19/8/16

CS-502.

(Automata Theory) (Lecture # 01)

Marks Distribution:-

Final = 70
Quiz = 10
Mid term = 20

Book:-
Computer theory By Daniel I.A Cohen.

Course Topics:-

1. introduction
2. Language
3. Recursive Diffination.
4. Regular Expression.
5. Finite Automata Theory
6. Transition Graph
7. Kleens theorem.
8. FA with O/P
9. CFG
10. Push down Automata.

* INTRODUCTION:-

*- How to Understand Computer theory?

Computer Theory Course	Practical Engine Course
*- We deal with those machines that do not Exist currently that can ever be demand off	*- We deal with those machines that currently Exist.
*- Concern of probability.	*- Concept of optimality.

*- Compiler Construction:-

Lexical Analizer
↓
Syntax Analyzer
↓
Semantic Analyzer
↓
code Generator
↳ code optimization.....

*- Lexical Analyzer:-

TO Generate a Token.

E.g

int a ;

Token set:-

int → keyword

a → identifier

; → semicolum.

*- Syntax Errors:- / Syntax Analyzer:-

letter (letter / digit)

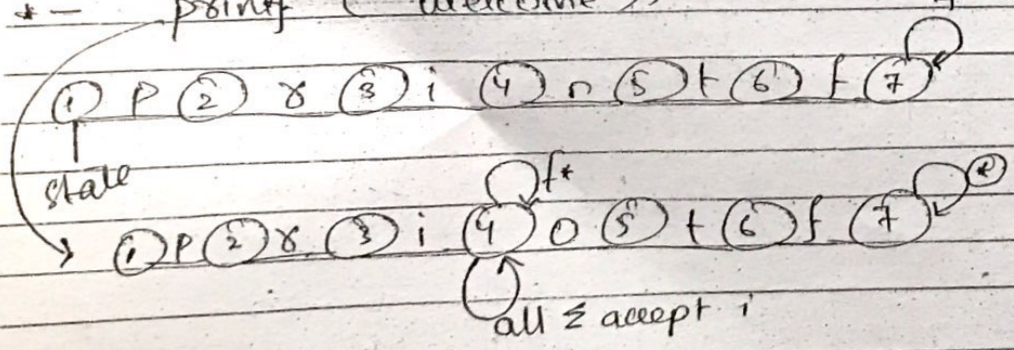
E.g:-

int a = 4.8;

*- Semantic Errors:-

Type checking errors

*- printf ("welcome");



Chapter 2:- LANGUAGE:

*- **Alphabets** (Σ):

A finite set of Symbols.

*- **Word / String**:

A finite sequence of alphabetic symbols.

*- **Language**:-

Set of Strings over some input alphabets.

Examples:-

sign of input alphabet Σ (a, b) than

separator

a , b, aa, ab, bb, abbb.....

OR

L = {a b aa ab bb abbb.....}

infinite

*) language strats from a capital letter.

*) In carli braces the seprator (,) are not use.

Example:-

From a language over $\Sigma = (a,b)$ that have a length 2.

$L = \{aa \ ab \ ba \ bb\} \rightarrow$ finite

*- Function:-

① Length:-

Length $(x) = 1$
Length $(xx) = 2$
Length $(aba) = 3$
Length $((xx)(xxx)) = 9$
Length $(\Lambda) = 0$

② Null String:-

Nullstring (ϵ or Λ).

*-③ Recursive:- / Reverse:-

reverse (ab) = ba
reverse (10) = 01
reverse (aa) = aa
↓
palindrom.

reverse (101) = 101
reverse (aba) = aba } palindrom

Q: Form a language of palindrom of length 3. $\Sigma = (0,1)$

$\Lambda = \{000 \ 111 \ 101 \ 010\}$.

26/8/16 :-

Lecture (2)

STRING CONCATENATION :-

The Concatenation of two Strings is formed by writing all characters of one string to the End of another.

Example:-

if x = data , y = base

xy = database

yx = basedata

$\therefore xy \neq yx$.

*) Aditive Identity :-

= 0 (Null value)

*) Multiplicative Identity :-

= (1)

*) Identity Element over concatenation:

$\Lambda S = S \Lambda = S$.

TERMS FOR PART OF A STRING:-

*-) **PREFIX OF S :-**

It is obtained by removing zero or more trailing symbols from a string.

∴ trailing → (at the end)

Examples:-

*- man is the prefix of mankind

*- data is the prefix of database.

*-) **SUFFIX OF S :-**

It is obtained by deleting zero or more leading symbols from a string.

Examples:-

*- kind is suffix of mankind

*- base is the suffix of database

***-) SUBSTRING OF S:-**

It is obtained by deleting a prefix and / or a suffix from string S.

Examples:-

*- nki is the substring of mankind.

*- d is the suffix of mankind.

***) SUB SEQUENCE OF S:-**

Any string formed by deleting zero or more not necessarily contiguous symbol from S.

Examples:-

*- maid is the subsequence of mainkind.

* mkd is the subsequence of mainkind.

*) Contiguous word is substring.

*). No Contiguous word is subsequence

OPERATION ON LANGUAGE:-

Suppose L and M are two languages:

***1)- UNION OF L and M : LUM**

$LUM = \{S \mid S \text{ is in } L \text{ or } S \text{ is in } M\}$

***2)- CONCATENATION OF L and M:**

$LM = \{st \mid s \in L \text{ and } t \in M\}$
$ML = \{st \mid s \in M \text{ and } t \in L\}$

*- order important.

Example:-

$xy \neq yx$.

***3)- KLEEN · CLOUSURE/STAR OF L:**

$L^* = \{\bigcup_{i=0}^{\infty} L^i\}$ (length 0- infinite)
null accept.

***4) - POSITIVE CLOUSURE OF L:**

$L^+ = \{\bigcup_{i=1}^{\infty} L^i\}$ (length 1- infinite)
not null (λ)

aaa aab aba abb bbb bab baa bba

*) L^* – Recursively Repeated its self infinite with accept null (Λ)

*) L^+ – Recursively Repeated its self infine with not accept null (Λ)

*) $L^* - L^+ = \Lambda$

QUESTIONS :-

Q) $L = \{A\ B\ C\ ----\ Z\}$
$M = \{0\ 1\ 2\ ----\ 9\}$

(1) ~~$L \cup M$ = {set of letter and digits}~~

(2) LM = {set of strings start from a letter and followed by a digits}.

Example:-
ABC0123

(3) ML = {set of strings start from a digits and followed by a letter}.

(3) $L^* = \{\Lambda\ A\ B\ C\ ----\ Z\ AA\ AB\ AC\ ---$ | set of all possible letters including $\Lambda\}$.

(4) M^+ = {set of all possible strings of digits excluding Λ}

(5) $L(L \cup M)^*$ = {set of all possible strings of digits excluding letter and also excluding Λ begining with a letter}.

Q: Consider the language S^* where $S = \{a, b\}$: How many words does this language have of length 2? of length 3?

*– $S^* = \{\Lambda\ a\ b\ ab\ abb\ abab\ abbaba ...\}$

*– $S^2 = \{aa\ ab\ ba\ bb\}$.

*– $S^3 = \{aaa\ aab\ aba\ baa\ bbb\ bba\ abb\ bab\}$.

7

Q: Consider the language S^* where $S = \{aa, b\}$.
How many words does the language have of length 4? of length 5? if length 6?

*- $S^4 = \{$aaaa aabb bbaa bbbb baab$\}$

*- $S^5 = \{$bbbbb bbbaa baabb aabbb aaaab baaaa aabaa$\}$

*- $S^6 = \{$ aaaaaa aaaabb aabbaa bbaaaa bbbbbb bbbbaa bbaabb aabbbb baaaab baabaa $\}$

Q: Consider the language S^*. where $S = \{ab, ba\}$.
Write out all the words in S^* that have seven or fewer letter. Can any words in this language contain the substring aaa or bbb?

$(aa+b)^*$
$(aa+b)^3 (aa+b)^2 (aa+b) (aa+b) (aa+b)$
$\cancel{aa}($aaaa aab baa bb$)($aaaa aab baa bb$)(aa+b)$
$($aaaa aab baa bb$)($aaaaaa aaaab baaaa bbaa aaaab aabb baab bbb

*- Can not any words in this language contain the substring aaa or bbb because the given input alphabets dont match aaa or bbb.

*- $S^* = \{ \Lambda$ ab ba abab abba baab baba ababab bababa ababba abbaab baabab babaab baabba abbaba $\}$.

Q: Consider the language S^*. where $S = \{a\ ab\ ba\}$ is the string (abbba) a word in this language? Write out all the words in this language with seven or fewer letters.

*- The word (abbba) in this language dont repeat.

*- $S = \{ \Lambda$ a ab ba aab aba abba baa aba baab aabba abaab aa aaa aaaa abaa aaba baaa aaba abab abba baab baba $\}$.

Q: Consider the language S^*. whose $S = \{aa \ aba \ baa\}$ show that the words aabaa, baabaaa and baaaababaaaa are all in this language have an odd total no of a's?

*- there is no odd no of a's in this language because the given alphabets are even no of a's.

* - (aa)(baa) Yes.

* - (baa)(aba)(aa) Yes

* - (baa)(aa)(aba)(baa)(aa) Yes.

2/9/16 :- Chapter (3) lecture (3)

RECURSIVE DEFINATIONS:-

Its a three steps process.

① We declear basic object in the set.

② We construct some rule for constructing more objects of the set with the help of ①

③ Nothing Else belongs to the set.

*) Recursive function means:
Repeat by themselves.

Example:-
Factorial.

① Recursive defination for set EVEN.
solution

① 2 is in Set EVEN.
② If x is in EVEN, than $x+2$ is also EVEN.
③ only those are in EVEN. Which helped to ① and ②.

9

prove that (12) is an Even number.

1. 2 is in Even. By Rule 1
2. 2+2=4 is in Even By Rule 2
3. 4+2=6 is in Even By Rule 2
4. 6+2=8 is in Even By Rule 2
5. 8+2=10 is in Even By Rule 2
6. 10+2=12 is in Even By Rule 2

*) Any NO is to be given the steps is followed by $(n \div 2)$ steps.

Example:-

NO = 12

so,

$\frac{12}{2} = 6$

No of steps = 6.

Better Even Recursive Defi:-

(2) if x and y are in Even then x+y is also a Even.

