



INTERNAL ASSESSMENT PAPER (5th Sem)

Internal Test: 1	Internal Quiz: 1	Academic Year: ODD/ 2021-22
Subject : Automata theory and Computability	Sub-Code: 18CS54	Sem: V-(A,B,C)
Date: 19-11-2021	Time: 2.00-3.45	Dur: 105 min
Internal Test max marks: 50		Internal Quiz max marks: 10
Staff-Incharge: Dr Gopalakrishna MT/Prof.Chandrashekar D K		

I. Quiz (Answer all multiple-choice question in first sheet of your answer book)

Que sion No	<u>Multiple choice question</u>	BT level	CO Mapped
1	Finite automata are good models for computers with an extremely limited A. Memory ✓ B. Computation C. Size of the system D. None of the above	L2	CO1, PO1
2	How to distinguished the Finite state machine, Push down automata and Turing machine? A. Power of Computation ✓ B. Based on Complexity C. Based on Definition D. All the above	L2	CO1, PO1
3	How do you represent the Finite state machine? A. Transition Diagram B. Transitions Table C. Both (A) and (B) ✓ D. Only A	L2	CO1, PO1
4	A language is called a----- if some finite automaton recognizes it. A. Language B. Recognizer C. Transition D. Regular language ✓	L2	CO1, PO1
5	When the machine is in a given state and reads the next input symbol, we know what the next state will be call----- A. Deterministic ✓ B. Nondeterministic C. Non final state D. Intermediate state	L2	CO2, PO2

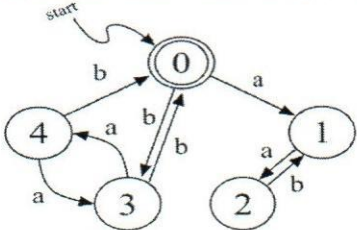
6	In a ----- machine, several choices may exist for the next state at any point A. Nondeterministic ✓ B. Deterministic C. Accepting state D. Initial state	L2	CO2, PO2
7	Every state of a DFA always has exactly -----transitions for each symbol in the alphabet. A. Two B. One ✓ C. Multiple D. Both (a) and (b)	L2	CO2, PO2
8	Nondeterminism may be viewed as a kind of A. Parallel Computation ✓ B. Sequential Computation C. Both (a) and (b) D. None of the above	L2	CO2, PO2
9	Major difference between DFA and NFA is A. In terms of Final state B. In terms of non-Final state C. In terms of transitions functions ✓ D. None of the above	L2	CO5, PO4
10	There is no way for it to reach an accepting state, so we already know that the string is going to be rejected- is called. A. Dead state ✓ B. Start state C. Final state D. Either (a) or (c)	L2	CO2, PO2

II. Internal Test (Answer any two full questions choosing one from each part)
(Each full question carries 25 marks)

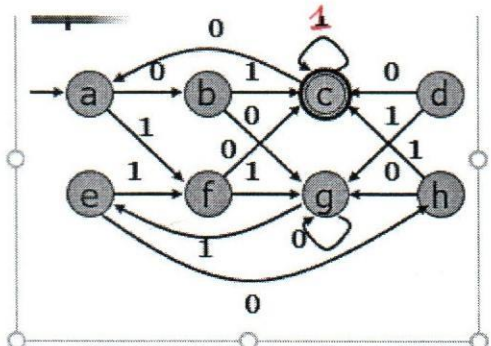
Main Ques	Sub Ques	Full Question	Marks	BT Levels	CO -PO Outcome																				
Part A																									
1.	A)	Design DFSM for the following Language: $L = \{w \in \{a, b\}^* : w \text{ does not end in } ba\}$.	10	L3	CO3,PO2, PO4																				
	B)	Define the following terms with an example for each: (i) Alphabets (ii) Strings (iii) Kleene star and Plus (iv) Language	5	L1	CO1, PO1																				
	C)	Design DFSM for the following Language: $L = \{w \in \{a, b\}^* : w \text{ contains even number of a's and odd number of b's}\}$	10	L3	CO3,PO2, PO4																				
OR																									
2.	A)	For each of the following NDFSM, use ndfsm to dfsm to construct an equivalent DFSM. <table><tr><td>δ</td><td>ϵ</td><td>a</td><td>b</td><td>c</td></tr><tr><td>$\rightarrow p$</td><td>φ</td><td>{p}</td><td>{q}</td><td>{r}</td></tr><tr><td>q</td><td>{p}</td><td>{q}</td><td>{r}</td><td>φ</td></tr><tr><td>*r</td><td>{q}</td><td>{r}</td><td>φ</td><td>{p}</td></tr></table>	δ	ϵ	a	b	c	$\rightarrow p$	φ	{p}	{q}	{r}	q	{p}	{q}	{r}	φ	*r	{q}	{r}	φ	{p}	10	L2	CO2, PO2, PO3
δ	ϵ	a	b	c																					
$\rightarrow p$	φ	{p}	{q}	{r}																					
q	{p}	{q}	{r}	φ																					
*r	{q}	{r}	φ	{p}																					

2.	B)	Mention the differences between DFSM, NDFSM and ϵ -NDFSM	5	L1	CO1, PO1
	C)	Define regular expression and I. Write the regular expression for string 0's and 1's which have at least one pair of consecutive 0's II. Find the Regular expression for strings of 0's and 1's which having 0 or more zero's or 0 or more 1's	10	L2	CO1, PO1

Part B

3.	A)	Minimize the following Finite State Machine by partition Methods 	10	L2	CO4, PO3
	B)	Define distinguishable and indistinguishable states	5	L2	CO1, PO1
	C)	Design a NDFSM for the following Language: 1. $L = \{w \in \{a, b\}^* : w = aba \text{ or } w \text{ is even}\}$ 2. $L = \{w \in \{a, b, c\}^* : \exists x, y \in \{a, b, c\}^* (w = xabcabb y)\}$.	10	L2	CO3, PO2, PO4

OR

	A)	Convert the given NDFSM to DFSM <table><tr><th>q</th><th>$\delta(q, 0)$</th><th>$\delta(q, 1)$</th></tr><tr><td>$\rightarrow 1$</td><td>$\{1\}$</td><td>$\{2\}$</td></tr><tr><td>2</td><td>$\{2,3\}$</td><td>$\{2\}$</td></tr><tr><td>3^*</td><td>$\{3\}$</td><td>$\{2,3\}$</td></tr></table>	q	$\delta(q, 0)$	$\delta(q, 1)$	$\rightarrow 1$	$\{1\}$	$\{2\}$	2	$\{2,3\}$	$\{2\}$	3^*	$\{3\}$	$\{2,3\}$	10	L2	CO2, PO2, PO3
q	$\delta(q, 0)$	$\delta(q, 1)$															
$\rightarrow 1$	$\{1\}$	$\{2\}$															
2	$\{2,3\}$	$\{2\}$															
3^*	$\{3\}$	$\{2,3\}$															
4.	B)	Minimize the following Finite State Machine by Table Filling Method 	10	L2	CO4, PO3												
	C)	Mention the five Applications of Finite State Automata.	5	L1	CO1, PO1												

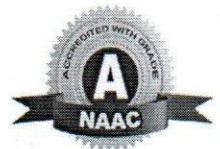
Comments:

* follow the format, marks distributed correct, no & foreign copy
mapping done. selected.


