,\underline{abb}\,b \notin L$ ✓ Ans

Ex-(4.21)
KLP(102) Show that $L = \{ww \mid w \in \{a,b\}^*\}$ is not regular.

Solⁿ- Let $w = a^n b$
so, $ww = a^n b a^n b$
$L = \{abab, aabaab, aaabaaab, ----\}$

$n = 1$
Let $w = a^n b$
$\therefore ww = a^n b a^n b$
$= a^1 b a^1 b \Rightarrow abab$

$n = 2$
$= a^2 b a^2 b \Rightarrow aabaab$

$n = 3$
$= a^3 b a^3 b \Rightarrow aaabaaab$

Let $w = xyz$
$w = \underbrace{a}_{x}\underbrace{abaa}_{y}\underbrace{b}_{z}$

Case 1:- $|xy| \le n$
$5 \le 6$ holds True.

Case 2:- $|y| > 0$
$4 > 0$ holds True.

Case 3:- $w = xy^i z$

Let $i = 1$
$= xy^1 z$
$= a(abaa)^1 b$
$= aabaab \in L$

(but, we'll not consider, as checking for -ve test)

Let $i = 2$
$= xy^2 z$
$= a(abaa)^2 b$
$= a\underline{abaa}\,\underline{abaa}\,b \notin L$

So, it is proved that Language is not regular.
✓ Ans.

* **Mealy Machine** - A mealy machine is defined as a machine in automata whose output values are determined by both its current state & current inputs. In this machine, at most one transition is possible.

It has 6 tuples $(Q, q_0, \Sigma, \Delta, \delta, \lambda^*)$
where,

Q is a finite set of states.

Σ is the input alphabet.

q_0 is the initial state.

Δ is the output alphabet.

δ is the transition function which maps
$Q \times \Sigma \rightarrow Q$.

λ^* is the output function which maps
$Q \times \Sigma \rightarrow \Delta$.

* **Moore Machine** - A moore machine is defined as a machine in automata whose output values are determined only by its current state.

It also has 6 tuples $(Q, q_0, \Sigma, \Delta, \delta, \lambda)$

Q is a finite set of states.

q_0 is the initial state.

Σ is the input alphabet.

Δ is the output alphabet.

δ is the transition function which maps
$Q \times \Sigma \rightarrow Q$.

λ is the output function which maps $Q \to \Delta$.

Some Points about Moore Machine:-

(1.) Output depends only upon present state.
(2.) If input changes, output does not change.
(3.) More no. of states are required.
(4.) There is more hardware requirement.
(5.) They react slower to inputs.
(6.) Synchronous output and state generatio
(7.) Output is placed on states.
(8.) Easy to design.

me Points about Mealy Machine:-

(1.) Output depends on present state as we
as present input.

(2.) If input changes, output also changes. (3)
(3.) Less no of states are required.
(4.) There is less hardware requirement.
(5.) They react faster to inputs.
(6.) Asynchoronous output generation.
(7.) output is placed on transitions.
(8.) It is difficult to design.

* Difference between Mealy & Moore Machine

Mealy Machine	Moore Machine
(1.) O/P depends both upon present state & present input.	(1.) O/P depends only upon present state.
(2.) Generally, it has fewer states than Moore Machine	(2.) Generally, it has more states than Mealy Machine.
(3.) Value of O/P fun. is a fun. of transitions & changes, when i/p logic on present state is done.	(3.) Value of O/P fun. is a fun. of current state & changes at clock edges whenever state changes occur.
(4.) Mealy Machines react faster to i/ps. They generally react in same clock cycle.	(4.) In Moore Machines, more logic is required to decode the o/ps resulting in more circuit delays. They generally react one clock cycle.