(2) if x is in Even than 2x is also Even.

Remainder

(2) if $x/2 == 0$, than x is in Even.

(2) Recursive Defination for x^+.

$\{x \quad xx \quad xxx \quad xxxx\ldots\}$

Solution.

1. x is in x^+
2. if Q is in x^+, than xQ is also in x^+.
3. Nothing Else are in x^+.

so,

$xxxx \in x^+$

1. x is in x^+ Q — By Rule 1
2. $x(x) = xx$ is in x^+ — By Rule 2
3. $x(xx) = xxx$ is in x^+ — By Rule 2
4. $x(xxx) = xxxx$ is in x^+ — By Rule 2

xQ

(3) Recursive Defination of x^*.

solution.

1. Λ is in x^*
2. if Q is in x^*, than x^*Q is in x^*
3. Nothing Else are in x^*.

4) Recursive Deffination for Polynomial:-

(1) The variable x is in polynomial and any number is in polynomial.

(2) if P and q are in polynomial then $p+q$, pq, (P).... are also in polynomial.

(3) Nothing Else are in polynomial.

prove : $3x^2 + 8x - 4$

Solution

1. 3 is in polynomial — By Rule 1
2. x is in polynomial — By Rule 1
3. $(3)(x) = 3x$ is in polynomial — By Rule 2
4. $(x)(3x) = 3x^2$ is in polynomial — By Rule 2
5. 8 is in polynomial — By Rule 1
6. $(x)(8) = 8x$ is in polynomial — By Rule 2
7. $(3x^2) + (8x) = 3x^2 + 8x$ is in polynomial — By Rule 2
8. -4 is in polynomial — By Rule 1
9. $3x^2 + 8x + (-4) = 3x^2 + 8x - 4$ is in poly — By Rule 2

(2) proved $5x^2 - 2x + 9$

1. 5 is in polynomial — By Rule 1
2. x is in polynomial — By Rule 1
3. $(5)(x) = 5x$ is in polynomial — By Rule 2
4. $(x)(5x) = 5x^2$ is in polynomial — By Rule 2
5. -2 is in polynomial — By Rule 1
6. $(x)(-2) = -2x$ is in polynomial — By Rule 2
7. $5x^2 + (-2x) = 5x^2 - 2x$ is in polynomial — By Rule 2
8. 9 is in polynomial — By Rule 1
9. $5x^2 - 2x + 9$ is in polynomial — By Rule 2

chapter (4):

REGULAR EXPRESSION:- (RE)

It is a notation used to describe the regular language.

<u>examples:-</u>

1. $\{a \quad aa \quad aaa \quad aaaa \ldots\}$
   $RE = a^+$

2. $\{\Lambda \quad x \quad xx \quad xxx \quad xxxx \ldots\}$
   $RE = x^*$

3. $\{aa \quad aba \quad abba \quad abbba \ldots\}$
   $RE = ab^*a.$

4. $\{ab \quad abab \quad ababab \ldots\}$
   $RE = (ab)^*$

5. $\{a \quad b\}$
   $RE = a+b$
   $(or) + (U)$

Q write the RE for all possible strings of a's and b's including Λ

$RE = (a+b)^*$

Q write the RE for all possible strings of a's and b's excluding Λ

$RE = (a+b)^+.$

Scanned with CamScanner

9/9/16:

(Test)

Q1(a): write the RE for the language $L = \{w \in (a,b)^*, n_a(w) \bmod 3 = 0\}$

(b) write the RE for the following.

(i) The lang of all strings ending with 1 and dont contain 00. $\Sigma = \{0,1\}$.

(ii) The set of all strings in which there at least 2 occurances of b between any two occurances of a. $\Sigma = \{a,b\}$.

(c) Differentiate between practicle Engg causes V/s computer theory cause?

Q2(a):- write the recursive deffination for $T = \{2^n - 2\}$, $T = \{0, 2, 6, 14, 30, \ldots\}$ n is natural number > 0

(b) Differentiate between syntax Analyzer V/s lexical Analyzer.

(c) $\Sigma = \{a,b\}$

$L_1 = \{a \quad ab \quad ba\}$

$L_2 = \{\epsilon \quad ab \quad b\}$

i- write $L_1 \cup L_2$ as a set of string

ii- write $L_1 L_2$ as a set of string

iii- write a RE for L_2^*.

Answer:-

Q1(a): $b^* + (b^* a b^* a b^* a b^*)^*$

(b) (i) - $(1 + 01)^* 1$

(b) (ii) - $[a(bb)^* a (bb)^* a]^*$

(c) → theory.

then say consecutive 'a' so, $b^* + (b^* aaa b^*)^*$.

Q2(a) ① 0 is T

② If x is T, then $2x+2$ is also in T.

③ Nothing else is in T.

(c) (i) or (ii):

{a aab ab abab abb ba baab bab}

(iii) $(\epsilon + ab + b)^*$:

Lecture (4) :

REGULAR EXPRESSION :-

① { Λ a b aa ab bb aaa aab bbb ...} infinite

RE = $a^* b^*$

② { a c ab cb abb cbb abbb cbbb ...}

RE = $(a+c) b^*$

③ { aaa aab aba abb baa bab bba bbb } → Finite

RE = $(a+b)(a+b)(a+b)$

④ write the RE for the language of all words that have at Least two a's . $\Sigma = \{a, b\}$.

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

then Consecutive 'aa'

so,

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

⑤ write the RE for the language of all words that have Exactly three p's in total. $\Sigma \{p, q\}$.

RE = $q^* p q^* p q^* p q^*$

in case Consecutive 'ppp'

then,

RE = $q^* ppp q^*$.

⑥ { b bb bbbb bbbbb .... }

RE = $b + bb + bbbb^+$ OR.

RE = $b + bb + bbbbb^*$

⑦ write RE for the set of strings of 0's and 1's whose tenth symbol from the right End is 1 $\Sigma = \{0, 1\}$.

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

OR

RE = $(0+1)^* 1 (0+1)^9$

14

(8) write the RE for the set of String with even no of a's followed by odd no of b's. i.e for language.

$L = \{ a^{2n} b^{2m+1} ; n \geq 0 ; m \geq 0 \}$

$RE = (aa)^* (bb)^* b$.

(9) write the RE for the language $L = \{ w \in (0,1)^* ; w$ has no pair of consecutive 0's (zeros). $\}$

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

(10) write the RE for the language $L = \{ a^n b^m \mid n \geq 4 , m \leq 3 \}$.

$RE = aaaa^* (\Lambda + b + bb + bbb)$

(11) write the RE for the language $L = \{ a^n b^m ;$ $(n+m)$ is even $\}$

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

(12). $L = \{ a b^n w ; n \geq 3, w \in (a,b)^+ \}$ (15)

$RE = abbbb^*(a+b)^*$ or $abbb^+ (a+b)^+$.

(13) write RE for the language

$L = \{ w : |w| \bmod 3 = 0 , w \in (a,b)^* \}$

$RE = [(a+b)(a+b)(a+b)]^*$ or

$[(a+b)^3]^*$.

(14) write the RE for the language.

$L = \{ w \in (a,b)^* : n_a(w) \bmod 3 = 0 \}$.

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

$* = 1$

$b + bababab$.

$* = 2$

$bb + bbaabbaabbaabb$

(15) The machine accepts all strings over $\Sigma = \{0,1\}$ which end with 0.

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

(16) All strings with any combination of a's and ends with single b.

RE = a^*b

(17) All strings that start with (a) followed by any number of b's including Λ

RE = ab^*

(18) All strings starts with any number of a's followed by any number of b's.

RE = $a^* b^*$.

(19) Any combination of a's and b's but starts with a and end with b.

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

(20) write RE that accept $\Sigma = \{a,b\}$. starts and end with same symbols.

RE = $a(a+b)^*a + b(a+b)^*b$.

(21) starts and end with different symbols.

RE = $a(a+b)^*b + b(a+b)^*a$.

(22) Any combination of a's and b's but a must be followed by b.

RE = $(ab+b)^*$

(23) language of All strings over $\Sigma = \{a, b\}$ where abbb is substring.

$RE = (a+b)^* abbb (a+b)^*$

(24) Length of Every string is odd over $\Sigma = \{a, b\}$.

$RE = (a+b)[(a+b)(a+b)]^*$

(25) Length of Every string is Even over $\Sigma = \{a, b\}$.

$RE = [(a+b)(a+b)]^*$

(26) Second symbol (b) and second last symbol (a) and length then 3.

$RE = (a+b) b (a+b)^* a (a+b)$.

23/9/16

16

Lectures

Write the RE for : $\Sigma = \{0, 1\}$

① The Language of all strings containing at least two 0's.

② The Language of all strings containing at most two 0's.

③ The Language of all strings Ending with 1 and dont Contain 00.

① $(0+1)^* 0 (0+1)^* 0 (0+1)^*$

② $1^* + 1^* 0 1^* + 1^* 0 1^* 0 1^*$

③ $(1+01)^* 1 + (01)^*$.

FINITE AUTOMATA:-

Finite (No of states are finite)
Automata (Automaton - Automatic)

① FA is a deterministic machine which have finite no of states and their next state will be determined by the last.

② FA is a mathmatical model with discrete inputs and outputs.

③ FA is a generalized transition diagram through which a RE can be compiled.

④ FA Consists of three things.
① Finite no of states. One is marked as initial and at least one is marked as final.
② Transition depends upon Σ.
③ Transition must be unique.

NOTE:-
(*) Initial state (mark) aik hi ho ga. OR Final state (mark) aik say zeyada ho saktay hain.

Language Define by an FA:-

Set of Rules:-

① On state 'X' if I/P 'a' occurs it goes to state 'Y'.

② On state 'X' if I/P 'b' occurs it goes to state 'Z'.