Signature of Faculty


Scrutinizer


Head

SCHEMES & SOLUTIONS

Internal Test: 1	Internal Quiz: 1	Academic Year: EVEN / 2021-22
Sub: Automata Theory and Computability	Sub-Code: 18CS54	Sem: V-(A,B,C)
Date: 19-11-2021	Time: 2-3.45	Dur: 105 min
Internal Test max marks: 50	Internal Quiz max marks: 10	
Staff-Incharge: Dr Gopalakrishna MT/Prof.Chandrashekar DK		

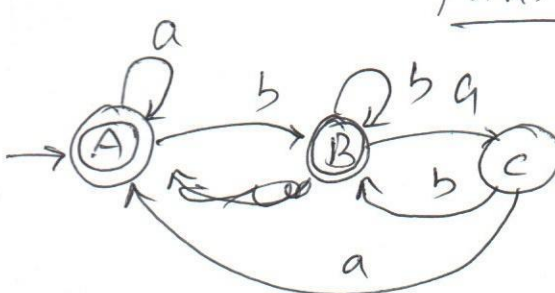
Comments:

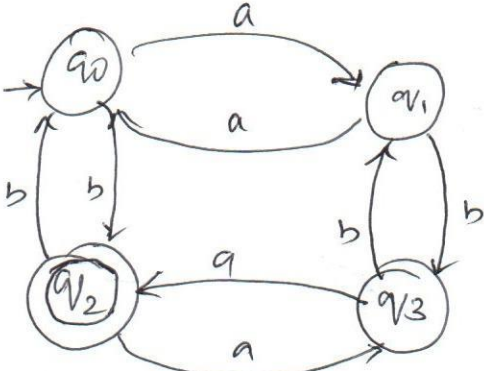
Solution by student. write in formal & more
doubts in each was done.

Signature of Faculty:

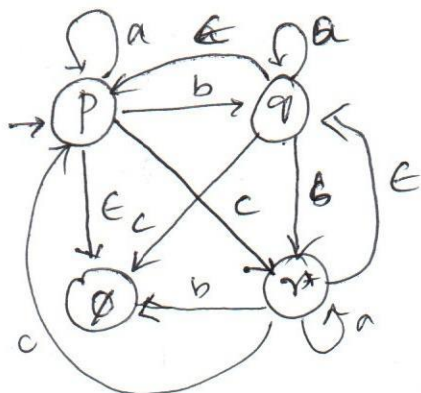
Signature of Scrutinizer:

Signature of HOD:

Q. No.		Marks Alloted
1	a	10x1=10
2	a	
3	c	
4	d	
5	a	
6	a	
7	b	
8	a	
9	c	
10	a	
<u>PART-A</u> 1 a)  $L = \{a, b, ab, baa, \dots\}$ Define $M = \{Q, \Sigma, \delta, q_0, F\}$		5 2 3 10

Q. No.		Marks Alloted
15	(i) Alphabet $\rightarrow$ finite set of input symbols. $\therefore \Sigma = \{a, b\}$ (ii) Strings $\rightarrow$ sequence of character/symbols. Ex. abc, 110, --- (iii) Kleen star $\rightarrow * \Rightarrow$ zero or more occurrence of symbols $a^* = \{\epsilon, a, aa, aaa, \dots\}$ & Kleen plus $\rightarrow + \Rightarrow$ one or more occurrence of symbols. $a^+ = \{a, aa, aaa, \dots\}$ (iv) Language $\Rightarrow$ A Language is set of finite infinite string over alphabet Σ^* $\Sigma = \{a, b\} = \Sigma^* = \{\epsilon, a, b, ab, ba, \dots\}$	1M 1M 2M 1M <u>5M</u>
16	(c)  $L = \{b, aab, aba, \dots\}$ Define $M = \{Q, \Sigma, \delta, q_0, F\}$	2M 5M 5M 3M <u>10</u>

2a)



$$E\text{closure}(p) = \{p, s\} \rightarrow A$$

$$\delta(A, a) = E\text{closure}(p) = \{p, s\} \rightarrow A$$

$$\delta(A, b) = E\text{closure}(q) = \{q, p, s\} \rightarrow B$$

$$\delta(A, c) = E\text{closure}(s)$$

$$E\text{closure}(q) = p, \{q, p, s\} \rightarrow B$$

$$E\text{closure}(s) = \{s, q, p, s\} \rightarrow C$$

$$\delta(A, a) = \delta(B, a/b/c)$$

$$\delta(A, b) = \delta(C, a/b/c)$$

$$\delta(A, c)$$

Detailed solution

$$\text{Definis } M = \{Q, \Sigma, \delta, q_0, f\}$$

2m

5 marks

2m

10 m.

Q. No.				Marks Alloted											
2(b)	DFSM (i) $Q \times \Sigma \rightarrow Q$ (ii) ^{or} $Q, 1, \dots$ on transition on each symbol (iii) no xly transition on every symbol. (iv) No ϵ' transition (v) can difficult to construct	NDFSM $Q \times \Sigma \rightarrow 2^Q$ 2^Q or, one, or more transition on each symbol. possess xly transition accept ϵ' easy	E-NDFSM. $Q \times (\Sigma \cup \epsilon) \rightarrow 2^Q$ -- do -- - do - Accept ϵ' easy	5x1m = 5m											
2(c)	Definition of RE (i) $(0+1)^* 00 (0+1)^*$ (ii) $0^* 1^*$	2M	4M 4m												
3(a)	Step 1: NO Dead States / missing state transition (2). Step 2: partition the states into two groups $\pi = (1, 2, 4, 3) (0) \epsilon$ Step 3: Obtain $\bar{\pi}_{\text{new}}$ from π. Consider $(1, 2, 4, 3)$ for $1/p$ a U b <table> <tr> <td>$\delta(1, a) = 2$</td> <td>$\delta(1, b)$</td> </tr> <tr> <td>$\delta(2, a) = \emptyset \epsilon'$</td> <td>$\delta(2, b)$</td> </tr> <tr> <td>$\delta(3, a) = 4$</td> <td>$\delta(3, b)$</td> </tr> <tr> <td>$\delta(4, a) = 3$</td> <td>$\delta(4, b)$</td> </tr> <tr> <td>$\delta(\epsilon, a) = \epsilon$</td> <td>$\delta(\epsilon, b) = \epsilon$</td> </tr> </table>			$\delta(1, a) = 2$	$\delta(1, b)$	$\delta(2, a) = \emptyset \epsilon'$	$\delta(2, b)$	$\delta(3, a) = 4$	$\delta(3, b)$	$\delta(4, a) = 3$	$\delta(4, b)$	$\delta(\epsilon, a) = \epsilon$	$\delta(\epsilon, b) = \epsilon$	10m 1M 2m 3m	final 5m 10m
$\delta(1, a) = 2$	$\delta(1, b)$														
$\delta(2, a) = \emptyset \epsilon'$	$\delta(2, b)$														
$\delta(3, a) = 4$	$\delta(3, b)$														
$\delta(4, a) = 3$	$\delta(4, b)$														
$\delta(\epsilon, a) = \epsilon$	$\delta(\epsilon, b) = \epsilon$														

3(b). A pair of states (p, q) is said to be distinguishable, if there is a string w such that either

$$\delta(p, w) \in F \text{ \& \& } \delta(q, w) \notin F$$

OR

$\delta(p, w) \notin F \text{ \& \& } \delta(q, w) \in F$, then (p, q) are distinguishable states

2 1/2

A pair of states (p, q) is said to be

indistinguishable, if there is a string w such

that either $\delta(p, w) \in F \text{ \& \& } \delta(q, w) \in F$

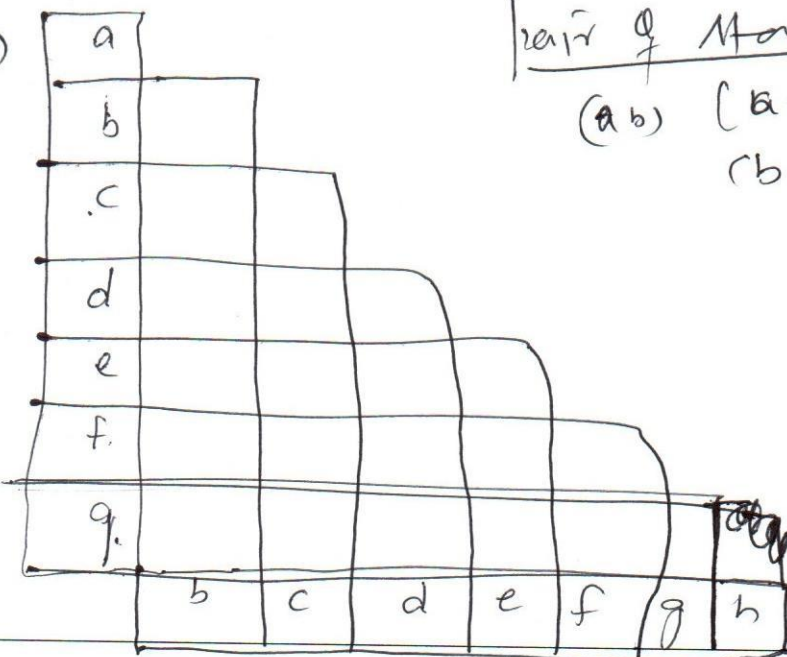
OR

$\delta(p, w) \in Q - F \text{ \& \& } \delta(q, w) \in Q - F$, then (p, q) are indistinguishable states.