Algorithm for Moore Machine to Mealy Machine

Input- Moore Machine
output- Mealy machine
step 1- Take a blank Mealy Machine transi-tion table format.

step 2- copy all the Moore Machine transi-tion states into this table format.

step 3- Check the present states and their corresponding outputs in the Moore Machine state table; if for a state Q_i output is m, copy it into the output columns of the Mealy Machine state table wherever Q_i appears in the next state.

Algorithm for Mealy Machine to Moore Machine

Input - Mealy machine
output- Moore Machine
step 1- Calculate the no. of different outputs for each state (Q_i) that are avail-able in the state table of the

Mealy machine.

Step 2- If all the outputs of Q_i are same, copy state Q_i. If it has n distinct outputs, break Q_i into n states as Q_{in} where $n = 0, 1, 2, \ldots$

Step 3- If the output of the initial state is 1, insert a new initial state at the begining which gives 0 output.

② Moore to Mealy Machine Conversion

Step 1- Write down the association of O/P with respect to Present State.

Step 2- Construct Mealy Machine Table with copying the Present states.

Step 3- Write the Next States in Mealy Machine as it is from Given Table.

Step 4- Write the associated O/P values (calculated in Step 1) in O/P column.

① Mealy to Moore Machine Conversion

Step 1- Split the states under input column into different states, the no. of such state being equal to no. of different outputs associated with these states.

E Q-(2.9) KLP(39) Consider the Mealy Machine given in transition table. Construct a Moore Machine which is equivalent to the Mealy machine.

Transition Table of Mealy Machine

Present State	Next State			
	input a=0		input a=1	
	state	output	state	output
$\rightarrow q_1$	q_3	0	q_2	0
q_2	q_1	1	q_4	0
q_3	q_2	1	q_1	1
q_4	q_4	1	q_3	0

Step 1- q_3 is associated with output 0

q_1 " " " 1

q_2 " " " 0 & 1

q_4 " " " 0 & 1

Step 2- Break q_2 into q_{20} & q_{21}

" q_4 " q_{40} & q_{41}

Step 3- Transition Table of Moore Machine

Present state	Next State		output
	a=0	a=1	
$\rightarrow q_0$	q_3	q_{20}	0
q_1	q_3	q_{20}	1
q_{20}	q_1	q_{40}	0
q_{21}	q_1	q_{40}	1
q_3	q_{21}	q_1	0
q_{40}	q_{41}	q_3	0
q_{41}	q_{41}	q_3	1

Ex-(2.9) ①
KLP(3.9) Consider the Mealy Machine given in Transition Table. Construct a Moore Machine which is equivalent to the Mealy Machine.

Transition Table of Mealy Machine (Given)

Present State	Next State			
	input a=0		input a=1	
	state	output	state	output
$\rightarrow q_1$	q_3	0	q_2	0
q_2	q_1	1	q_4	0
q_3	q_2	1	q_1	1
q_4	q_4	1	q_3	0

Transition Table of Moore Machine

I Present State	Next State		output
	a=0	a=1	
$\rightarrow q_1$	q_3	q_{20}	1
q_{20}	q_1	q_{40}	0
q_{21}	q_1	q_{40}	1
q_3	q_{21}	q_1	0
q_{40}	q_{41}	q_3	0
q_{41}	q_{41}	q_3	1

उदा.

Ex:-(2.10)
KLP-(41) Construct a Mealy Machine which is equi-valent to the Moore Machine given in Table.

Transition Table of Moore Machine (Given)

Present State	Next State a=0	Next State a=1	output
$\rightarrow q_0$	q_3	q_1	0
q_1	q_1	q_2	1
q_2	q_2	q_3	0
q_3	q_3	q_0	0

q_0 is associated with output 0
q_1 " " " " 1
q_2 " " " " 0
q_3 " " " " 0

Transition Table of Mealy Machine

Present State	Next state			
	a=0		a=1	
	state	output	state	output
$\to q_0$	q_3	0	q_1	1
q_1	q_1	1	q_2	0
q_2	q_2	0	q_3	0
q_3	q_3	0	q_0	0

उR.