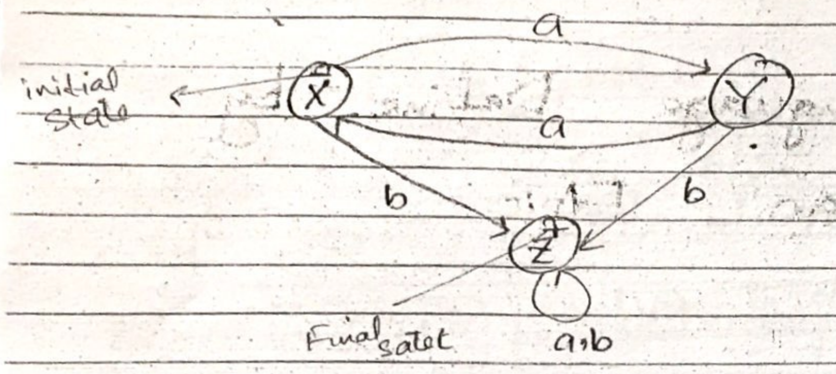
③ On state 'Y' if I/P 'a' occurs it goes to state 'X'

④ On state 'Y' if I/P 'b' occurs it goes to state 'Z'.

⑤ On state 'Z' any input occurs it remain on same state

X – initial , Z → final.

GRAPH :-



Initial state	Final state
$\vec{O}$	◎
start O	final O
⊖	⊕

TRASITION TABLE :-

	a	b
X^-	Y	Z
Y	X	Z
Z^+	Z	Z

states → rows, inputs → columns

Formal Deffination of an FA :-

FA is a set of five tuples.

$(Q, \Sigma, \delta, q^0, F)$

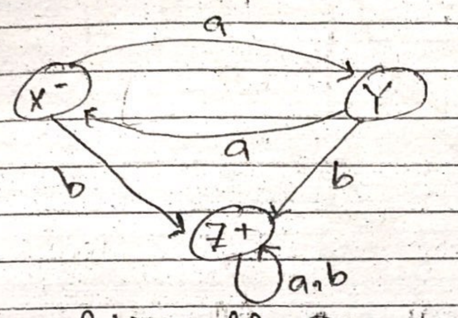
Q → set of finite states
Σ → set of I/P alphabets
q^0 → initial state
F → Set of final state
δ → transition maping function.

19

i.e
$\delta(X, a) = Y$ ($X \xrightarrow{a} Y$)

state, input, new state

Acording to the Graph:-



Formal Definition:

$Q = \{X, Y, Z\}$
$\Sigma = \{a, b\}$
$q_0 = X$
$F = \{Z\}$
$\delta(X, a) = Y$
$\delta(X, b) = Z$
$\delta(Y, a) = X$
$\delta(Y, b) = Z$
$\delta(Z, a) = Z$
$\delta(Z, b) = Z$

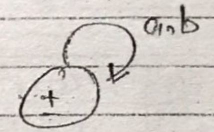
14/10/16.

Lecture (6)

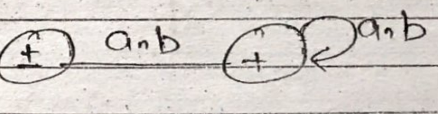
Q: Write the RE and draw an FA for the language of all things of a's and b's including Λ.

RE = $(a+b)^*$

FA :-

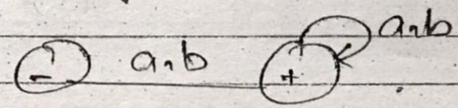


OR



Same above question is asked Excluding Λ.

FA →

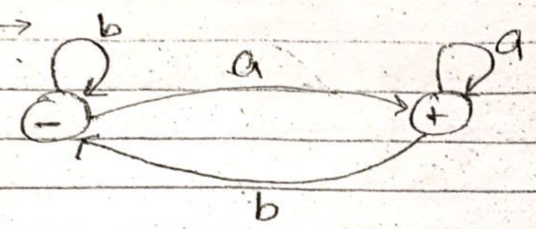


Q2: Write the RE and draw an FA for the language of all strings of a's and b's that must end on 'a'. $\Sigma = \{a, b\}$.

RE = $(a+b)^*a$.

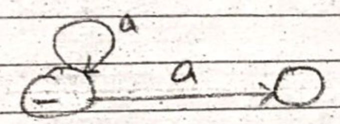
20

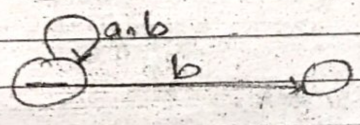
FA →



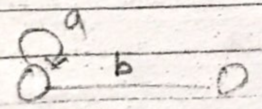
NOTE :-

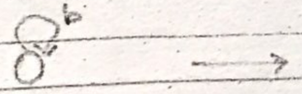
*- aik state say TWO same name consume nahi ho saktay.
i.e

1)  (X)

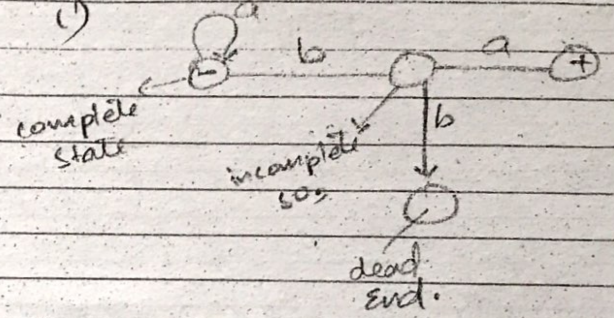
2)  (X)

*- aik state say given no of strings pass hongy. agar aik bhi miss huwa to FA (incomplete) ho ga.
i.e $\Sigma = \{a, b\}$

1)  (✓)

2)  (X)

*- Jb aisa ho ga to 2nd alpha deadend main chla jai ga.
i.e
$\Sigma = \{a, b\}$

1) 

Qs: Write RE and draw an FA for the language of all strings of a's and b's that have both the no of a's and b' are even include Λ.

Strings:

Λ
aa
aaaa
⋮
bb
bbbb
⋮
abba
aabb
abab

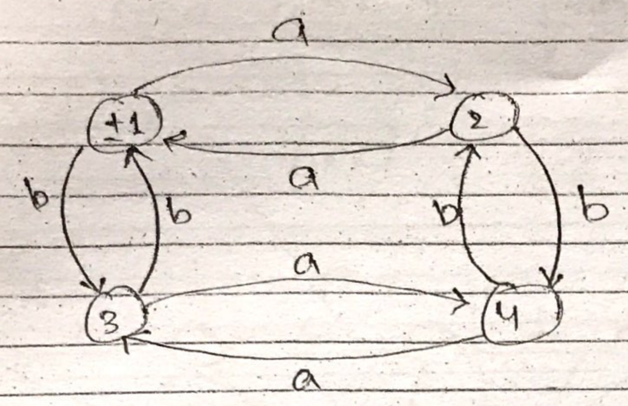
RE:-

$[aa + bb + (ab + ba)(aa+bb)^*(ab+ba)(aa+bb)^*$

OR

$(aa+bb)^* + [(aa+bb)^*(ab+ba)(aa+bb)^*(ab+ba)(aa+bb)^*$

FA :-



Q: We have to prove the string abaaba is an acceptable string in previous FA.

Solution:

$\delta(1, ab) = \delta(\delta(1, a), b) = \delta(2, b) = 4$

$\delta(1, aba) = \delta(\delta(1, ab), a) = \delta(4, a) = 3$

$\delta(1, abaa) = \delta(\delta(1, aba), a) = \delta(3, a) = 4$

$\delta(1, abaab) = \delta(\delta(1, abaa), b) = \delta(4, b) = 2$

$\delta(1, abaaba) = \delta(\delta(1, abaab), a) = \delta(2, a) = 1$

So, the string starting initial state and end with Final state so it's proved.

NOTE:-

*– prove string waly question main given string initial state say start ho kar Final state ma khatam ho gi.

*– Agar Final say pehlay hi khatam ho jai to wo prove nahi kehlai gi.

*– Answer start karty waqt string kay 1st say (2) alphabet select kerain gay.

i.e given string abaaba

select two alphabets

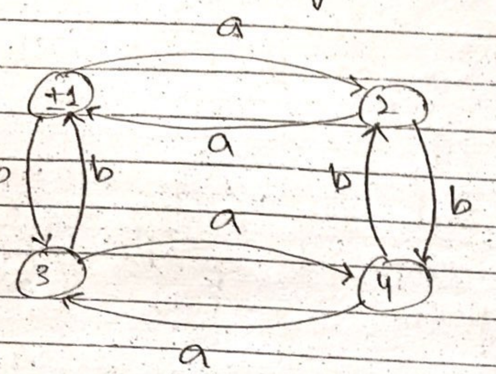
state one (initial)

$\delta(1, ab) = \delta(\delta(1, a), b) = \delta(2, b) = 4$

1. one pay (a) aye ga to state (2) main jai ga.

2. pay (b) aye ga to state (4) main jai ga.

Q5: We have to show whether baabba is an acceptable string or not the given FA.



Solution:

$\delta(1, ba) = \delta(\delta(1, b), a) = \delta(3, a) = 4$
$\delta(1, baa) = \delta(\delta(1, ba), a) = \delta(4, a) = 3$
$\delta(1, baab) = \delta(\delta(1, baa), b) = \delta(3, b) = 1$
$\delta(1, baabb) = \delta(\delta(1, baab), b) = \delta(1, b) = 3$
$\delta(1, baabba) = \delta(\delta(1, baabb), a) = \delta(3, a) = 4$

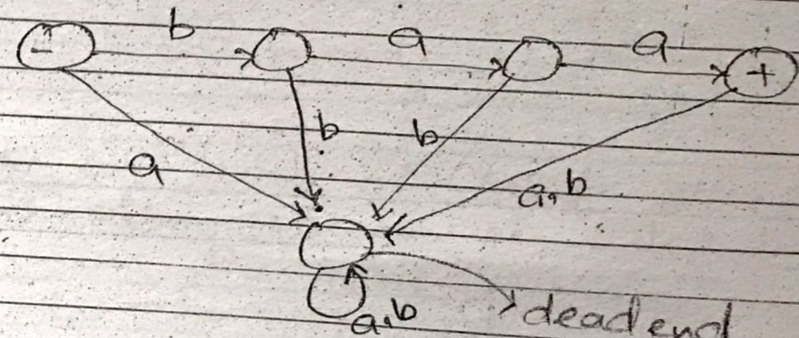
This string not end with the Final state so its not proved.

Q6: Construct an FA for the lang that only certain baa.

Solution.