2 1/2

5M

4(b)



pair of states. $\leftarrow$ 2M

(a b) (a c) (a d) (a e) (a f) (a g) (a h)
 (b c) (b d) (b e) (b f) (b g)
 (c d) (c e) (c f) (c g)
 (d e) (d f) (d g)
 (e f) (e g)
 (f g)

(2)

Table represents

4M

steps follow :- filling

4M

10M

Q. No.	Marks Alloted
3(c)	
(1) DFA $\rightarrow$ state diagram. ——— 5m	
(2) $\rightarrow$ define $M = \{Q, \Sigma, \delta, q_0, F\}$ — 3	
$\rightarrow$ any 1 form $\rightarrow$ 2	
10m	
(4)	
convert M steps ——— 3	
(a)	
state diagram $\rightarrow$ 3	
define $M = \{Q, \Sigma, \delta, q_0, F\}$ — 4	
10m	



SJBIT

|| Jai Sri Gurudev ||

Sri Adichunchanagiri Shikshana Trust®

SJB Institute of Technology

(Affiliated to Visvesvaraya Technological University, Belagavi and

Approved by AICTE and Accredited by NAAC with 'A' Grade, CGPA-3.22 - New Delhi)

#67, BGS Health & Education City, Dr. Vishnuvardhan Road, Kengeri, Bengaluru - 560060.

Website : www.sjbit.edu.in

INTERNAL ASSESSMENT BOOK

Student Name : Rajeshwari PatilSemester & Section : 5th 4 B USN : 1JB19CS108Subject: Automata theory & Computability Subject Code: 18CS54 Branch: CSEName of Faculty in charge : Dr. Gopalakrishna MT

Internal Assessment Test - I					Internal Assessment Test - II					Internal Assessment Test - III				
Q No.	Date : <u>19/11/21</u>				Q No.	Date : <u>28-12-21</u>				Q No.	Date : <u>28-1-22</u>			
	Max. Marks : _____					Max. Marks : <u>50</u>					Max. Marks : <u>50</u>			
	PART - A					PART - A					PART - A			
	A	B	C	Total		A	B	C	Total		A	B	C	Total
1	10	12	10	22	1	7	6	8	19	1	10	4	8	22
2					2					2				
PART - B					PART - B					PART - B				
3					3	10	5	10	25	3	10	10	5	25
4	10	10	5	25	4					4				
I Test IA Marks Total : <u>47</u>					II Test IA Marks Total : <u>44</u>					III Test IA Marks Total : <u>47</u>				
Quiz 1/Assignment etc., : <u>10</u>					Quiz 2/Assignment etc., : <u>04</u>					Quiz 3/Assignment etc., : <u>01</u>				
Student Signature : <u>Rajeshwari</u>					Student Signature : <u>Rajeshwari</u>					Student Signature : <u>Rajeshwari</u>				
Signature of Invigilator : <u>Dr. M.</u>					Signature of Invigilator : <u>[Signature]</u>					Signature of Invigilator : <u>[Signature]</u>				
Signature of Faculty in charge : <u>[Signature]</u>					Signature of Faculty in charge : <u>[Signature]</u>					Signature of Faculty in charge : <u>[Signature]</u>				
Avg. IA Marks for <u>30</u> (A) : <u>27.6</u>					Assignment /Quiz etc., for <u>10</u> (B) : <u>5</u>					Total IA Marks for <u>40</u> (A+B) : <u>33</u>				

HOD

Principal

Department of CSE

Dept. Vision:

To become a Centre of Excellence producing "creators of Innovative technology" who can contribute positively to the ever changing & industrial demands and societal needs.

Dept. Mission:

To encourage participation of faculty and students in research activities for enhancing their subject knowledge & acquire info. regarding current trends. To provide exposure to students on latest tools & technologies in areas of CSE. Preparation of our graduates for leadership in profession and in higher education by providing excellent teaching learning environment enabling them to serve the society.

About Anti - Ragging

SJBIT has zero tolerance policy for ragging. The Institute views ragging, is an uncivilized, and inhuman practice. We do not subscribe to the view that one could wait till something happens in order to initiate stringent action. Any rigorous action in such cases may damage a young career. So we repose faith in averting such eventualities. For this, the Institute has proactive policy.

Punishments for Ragging

1. Cancellation of admission.
2. Suspension from attending classes.
3. Withholding/withdrawing scholarship/fellowship and other benefits.
4. Debarring from appearing in any test/examination or other evaluation process.
5. Withholding results.
6. Debarring from representing the University in any regional, national or international meet, tournament, youth festival etc.
7. Suspensions/expulsion from the hostel.
8. Rustication from the college and University for period varying from 1 to 4 years.
9. Expulsion from the college and consequent debarring from admission to any other college.
10. Rigorous imprisonment of three years and/or a fine of upto Rs.25,000.
11. Collective punishment: When the persons committing or abetting the crime or ragging are not identified, the institution has resort to collective punishment as a deterrent to ensure community pressure on the potential raggers.

- 1 A Memory ✓
- 2 A. Power of Computation ✓
- 3 C Both (A) & (B) ✓
- 4 D Regular Language ✓
- 5 A Deterministic ✓
- 6 A Nondeterministic ✓
- 7 B One ✓
- 8 A Parallel Computation ✓
- 9 C In terms of transition functions ✓
- 10 A Dead state ✓

10
50

Am

PART-B

4A)

NDFSM to DF5M

Given:

q	$\delta(q, 0)$	$\delta(q, 1)$
→ 1	{1}	{2}
2	{2, 3}	{2}
3	{3}	{2, 3}

Transition table

	0	1
→ {1}	{1}	{2}
{2}	{2, 3}	{2}
{2, 3}	$\{2, 3\} \cup \{3\} = \{2, 3\}$	$\{2\} \cup \{2, 3\} = \{2, 3\}$
{3}	{3}	{2, 3}

Consider the start state 1 and final state as 3 in above table, steps followed are

$$\rightarrow \delta(\{1\}, 0) = \{1\}$$

$$\delta(\{1\}, 1) = \{2\}$$

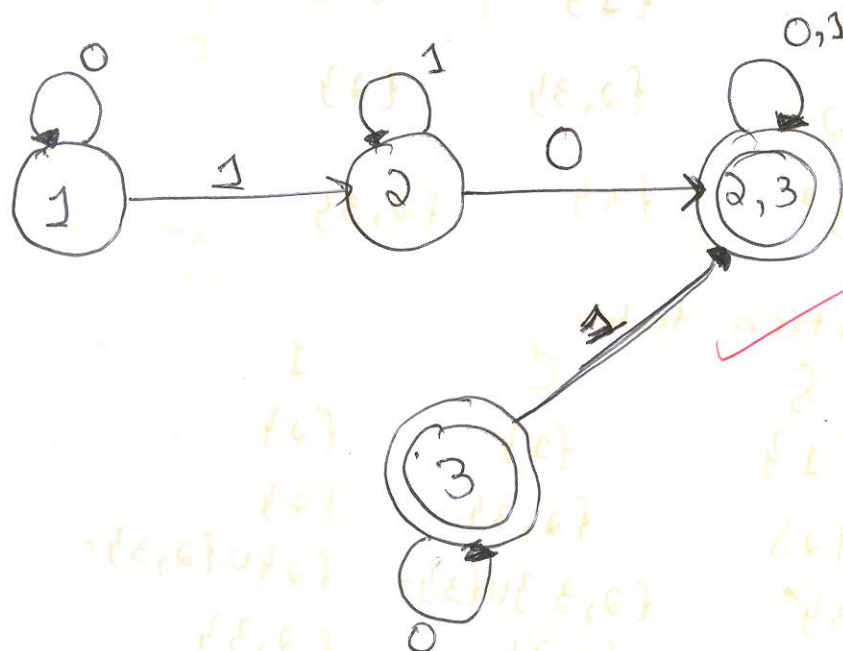
$$\rightarrow \delta(\{2\}, 0) = \{2, 3\}$$