Ex:-(2.11) KLP-(42) Consider the Moore Machine described by the transition table. Construct the corresponding Mealy Machine.

Transition Table of Moore Machine (Given)

Present State	Next State		output
	a=0	a=1	
$\to q_1$	q_1	q_2	0
q_2	q_1	q_3	0
q_3	q_1	q_3	1

q_1 is associated with output 0

q_2 " " " " 0

q_3 " " " " 1

Transition Table of Mealy Machine

Present state	Next state			
	a=0		a=1	
	state	output	state	output
$\to q_1$	q_1	0	q_2	0
q_2	q_1	0	q_3	1
q_3	q_1	0	q_3	1

उR.

Ex.-(2.12) KLP-(43) Consider a Mealy Machine given in Table. Construct a Moore Machine equivalent to this Mealy Machine.

Transition Table of Mealy Machine (Given)

Present State	Next State			
	$a=0$		$a=1$	
	state	output	state	output
$\rightarrow q_1$	q_2	Z_1	q_3	Z_1
q_2	q_2	Z_2	q_3	Z_1
q_3	q_2	Z_1	q_3	Z_2

Transition Table of Moore Machine

Present State	Next State		output
	$a=0$	$a=1$	
$\rightarrow q_1$	q_{21}	q_{31}	—
q_{21}	q_{22}	q_{31}	Z_1
q_{22}	q_{22}	q_{31}	Z_2
q_{31}	q_{21}	q_{32}	Z_1
q_{32}	q_{21}	q_{32}	Z_2

उत्र

Q-(11.) KLP-(52) Construct a Mealy Machine which is corresponding to the Moore Machine giv in Table.

Transition Table of Moore Machine (Given)

Present State	Next State		output
	$a=0$	$a=1$	
$\rightarrow q_0$	q_1	q_2	1
q_1	q_3	q_2	0
q_2	q_2	q_1	1
q_3	q_0	q_3	1

Q-(11.)
KLP(52) Construct a Mealy Machine which is equivalent to the Moore Machine given in Table below.

Transition Table of Moore Machine

Present State	Next state a=0	Next state a=1	output
$\rightarrow q_0$	q_1	q_2	1
q_1	q_3	q_2	0
q_2	q_2	q_1	1
q_3	q_0	q_3	1

Step 1 + Step 2

Present State	Next state a=0 state	output	a=1 state	output
$\rightarrow q_0$	q_1	0	q_2	1
q_1	q_3	1	q_2	1
q_2	q_2	1	q_1	0
q_3	q_0	1	q_3	1

उत्तर.

Mealy Machine Table

Present state	Next State a = 0 state	output	a = 1 state	output
→ q_0	q_1	0	q_2	1
q_1	q_3	1	q_2	1
q_2	q_2	1	q_1	0
q_3	q_0	1	q_3	1

q_0 is associated with output 1 — उत्तर.
q_1 " " " " 0
q_2 " " " " 1
q_3 " " " " 1

Q-(12.) / KLP-(52.) Construct a Moore machine equivalent to the Mealy machine given in Table.

Mealy machine Table

Present state	Next state a = 0 state	output	a = 1 state	output
→ q_1	q_1	1	q_2	0
q_2	q_4	1	q_4	1
q_3	q_2	1	q_3	1
q_4	q_3	0	q_1	1

q_1 is associated with output 1
q_{20} " " " " 0
q_{21} " " " " 1
q_{30} " " " " 0
q_{31} " " " " 1
q_4 " " " " 1

Moore Machine Table

Present State	Next State $a=0$	Next State $a=1$	output
$\rightarrow q_1$	q_1	q_{20}	1
q_{20}	q_4	q_4	0
q_{21}	q_4	q_4	1
q_{30}	q_{21}	q_{31}	0
q_{31}	q_{21}	q_{31}	1
q_4	q_{30}	q_1	1