RE = baa

FA :-

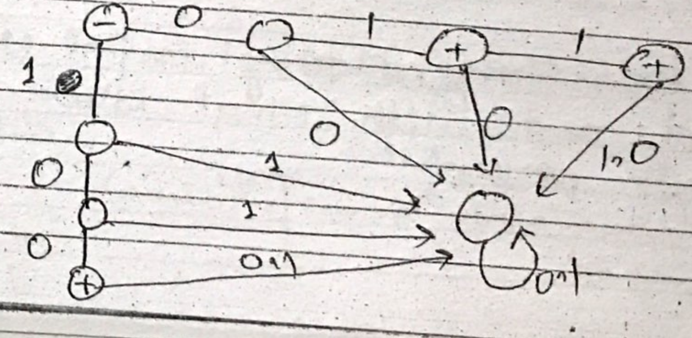


Q7: Construct an FA that only accept (01, 011, 100)

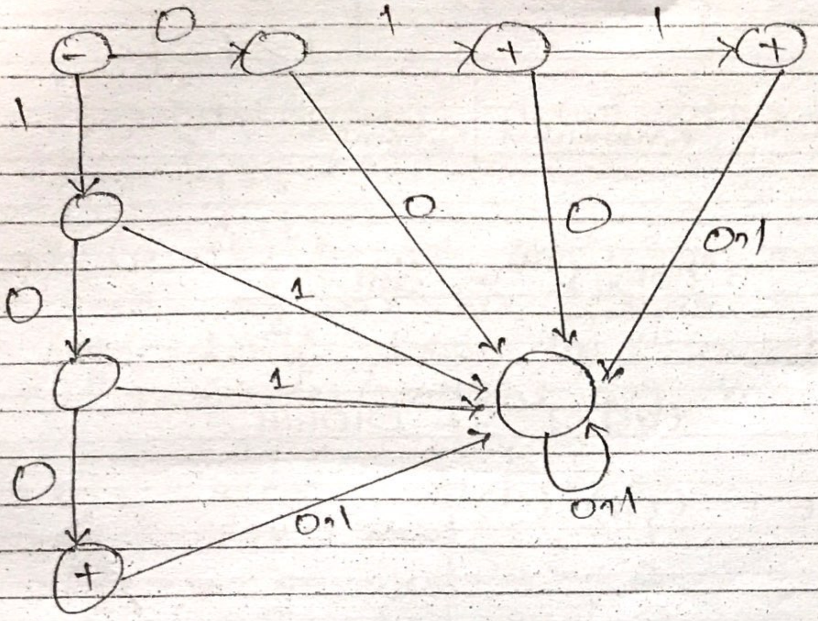
$\Sigma = \{0, 1\}$.

RE = (01 + 011 + 100)

FA:-



again draw FA:-

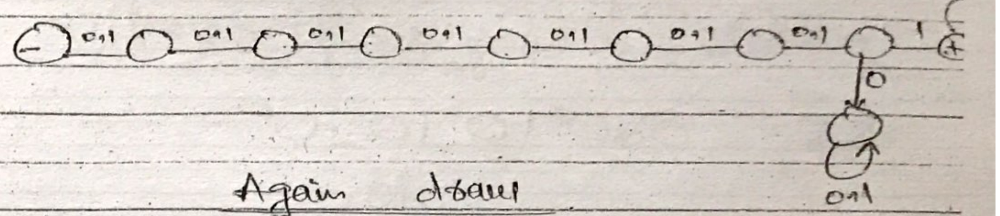


Q: Construct an FA for the lang of strings of 0's and 1's in which the tenth symbol from the right end is '1'.
$\Sigma = \{0,1\}$.

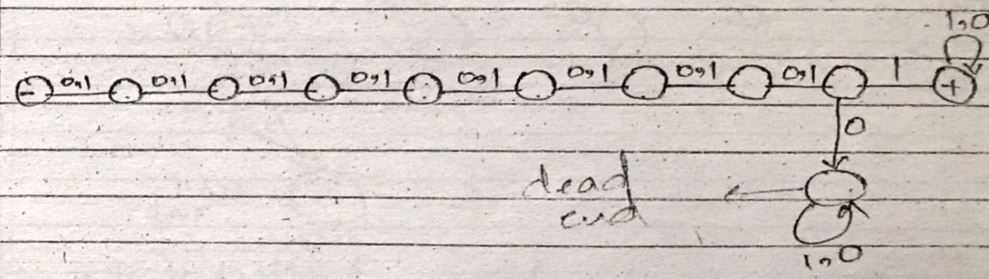
RE → $(0+1)^* 1 (0+1)^9$ (OR)

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

FA:-



Again draw



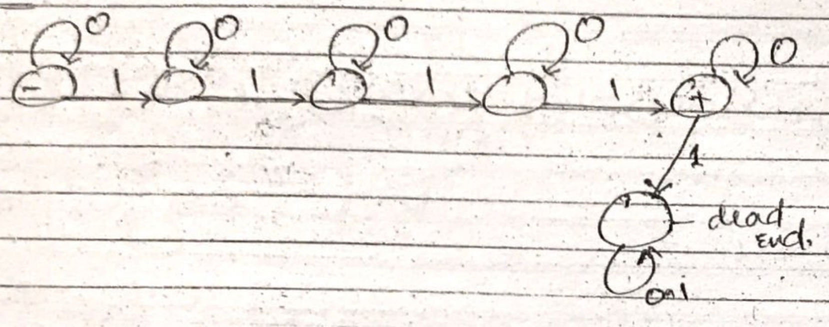
21/10/16 :- NOTE:
(FA → DFA)

Lecture (7)

Q: Design an FA which except set of Strings containing Exactly Four 1's in every string. $\Sigma = \{0,1\}$

RE = $0^*10^*10^*10^*10^*$

FA :

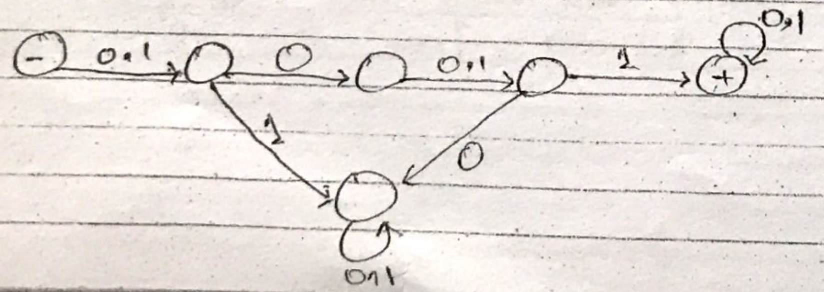


Q2) Design a DFA for the language.

$L = \{ w \in (0,1)^* \mid$ second symbol of w is 0 and Fourth symbol of w is 1 $\}$

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

FA :-



Q3:- Design a DFA for the lang:

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

RE = Minimum String :-

$i = 1$
$j = 1$

$(01)^1 1^{2(1)} = 0111$

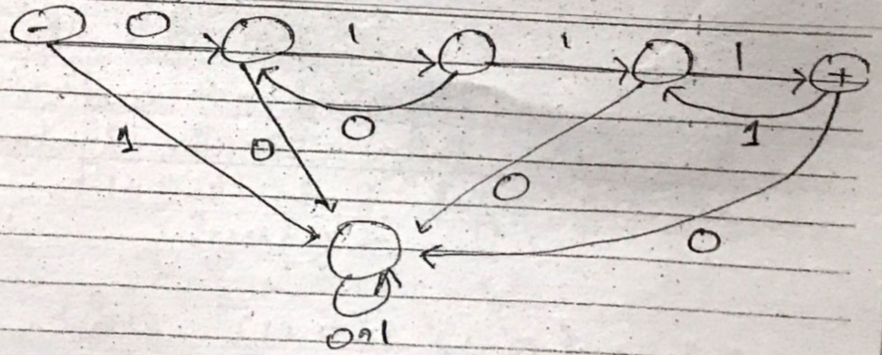
$i = 2$
$j = 2$

$(01)^2 1^{2(2)} = 010111 1$

⋮

RE = $(01)^+(11)^+$.

FA =



Q4) Design a DFA for the Language.

remainder

$$L = \{ w \in (a,b)^* \mid n_b(w) \bmod 3 > 1 \}$$

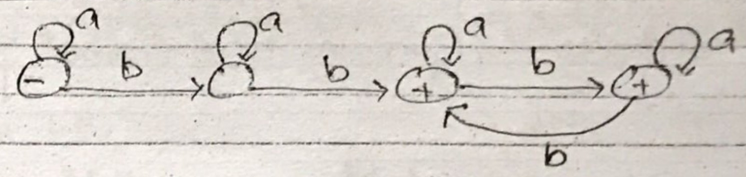
*) (a or b) ki asi string jis main num of b's ko agar 3 say divide krain to us ka remainder (mod) 1 say bra ho.

① × 0 mod 3 = 0
② × 1 mod 3 = 1
③ ✓ 2 mod 3 = 2
④ × 3 mod 3 = 0
⑤ × 4 mod 3 = 1
⑥ ✓ 5 mod 3 = 2

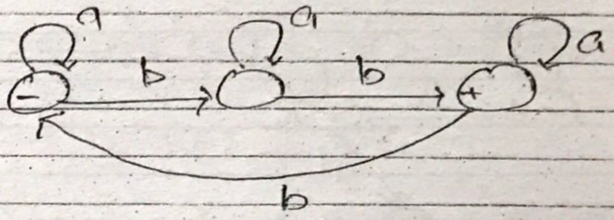
minimum no of b's 2, 5 ....

① $3\overline{)0}$ ② $3\overline{)10}$ ③ $3\overline{)20}$
④ $3\overline{)3}$ ⑤ $3\overline{)4}$ ⑥ $3\overline{)5}$

FA:-



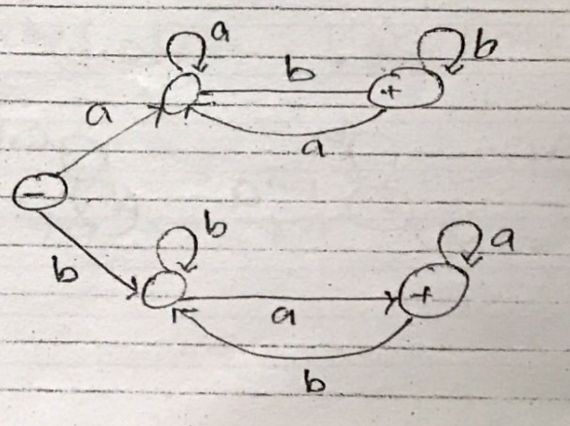
OR



Q5) Design an FA which accept the string that have different first and Last letter. $\Sigma = \{a,b\}$.