$$\delta(\{2\}, 1) = \{2\}$$

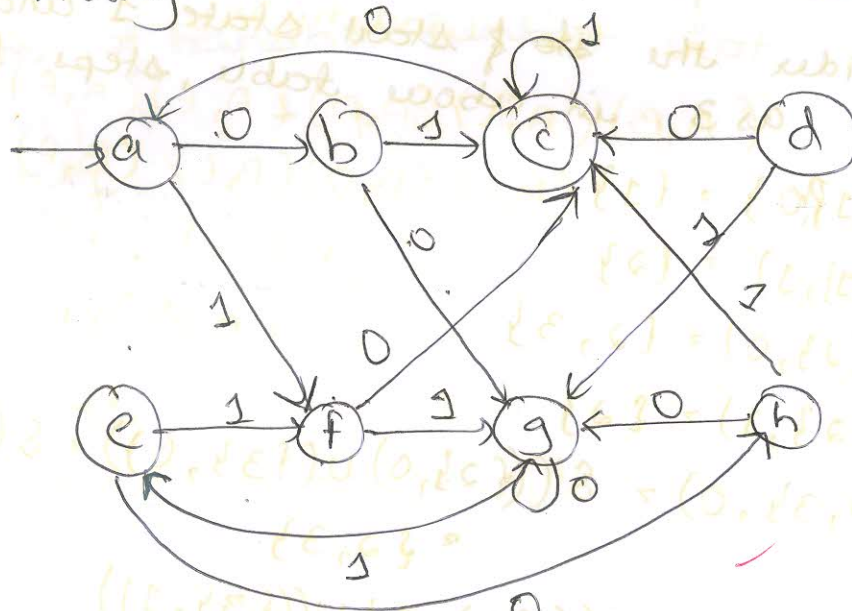
$$\rightarrow \delta(\{2, 3\}, 0) = \delta((\{2\}, 0) \cup (\{3\}, 0)) = \delta(\{2, 3\} \cup \{3\}) = \{2, 3\}$$

$$\begin{aligned} \delta(\{2, 3\}, 1) &= \delta((\{2\}, 1) \cup (\{3\}, 1)) \\ &= \delta(\{2\} \cup \{2, 3\}) \\ &= \{2, 3\} \end{aligned}$$

$\rightarrow \delta(\{34, 0\} = \{34\}$
 $\delta(\{34, 1\} = \{2, 34\}$



4B) Table Filling Method



Transition table

δ	0	1
$\rightarrow a$	b	f
b	g	c
c*	a	c
d	c	g
e	h	f
f	c	g
g	g	e
h	g	c

Table filled is
let's take the conditions:

(A, B) (A, C) (A, D) (A, E) (A, F) (A, G) (A, H)
 (B, C) (B, D) (B, E) (B, F) (B, G) (B, H)
 (C, D) (C, E) (C, F) (C, G) (C, H)
 (D, E) (D, F) (D, G) (D, H)
 (E, F) (E, G) (E, H)
 (F, G) (F, H)
 (G, H)

Here final state is c and
start state is a

Table

8.

B	X₁							
C	X ₀	X ₀						
D	X ₁	X ₁	X ₀					
E	X₂	X ₁	X ₀	X ₁				
F	X ₁	X ₁	X ₀	X	X ₁			
G	X ₂	X ₁	X ₀	X ₁	X ₂	X ₁		
H	X ₁	X	X ₀	X ₁	X ₁	X ₁	X ₁	
	A	B	C	D	E	F	G	

Possible conditions are:

$$\begin{aligned}
 (f,c) &\xleftarrow{1} (A,B) \xrightarrow{0} (b,g) \quad \times \\
 (f,g) &\xleftarrow{1} (A,D) \xrightarrow{0} (b,c) \quad \times \\
 (f,f) &\xleftarrow{1} (A,E) \xrightarrow{0} (b,h) \\
 (f,g) &\xleftarrow{1} (A,F) \xrightarrow{0} (b,c) \quad \times \\
 (f,e) &\xleftarrow{1} (A,G) \xrightarrow{0} (b,g) \\
 (f,c) &\xleftarrow{1} (A,H) \xrightarrow{0} (b,g) \quad \times \\
 (c,g) &\xleftarrow{1} (B,D) \xrightarrow{0} (g,c) \quad \times \\
 (c,f) &\xleftarrow{1} (B,E) \xrightarrow{0} (g,h) \quad \times \\
 (g,g) &\xleftarrow{1} (B,F) \xrightarrow{0} (g,c) \quad \times \\
 (c,e) &\xleftarrow{1} (B,G) \xrightarrow{0} (g,g) \quad \times \\
 (c,c) &\xleftarrow{1} (B,H) \xrightarrow{0} (g,g)
 \end{aligned}$$

$$(g, f) \xleftarrow{1} (D, E) \xrightarrow{0} (c, h) \times$$

$$(g, g) \xleftarrow{1} (D, F) \xrightarrow{0} (c, c)$$

$$(g, e) \xleftarrow{1} (D, G) \xrightarrow{0} (c, g) \times$$

$$(g, c) \xleftarrow{1} (D, H) \xrightarrow{0} (c, g) \times$$

$$(f, g) \xleftarrow{1} (E, F) \xrightarrow{0} (h, c) \times$$

$$(f, e) \xleftarrow{1} (E, G) \xrightarrow{0} (h, g)$$

$$(f, c) \xleftarrow{1} (E, H) \xrightarrow{0} (h, g) \times$$

$$(g, e) \xleftarrow{1} (F, G) \xrightarrow{0} (c, g) \times$$

$$(g, c) \xleftarrow{1} (F, H) \xrightarrow{0} (c, g) \times$$

$$(e, c) \xleftarrow{1} (G, H) \xrightarrow{0} (g, g) \times$$

- Removing all the states ^{from above} which have 'c' as c is a final state like $(A, c) (B, c) (C, c) (D, c) (E, c) (F, c) (G, c) (H, c)$

- In table fill with x_1 like $(f, c) (b, c)$:
whosoever have c is like $(f, c) (b, c)$:
So strings with x_1 will be $(A, D) (A, F) (A, H)$
 $(A, B) (B, D) (B, E) (B, F) (D, E) (D, G)$
 $(D, H) (E, F) (E, G) (E, H) (F, G) (F, H)$
 (B, G)

- left with we will keep it blank where the string is set is $(B, H) (B, G) (A, E) (A, F) (G, H)$

$$(f, c) \xleftarrow{1} (A, B) \xrightarrow{0} (b, g)$$

$$(f, e) \xleftarrow{1} (A, G) \xrightarrow{0} (b, g)$$

$$(f, c) \xleftarrow{1} (E, G) \xrightarrow{0} (h, g)$$

$$(f, f) \xleftarrow{1} (A, E) \xrightarrow{0} (b, h)$$

All these will be filled with x_2

4 C) Applications of Finite State Automata

1) Design of Digital Circuits

FA is used for checking and verifying the digital circuits using software. ~~The design of FA software is made to check~~
~~The design of FA is used in hardware to~~

2) Compiler Construction

In the large text of natural language ~~an~~ phrases input text is broken into multiple units such as keywords, words and phrase.

3) String Matching

It is to identify^{ing} the string in a large natural language text. It is to match the string word, keyword ~~pro~~ from the large text.

4) Software Design

FA is used to design the software considering all the properties.

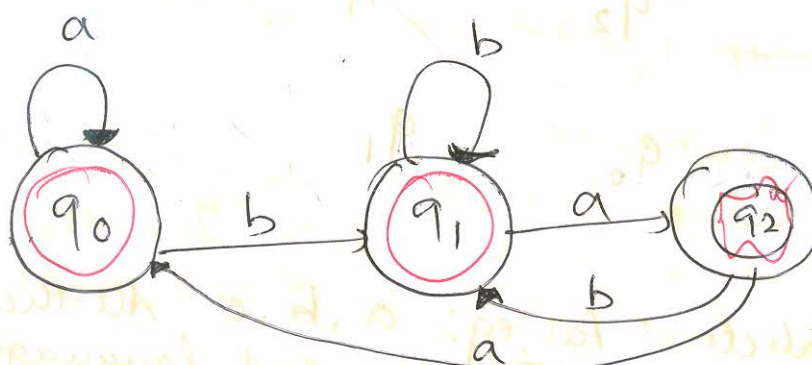
5) Other Applications

Other application used by FA is in Artificial Intelligence, knowledge engineering and game theory, games, graphics, linguistics etc.

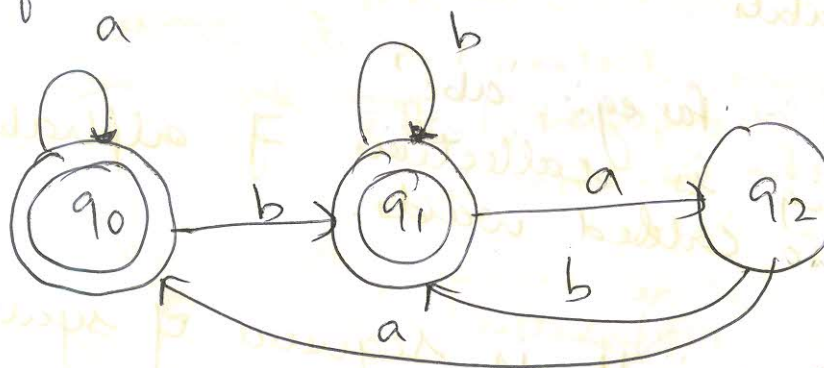
6) String Processing is also major application of finite Automata

PART - A

1A) Language $L = \{w \in \{a,b\}^* : w \text{ does not end in } ba\}$
 first consider from q_0 which ends with
 for that $L = \{ba, baba, ababa, bbbaba, \dots\}$



Now for w which does not end in ba



$$M = \{Q, \Sigma, \delta, q_0, F\}$$

$$Q = \{q_0, q_1, q_2\}$$

$$F = \{q_0, q_1\}$$

$$\delta = \{a, b\}$$

$$q_0 = \{q_0, q_1, q_2\}$$

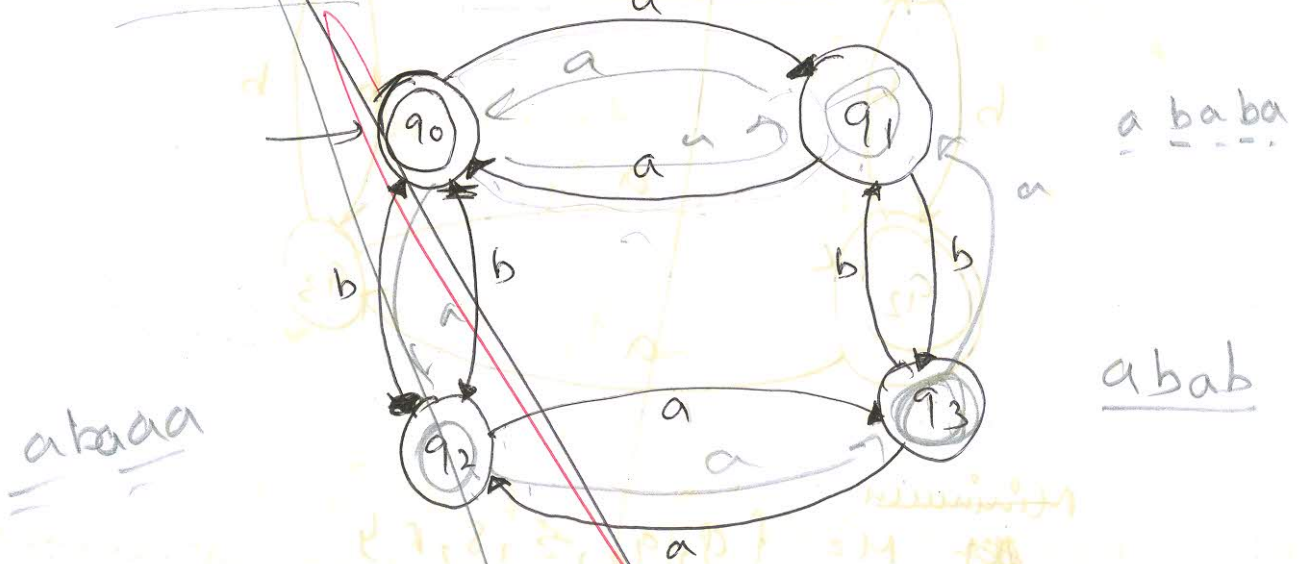
Transition Table

δ	a	b
q_0^*	q_0	q_1
q_1^*	q_2	q_1
q_2	q_0	q_1

- 1B) i) Alphabets : for eg: a, b, c. All these are alphabets of a strings and language consist of strings which in turn consist of alphabets.
- ii) Strings : for eg: 'abc'. String is collection of alphabets and also called words.
- iv) Language : It is sequence of symbols which consist of strings.

1c)

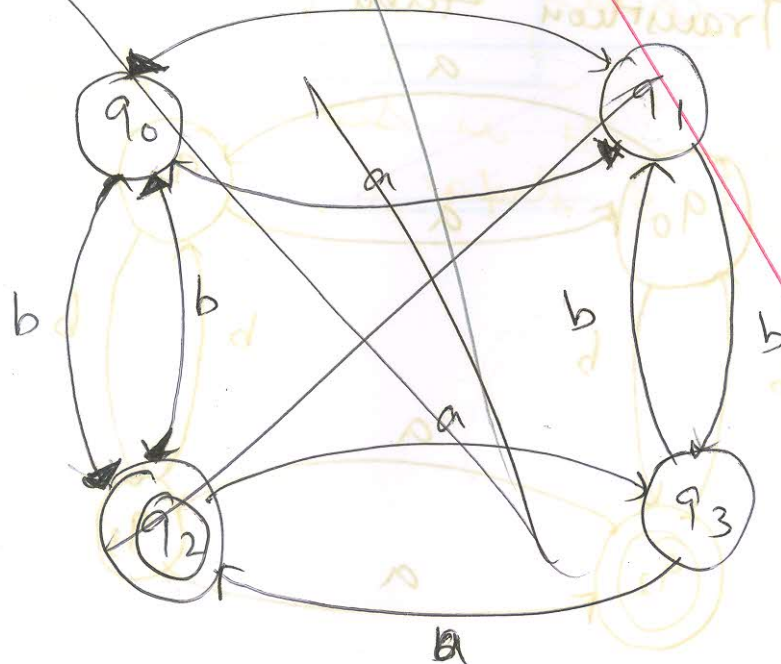
DFSM for language $L = \{w \in \{a,b\}^* : w \text{ contains even no's of a's and odd number of b's}\}$