$RE = a(a+b)^*b + b(a+b)^*a.$

FA:-

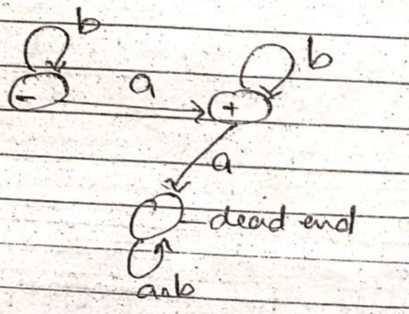


Q6: Design DFA for the Language

$L = \{ w : n_a = 1 , w \in (a,b)^* \}$

RE = $1(0)^*$ or $a(b)^*$

FA :-

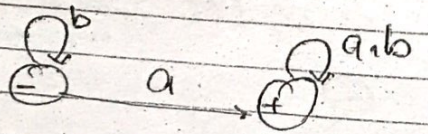


Q7) Design FA for the lang.

$L = \{ w : n_a \geq 1 , w \in (a,b)^* \}$

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

FA :-

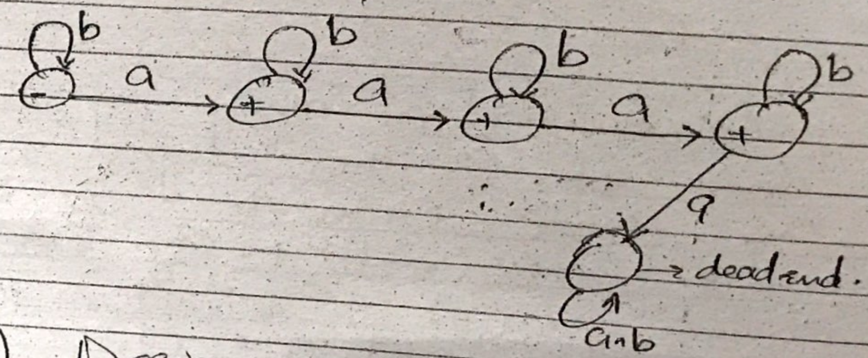


Q8) Design FA for the language.

$L = \{ w : n_a \leq 3 , w \in (a,b)^* \}$.

RE = $(a + aa + aaa)$

FA :-



Q9) Design FA for the language.

$L = \{ ab^5 w b^4 : w \in (a,b)^* \}$.

RE = $a(bbbbb)(a+b)^*(bbbb)$.

FA → next page

4/11/16 :-

Lecture (8)

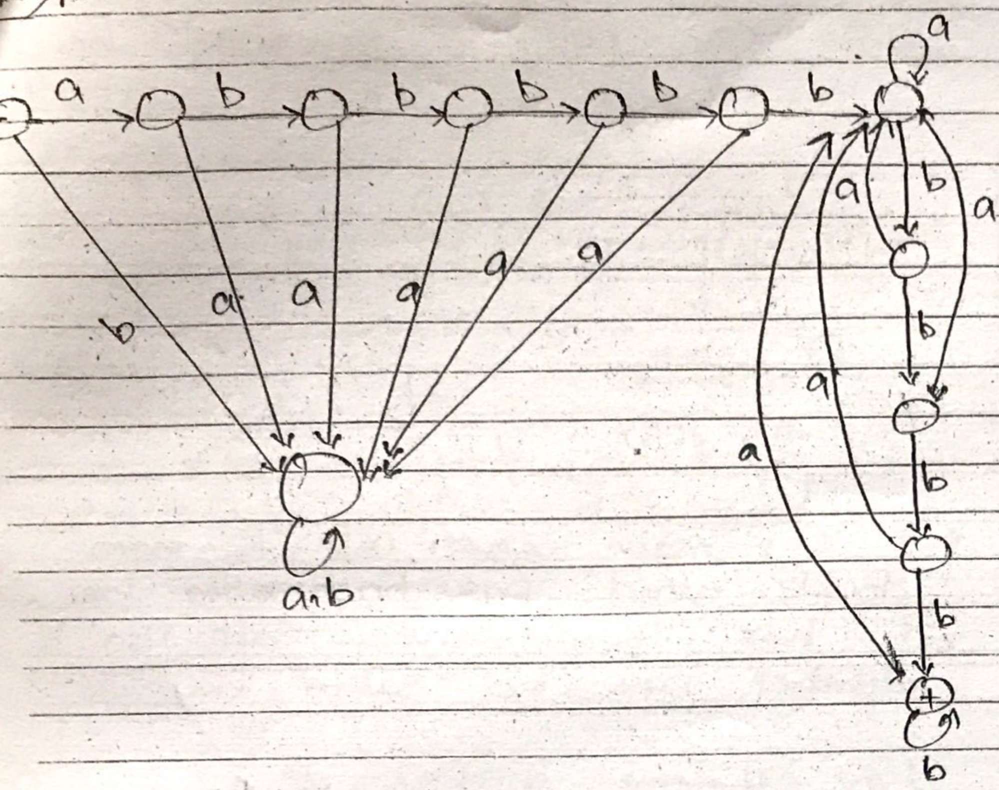
Transition Graph: (TG)

IN FA	IN TG
$\bigcirc \xrightarrow{aa} \bigcirc$ (X)	$\bigcirc \xrightarrow{aa} \bigcirc$ (✓)
$\bigcirc \xrightarrow{a} \bigcirc \xrightarrow{a} \bigcirc$ (✓)	$\bigcirc \xrightarrow{a} \bigcirc \xrightarrow{a} \bigcirc$ (✓)
$\bigcirc \xrightarrow{\Lambda} \bigcirc$ (X)	$\bigcirc \xrightarrow{\Lambda} \bigcirc$ (✓)
$\ominus$ (null state) ✓	$\ominus$ null state (X)
$\bigcirc \xrightarrow{a} \bigcirc$, $\bigcirc \xrightarrow{a} \bigcirc$ (X)	$\bigcirc \xrightarrow{a} \bigcirc$, $\bigcirc \xrightarrow{a} \bigcirc$ (✓) multiple allow

*) → Har FA TG Ho sakti hai But Har TG FA Nahi ho sakti.

FA → TG (✓)
TG → FA (X).

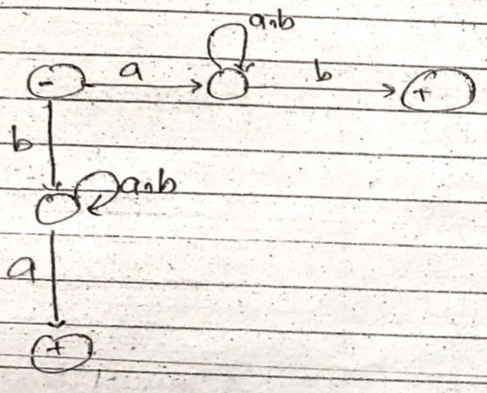
A) :-



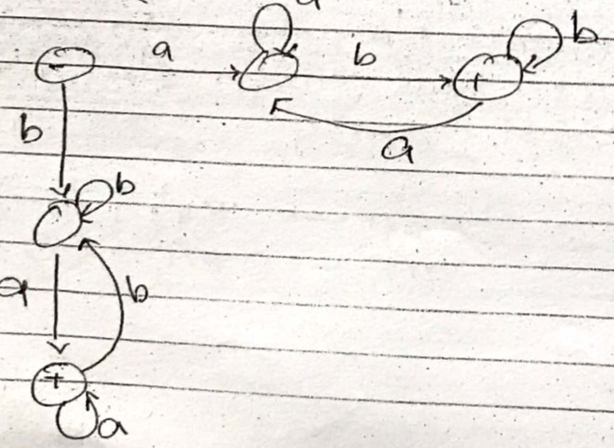
Q: Construct a TG for the language of those string which have different first and last letter. $\Sigma = \{a, b\}$.

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

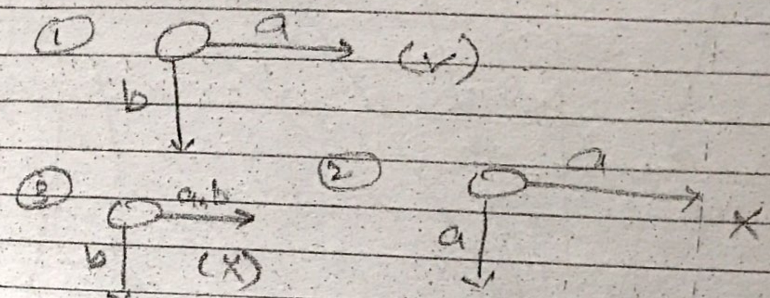
TG:-



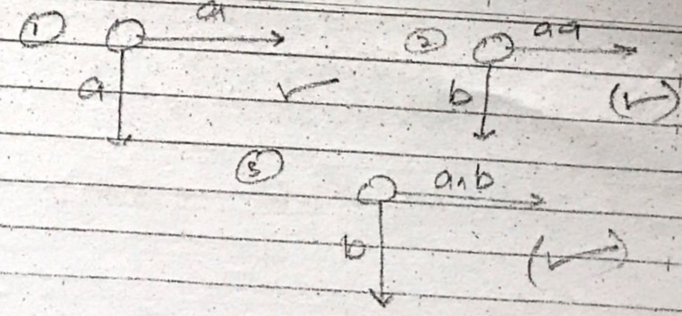
FA:-



*→ FA main zaroori tha kay her state say unique Transition hogi



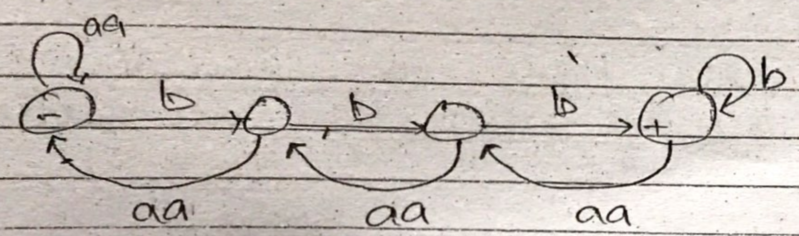
*→ TG main zaroori nahi. TG main double word pass ho sakta hai or her Transition main aik jisa alphabet pass ho sakta hai.