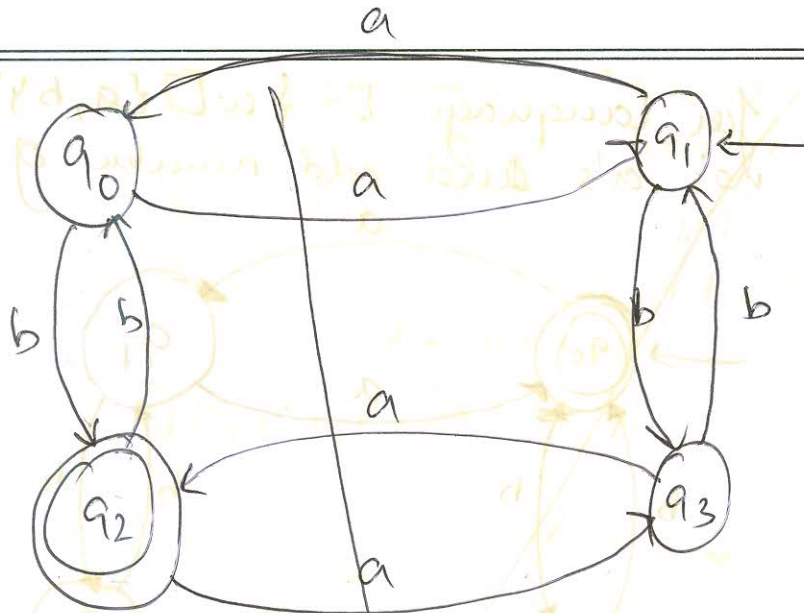


$Q = \{q, q_0, \epsilon, \delta, f\}$
when $Q = \{ \cdot \}$

1c)

DFSM for language $L = \{w \in \{a,b\}^* : w \text{ contains even no's of a's and odd number of b's}\}$





Minimum

$M = \{Q, \Sigma, \delta, q_0, F\}$

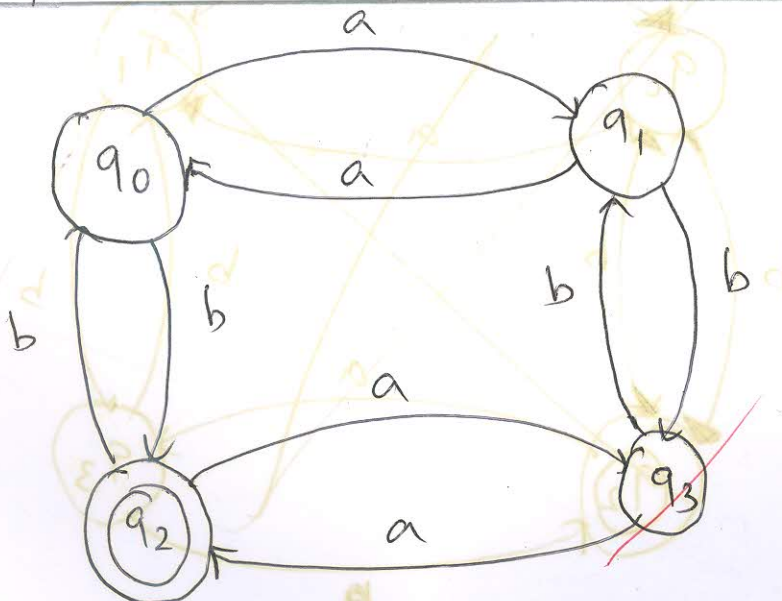
$Q = \{q_1\}$

$F = \{q_2\}$

$\Sigma = \{a, b\}$

$Q_0 = \{q_0, q_1, q_2, q_3\}$

Transition table



$L = \{w \in \{a,b\}^* ; w \text{ does not contain even no. of } a\text{'s and odd number of } b\text{'s}\}$

$M = \{Q, q_0, \epsilon, \delta, F\}$

~~$Q = \{q_0, q_1, q_2, q_3\}$~~ $F = \{q_2\}$

$\delta = \{a, b\}$

$q_0 = \{q_0, q_1, q_2, q_3\}$

Transition Table

δ	a	b
q_0	q_1	q_2
q_1	q_0	q_3
q_2	q_3	q_0
q_3	q_2	q_1

47/50 km

IInd Internals

- 1 B. Context free languages ✓
- 2 A. Regular ✓
- 3 B. Type 2 ✓
- 4 D. 0 ✓
- 5 Homomorphism A. ✓
- 6 B. Context free languages ✓
- 7 B. Derivation ✓
- 8 D. Both A & B ✓
- 9 D. All of the mentioned ✓
- 10 A. Ambiguous ✓

04
10

10

Part-A

1A) i) All strings in the language $L: \{a^u b^m a^{2u} \mid u, m \geq 0\}$
 $M = \{V, T, P, S\}$

Where V is the variable
 T is the terminal
 P is the production
 S is the start symbol

$$S \rightarrow a S a a \mid A$$

$$A \rightarrow b \mid \epsilon$$

$$M = \{V, T, P, S\}$$

$$V = \{A\} \quad T = \{a, b\}$$

$$T = \{a, b\}$$

$$P = S \rightarrow a S a a \mid A$$

$$A \rightarrow b \mid \epsilon$$

$$S = \{a\}$$

ii) All strings with more a's and b's than b's

i)

All palindromes

$$L = \{ aba, abba, baab, \dots \}$$

$$S \rightarrow asa / bsb / a / b / \epsilon$$

$$M = \{ V, T, P, S \}$$

$$V = \{ a, b \}$$

$$T = \{ a, b \}$$

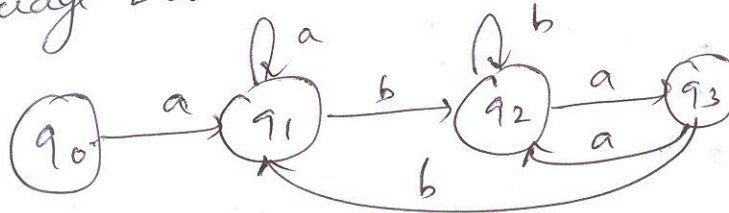
$$P = S \rightarrow asa / bsb / a / b / \epsilon$$

$$S = \{ a, b \}$$

ii)

All nonempty strings that start and end with same symbol

The language L will be $= \{ aba, bab, abba, baaab \dots \}$



$$S \rightarrow aSa / bsb / a / b / \epsilon$$

$$M = \{ V, T, P, S \}$$

$$V = \{ a, b \}$$

$$T = \{ a, b \}$$

$$P = S \rightarrow aSa / bsb / a / b / \epsilon$$

$$S = \{ a, b \}$$

3B i) Context free grammar $M = \{V, T, P, S\}$

An ~~language~~ grammar is said to be a context free if it satisfies

3B i) Context free grammar

H is the $M = \{V, T, P, S\}$ state machine where
V is the variable, T is the terminal, P is the
production and S is the start symbol.

ii) Leftmost and Rightmost derivation
In this derivation the given string is satisfied the
problem is satisfied by putting the conditions
given. For eg:-

We have to bring the string ~~aba~~ and given conditions are

$$S \rightarrow aSA / \epsilon$$

$$A \rightarrow ba$$

then here leftmost derivation will be

$$S \rightarrow aSA$$

$$\cancel{S \rightarrow aSba} \quad S \rightarrow a\epsilon A$$

$$\cancel{S \rightarrow a\epsilon} \quad S \rightarrow aA$$

$$S \rightarrow aba$$



then here rightmost derivation will be

$$S \rightarrow aSA$$

$$S \rightarrow aSba$$

$$S \rightarrow a\epsilon ba$$

$$S \rightarrow aba$$

20. Ambiguous = It means for ~~both~~ a grammar there should be two distinct trees not similar.

Given:- $E \rightarrow E + E$

$E \rightarrow E^* E$

$E \rightarrow (E) | id$

Condition :- $id + id * id$
for leftmost

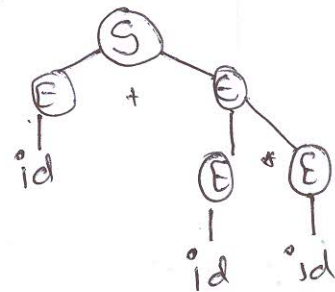
$S \rightarrow E + E$

$S \rightarrow id + E$

$S \rightarrow id + E^* E$

$S \rightarrow id + id^* E$

$S \rightarrow id + id^* id$



Now if the given taken as $S \rightarrow E^* E$

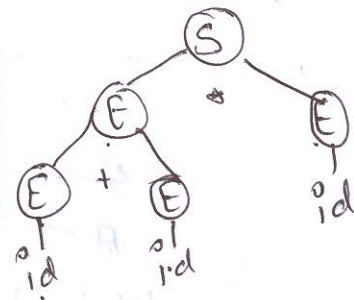
$S \rightarrow E^* E$

$S \rightarrow E + E^* E$

$S \rightarrow id + E^* E$

$S \rightarrow id + id^* E$

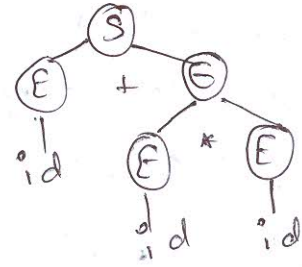
$S \rightarrow id + id^* id$



Hence the parse trees are different for both the situations
hence, it is ambiguous