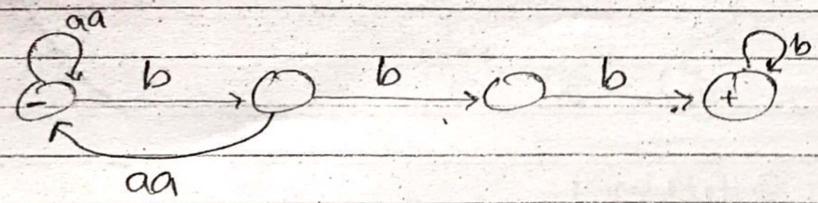
Q2: Construct TG for the language that have three or more b's and a occur in even climps. $\Sigma = \{aa, b\}$.

$RE = b^*(aa)^*b(aa)^*b(aa)^*b(aa)^*b^*$

TG:



OR



KLeen's Theorem:

RE → FA → TG (Relationship)

Part 1:

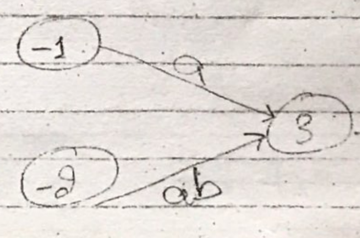
Every language that can be defined by an FA that can also be defined by TG.

Proof:-

Since Every FA is also a TG so therefore any language that can be defined by an FA that can also be defined by a TG.

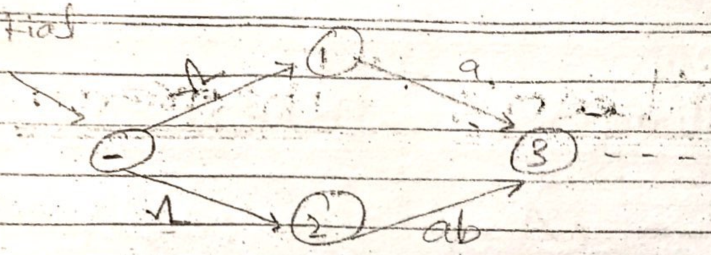
Part 2:

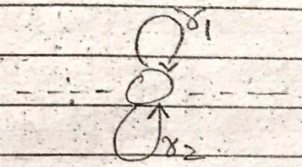
Every language that can be defined by a TG that can also be defined by a RE.



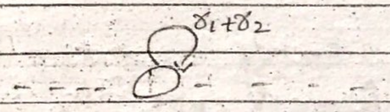
Two initial state

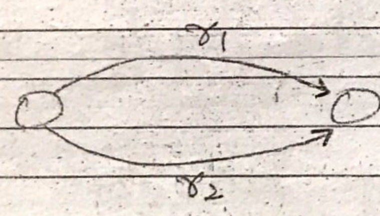
our initial state



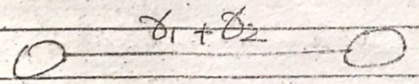


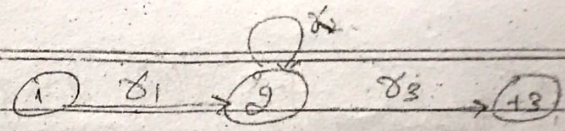
Two loops Convert to one loop.





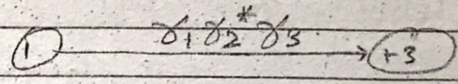
Two transitions Convert into one Transition.



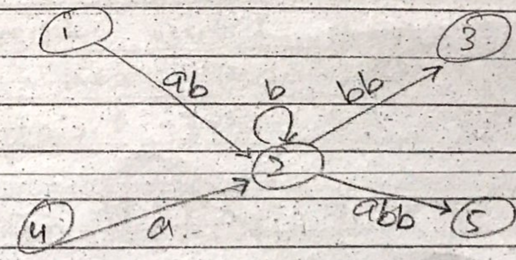


Using By pass operation:-

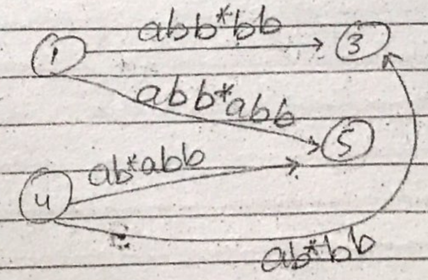
Skip r_2:-



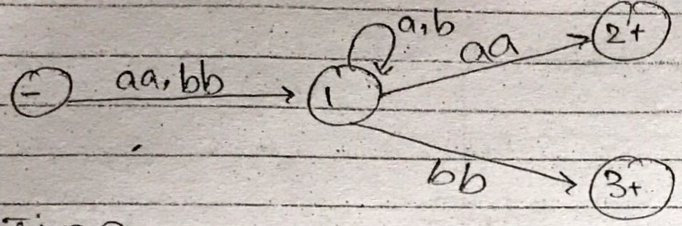
Q: we have to Eliminate State 2 by using By pass operation.



Solution:

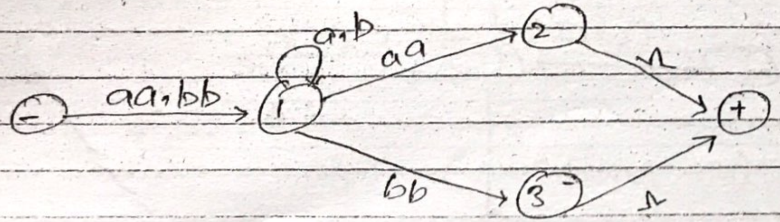


Q:- The Following graph Convert to RE:-

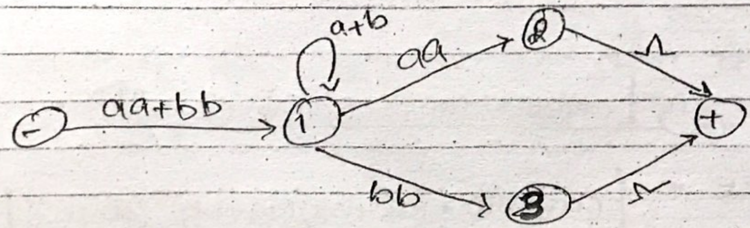


Solution:-

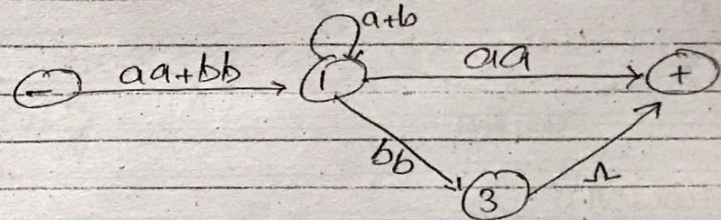
Step NO 1:-
Introduce a Unique Final State.



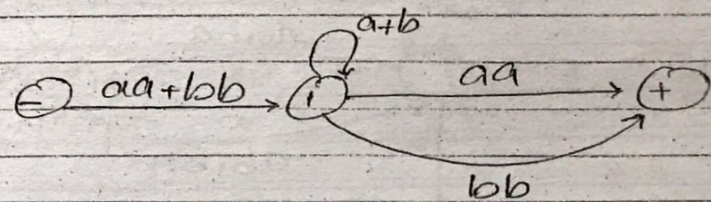
Step NO 2:
Replace the edges with RE.



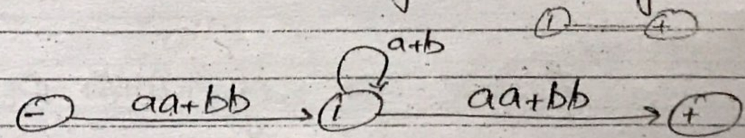
STEP NO 3:-
Eliminate State (2)



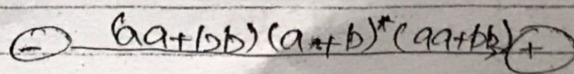
STEP NO 4:-
Eliminate State (3)



STEP NO 5:-
Combining the edges:

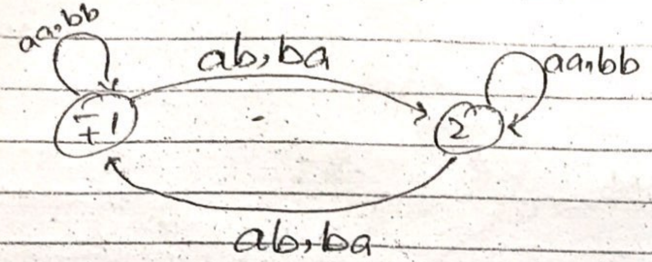


Step NO 6:-
Eliminate State (1)



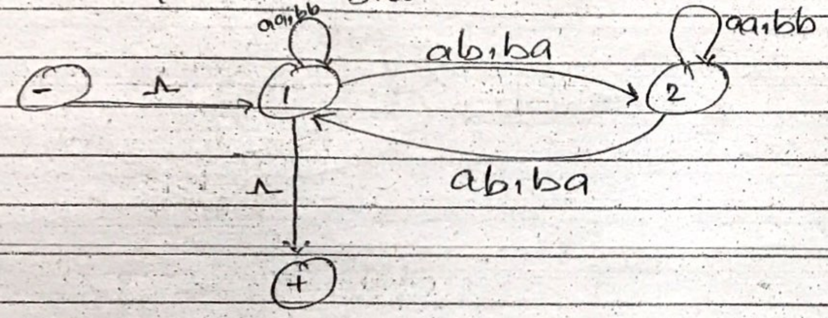
RE = $(aa+bb)(a+b)^*(aa+bb)$ Ans

Q: Convert into RE:-

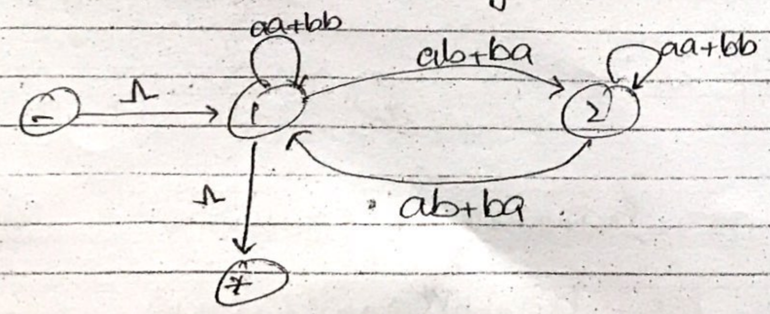


Solution:-

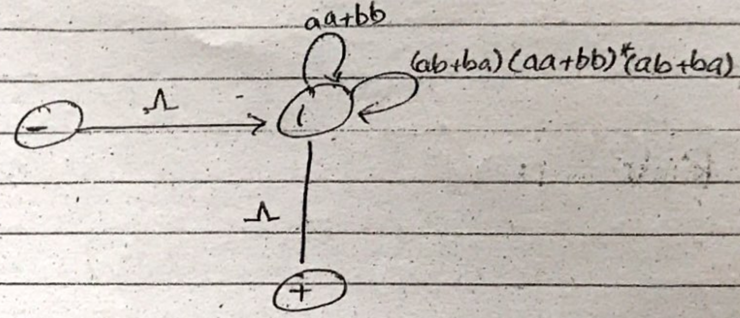
STEP NO 1:- Seprate the initial and Final State:



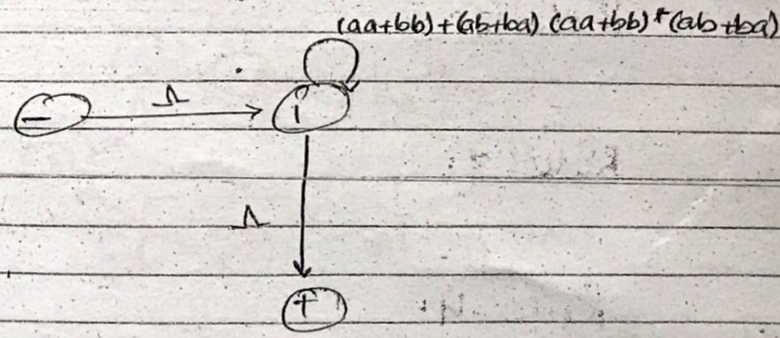
STEP NO 2:-
Replace the edges with RE



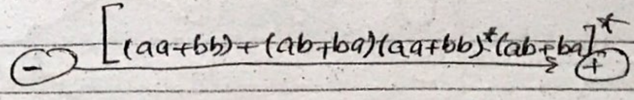
STEP NO 3:-
Eliminate State 2



STEP NO 4:-
Combining the edges.



STEP NO 5:-
Eliminate State 1



$$RE = \left[(aa+bb)+(ab+ba)(aa+bb)^*(ab+ba)\right]^*$$

Lecture (9)

part #3 :-

Every language that can be defined by a RE that can also be defined by an FA:

Rule – 1:

Null → (Λ) is a RE,

Any letter (α) is the RE

IF r_1 and r_2 than,

Rule - 2:

$r_1 + r_2$ (Add)

Rule - 3:

$r_1 r_2$ (concatenation)

Rule - 4:

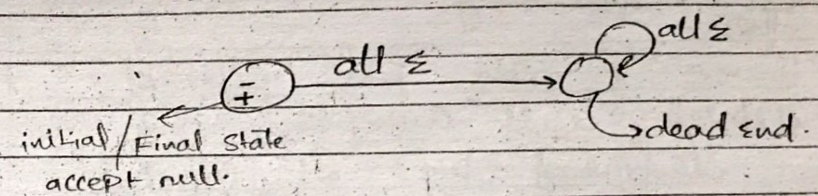
r_1^* (accept Null).

The above Rule-2, Rule-3 and Rule-4 the r_1 and r_2 also a RE that can accept FA.

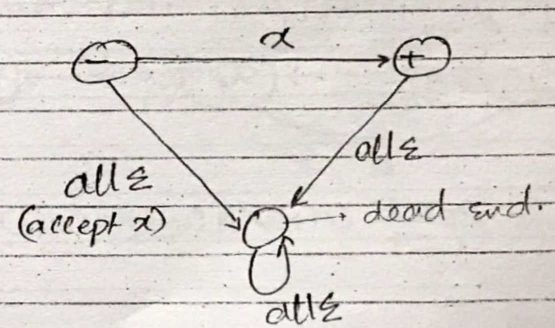
Explain:-

part #3 (Rule - 1):

There is an FA which accepts only (Λ.)



There is an FA which only accepts α.

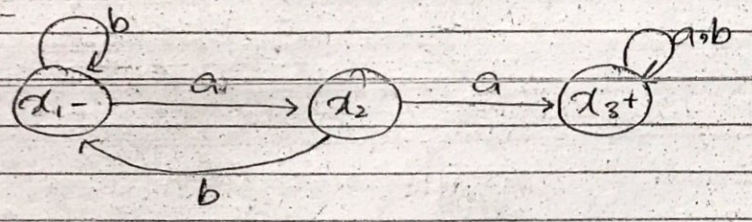


part #3 (Rule-2) (r_1+r_2)

If there is an FA called FA_1, which accept the language defined by RE r_1, there is an FA called FA_2 which accept the language defined by RE r_2,
then,
there is an FA called FA_3 which will accepts the language defined by RE (r_1+r_2).

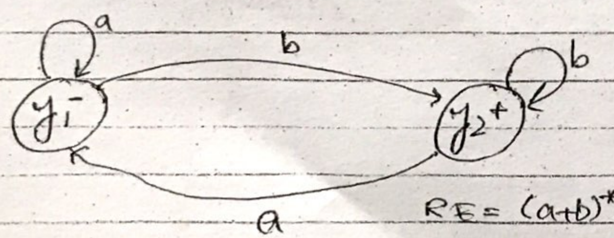
Example:-

FA_1 :-



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

FA_2:-



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

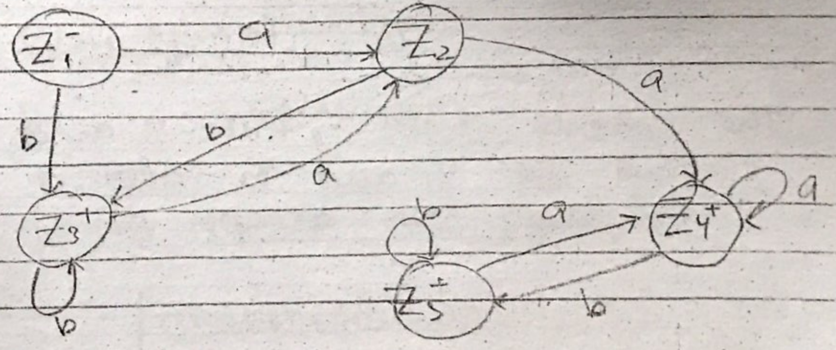
Let the States for FA_3 are $Z_1, Z_2, ----$.

*) Take initial State For FA_1 and FA_2

$-Z_1 = x_1, y_1$

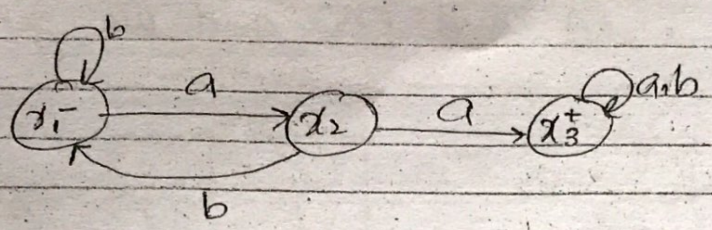
new State	inputs a	inputs b
$Z_1^- \equiv (x_1, y_1)$	$Z_2 \equiv (x_2, y_1)$	$Z_3 \equiv (x_1, y_2)$
$Z_2 \equiv (x_2, y_1)$	$Z_4 \equiv (x_3, y_1)$	$Z_3 \equiv (x_1, y_2)$
$Z_3^+ \equiv (x_1, y_2)$	$Z_2 \equiv (x_2, y_1)$	$Z_3 \equiv (x_1, y_2)$
$Z_4^+ \equiv (x_3, y_1)$	$Z_4 \equiv (x_3, y_1)$	$Z_5 \equiv (x_3, y_2)$
$Z_5^+ \equiv (x_3, y_2)$	$Z_4 \equiv (x_3, y_1)$	$Z_5 \equiv (x_3, y_2)$

Resultant FA_3:- $(FA_1 + FA_2)$

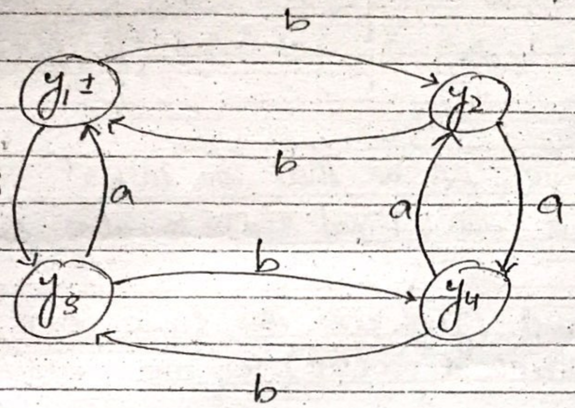


Example (2)

FA1:-



FA2:-



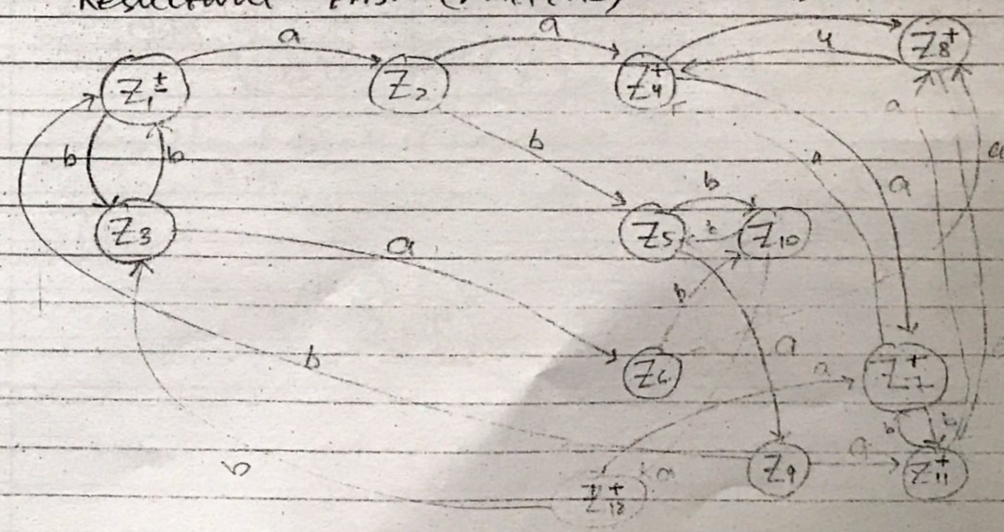
Solution:-

For initial state :- FA_1 ki initial or FA_2 ki initial state pick krain gy.