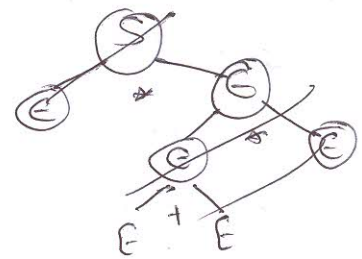
for rightmost derivation

$S \rightarrow E + E$
 $S \rightarrow E + E * E$
 $S \rightarrow E + E * id$
 $S \rightarrow E + id * id$
 $S \rightarrow id + id * id$

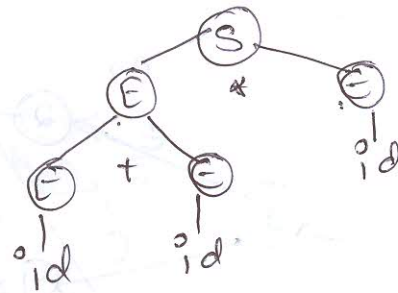


for condition

$S \rightarrow E * E$
 $S \rightarrow E * E * E$
 $S \rightarrow E * E + E * E$
 $S \rightarrow E * E + E * E$
 $S \rightarrow E + E * E$
 $S \rightarrow E + E * id$
 $S \rightarrow E + id * id$
 $S \rightarrow id + id * id$



$S \rightarrow E * id$
 $S \rightarrow E + E * id$
 $S \rightarrow E + id * id$
 $S \rightarrow id + id * id$



hence the parse tree is different for both the conditions so it is ambiguous.

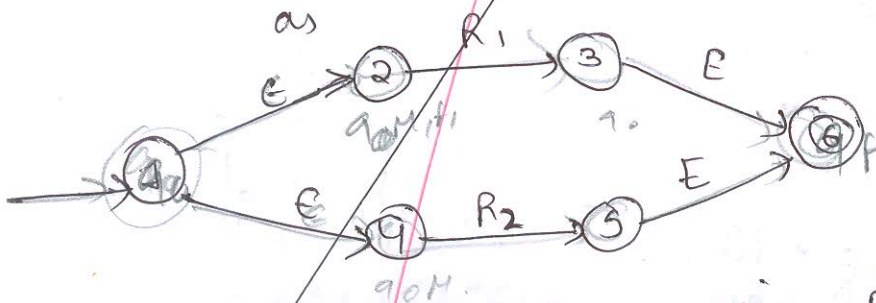
Part - B.

3A)

Meaning: Any language that can be defined with a regular expression can be accepted by some FSM and so is regular.

Proof: There are regular expressions R_1 and R_2 and they are regular. Regular expression R_1 and R_2 are defined by the language $L(R_1)$ and $L(R_2)$ which is accepted by FSM and it is regular. Next can be represented by the following given below.

Case 1:- if the regular expression is $R_1 + R_2$ that is in $L(R_1 + R_2)$ then it will be represented as



Here R_1 and R_2 are in parallel.
 M_1 M_2

Case 2:- if the regular expression is in $R_1 \cdot R_2$ form. then it will be represented as

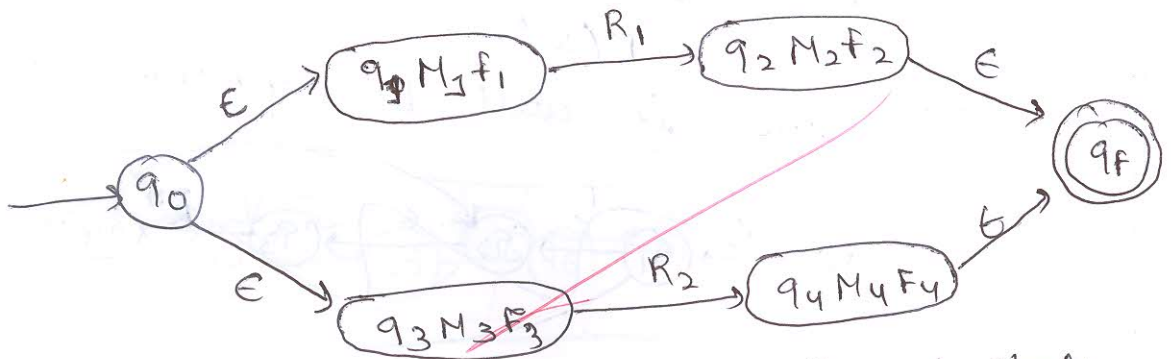
Part-B

3A) Theorem: Any language that can be defined with a regular expression can be accepted by some FSM and so is regular.

Proof: Consider the regular expression R_1 and R_2 which can exist in different forms like $R_1 + R_2$, $R_1 \cdot R_2$ and R^* which are accepted by FSM and is regular. These languages can be represented by given below

Case 1:-

If the regular expression R_1 and R_2 are in the form $L(R_1 + R_2)$ then it will be represented as:-



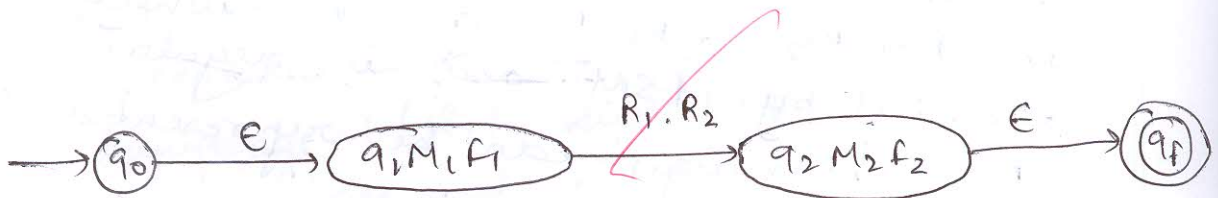
Here in above q_0 is the initial state

q_f is the final state

R_1 and $R_2 = R_1 + R_2$ are shown in parallel.

Case 2:-

If the regular expression R_1 & R_2 are in the form $R_1 \cdot R_2$ then it will be represented as.

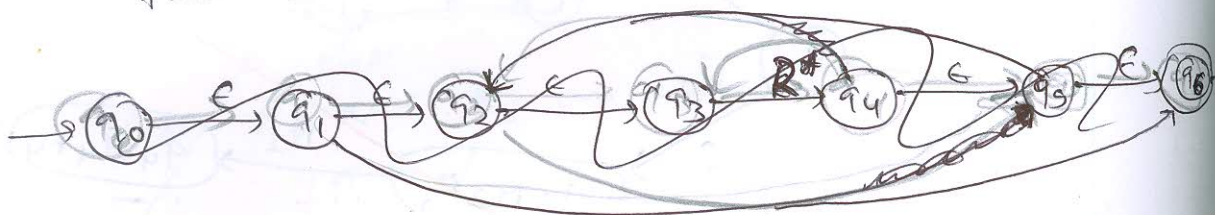


Here $R_1 \cdot R_2$ are in series.
where q_0 is the initial state
 q_f is the final state

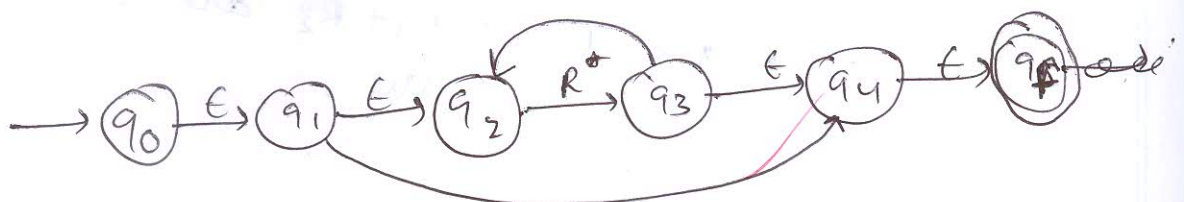
Q.