For Final state :-
FA_1 ki Final or FA_2 ki Final point jis jis new state main hongy wo Final State ho gi.

new state	inputs a	b
$^{+}Z_1^- \equiv (x_1, y_1)$	$Z_2 \equiv (x_2, y_3)$	$Z_3 \equiv (x_1, y_2)$
$Z_2 \equiv (x_2, y_3)$	$Z_4 \equiv (x_3, y_1)$	$Z_5 \equiv (x_1, y_4)$
$Z_3 \equiv (x_1, y_2)$	$Z_6 \equiv (x_2, y_4)$	$Z_1^- \equiv (x_1, y_1)$
$^{+}Z_4 \equiv (x_3, y_1)$	$Z_7 \equiv (x_3, y_3)$	$Z_8 \equiv (x_3, y_2)$
$Z_5 \equiv (x_1, y_4)$	$Z_9 \equiv (x_2, y_2)$	$Z_{10} \equiv (x_1, y_3)$
$Z_6 \equiv (x_2, y_4)$	$Z_8 \equiv (x_3, y_2)$	$Z_{10} \equiv (x_1, y_3)$
$^{+}Z_7 \equiv (x_3, y_3)$	$Z_4 \equiv (x_3, y_1)$	$Z_{11} \equiv (x_3, y_4)$
$^{+}Z_8 \equiv (x_3, y_2)$	$Z_{11} \equiv (x_3, y_4)$	$Z_4 \equiv (x_3, y_1)$
$Z_9 \equiv (x_2, y_2)$	$Z_{11} \equiv (x_3, y_4)$	$Z_1^- \equiv (x_1, y_1)$
$Z_{10} \equiv (x_1, y_3)$	$Z_{12} \equiv (x_2, y_1)$	$Z_5 \equiv (x_1, y_4)$
$^{+}Z_{11} \equiv (x_3, y_4)$	$Z_8 \equiv (x_3, y_2)$	$Z_7 \equiv (x_3, y_3)$
$^{+}Z_{12} \equiv (x_2, y_1)$	$Z_7 \equiv (x_3, y_3)$	$Z_3 \equiv (x_1, y_2)$

Resultant FA_3:- ($FA_1 + FA_2$)

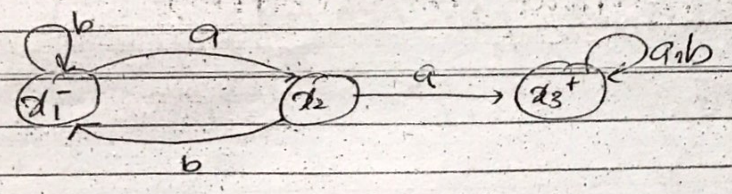


part # 3 (Rule - 3)

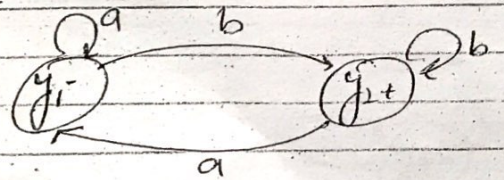
If there is an FA called FA_1, which accept the Language defined by RE r_1. there is an FA called FA_2 which accept the Language defined by RE r_2, then.
there is an FA called FA_3 which will accepts the language defined by RE $(r_1 r_2)$ Concatenation.

Example:-

FA_1:-



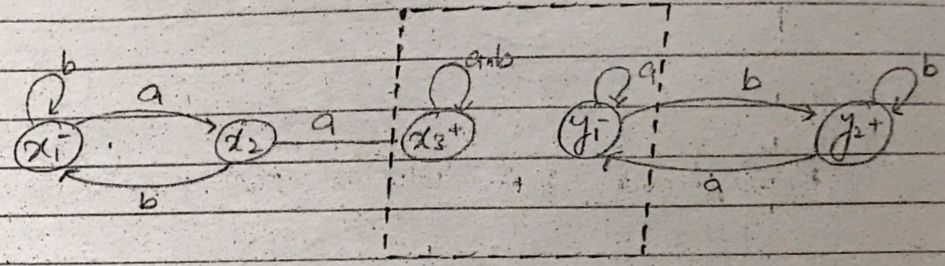
FA_2:



Solution:-

Let we Concatenate the Two FA's FA_1 and FA_2 and we get FA_3.

so,



*) The above FA in that the initial state is one (x_1^-) and the Final state is also one (y_2^+).

*) x_3^+ and y_1^- are the same point when we reached x_3^+ so, we write this.

$$Z \equiv (x_3, y_1)$$

we reached point y_1^- so, we write this

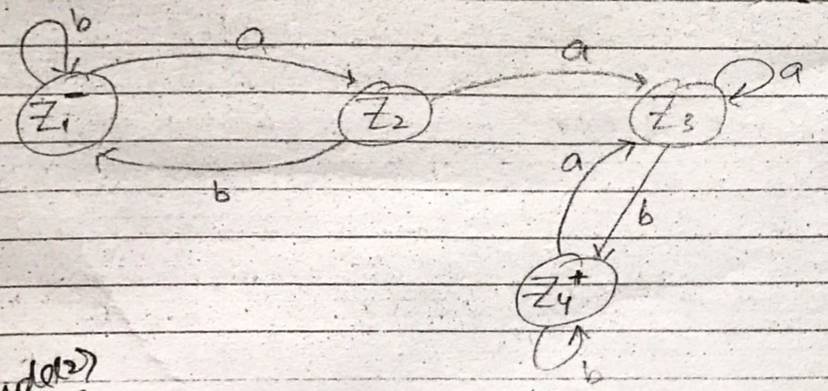
$$Z = (y_1, x_3)$$

so,

$$(x_3, y_1) \equiv (y_1, x_3).$$

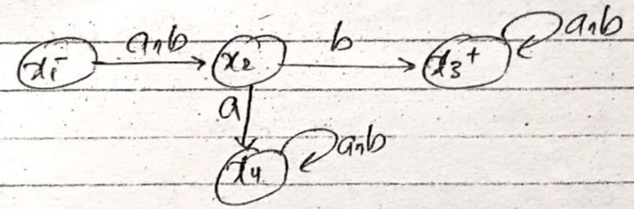
new States	inputs a	inputs b
$Z_1^- \equiv x_1$	$Z_2 \equiv x_2$	$Z_1 \equiv x_1$
$Z_2 \equiv x_2$	$Z_3 \equiv (x_3, y_1)$	$Z_1 \equiv x_1$
$Z_3 \equiv (x_3, y_1)$	$Z_3 \equiv (x_3, y_1)$	$Z_4 \equiv (x_3, y_1, y_2)$
$Z_4 \equiv (x_3, y_1, y_2)$	$Z_3 \equiv (x_3, y_1)$	$Z_4 \equiv (x_3, y_1, y_2)$

Resultant FA3 is:- (FA1 . FA2)

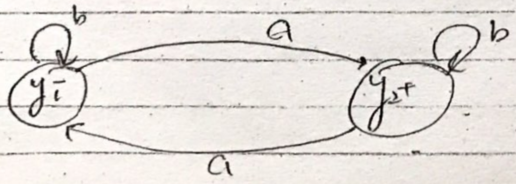


example(2)

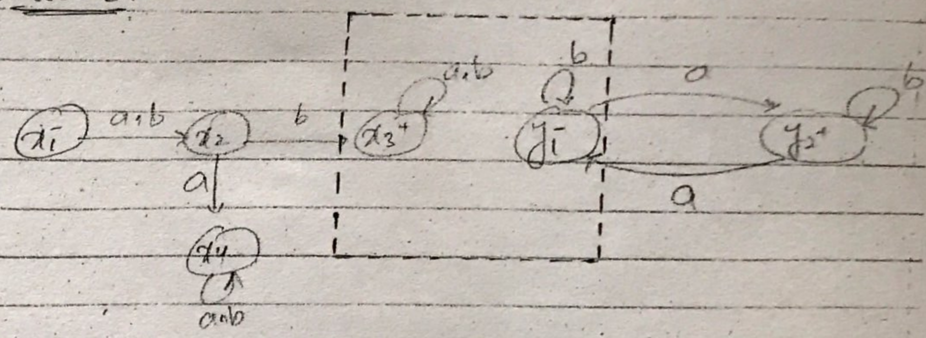
FA1:-



FA2:-

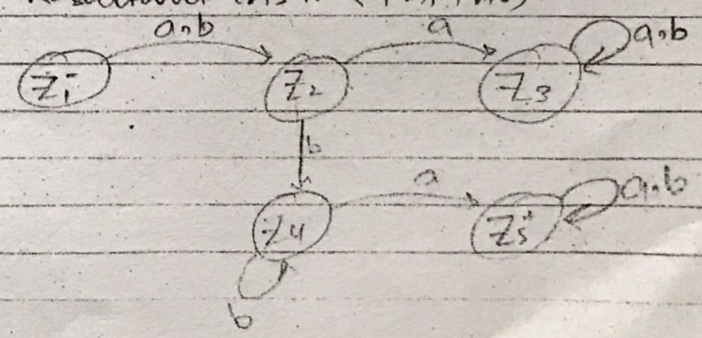


Solution:-



new State	inputs a	inputs b
$Z_1^- \equiv x_1$	$Z_2 \equiv x_2$	$Z_2 \equiv x_2$
$Z_2 \equiv x_2$	$Z_3 \equiv x_4$	$Z_4 \equiv (x_3, y_1)$
$Z_3 \equiv x_4$	$Z_3 \equiv x_4$	$Z_3 \equiv x_4$
$Z_4 \equiv (x_3, y_1)$	$Z_5 \equiv (x_3, y_1, y_2)$	$Z_4 \equiv (x_3, y_1)$
$^+Z_5 \equiv (x_3, y_1, y_2)$	$Z_5 \equiv (x_3, y_1, y_2)$	$Z_5 \equiv (x_3, y_1, y_2)$

Resultant FA3 is:(FA1 . FA2)



i.e

6 — k

12 13

$2k \to 2(6) = 12$ $2k+1 - 2(6)+1 = 13$

3 — k

6 7

$2(k) \to 2(3) = 6$ $2(k)+1 - 2(3)+1 = 7$