Case 3:-

If the regular expression is R^+ that is $[R^+]$ then it will be represented as

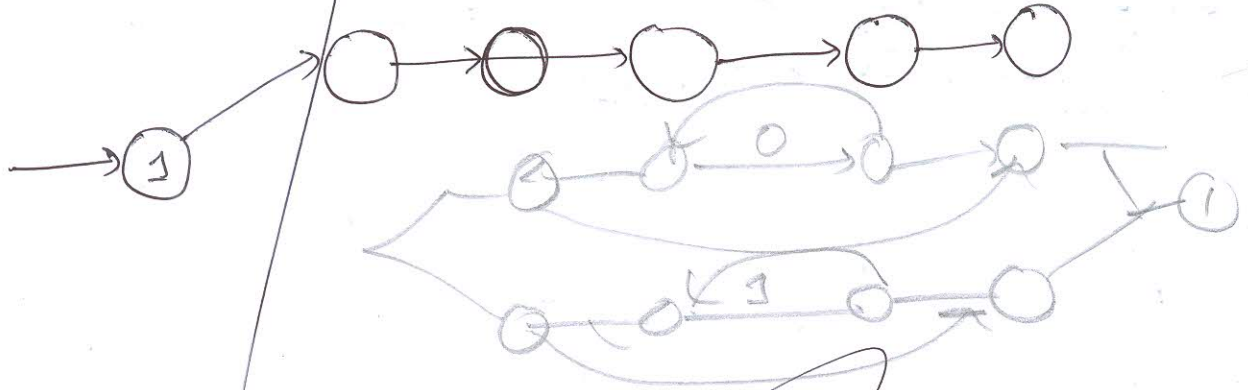


Here q_0 is the initial state
 q_f is the final state.



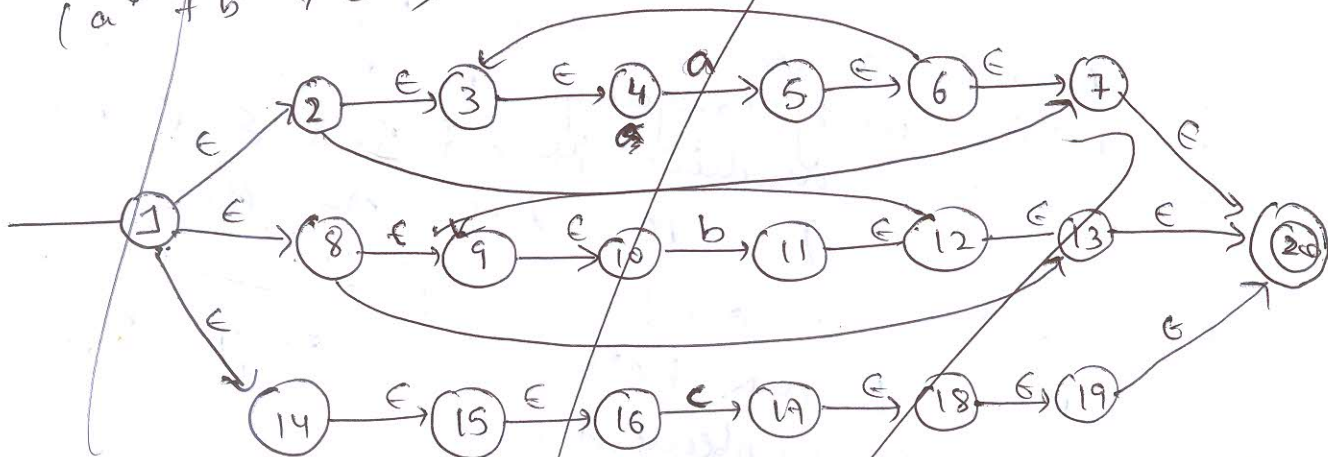
3B)

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

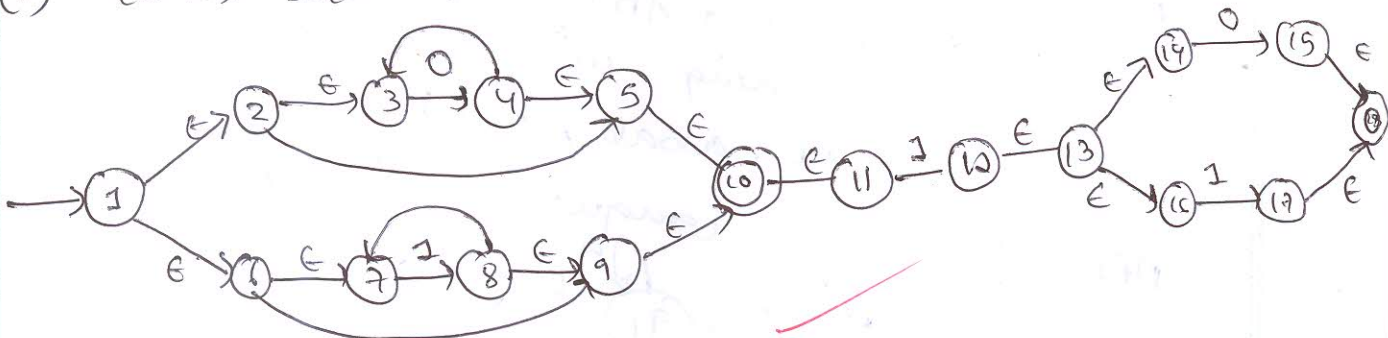


ii)

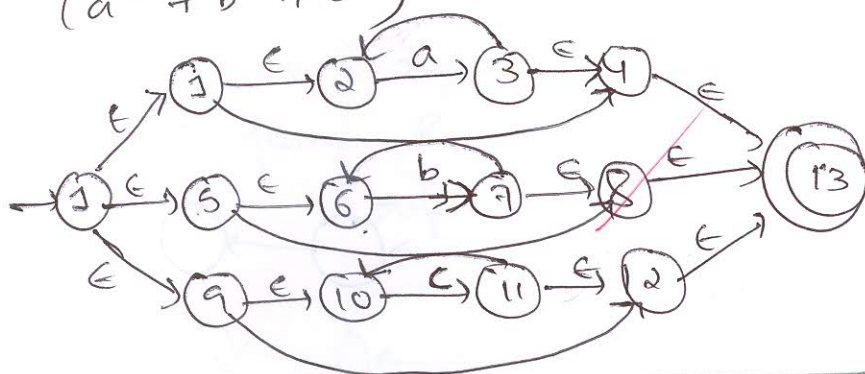
$(a^* + b^* + c^*)$



3B) (i) $(0+1)^* 1 (0+1)$



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



3c) Theorem :- Regular set are closed under complementation

Proof :- The languages which are context free languages and are associated by DFA must be regular complement.

Let is a state machine $M_1 = \{ Q, \delta, \epsilon, f \}$ which is of language L and then there is a state machine $M_2 = \{ Q, \delta, \epsilon, Q-f \}$ for language $\bar{L}$. Here both the M_1 and M_2 are same except the final states. Hence L and $\bar{L}$ are both same only difference is of complement.

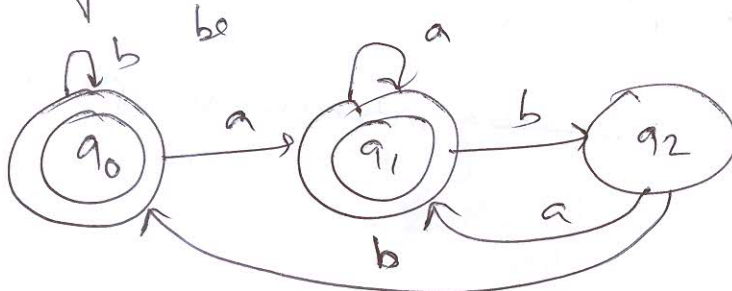
for eg:- There is a strings of a's and b's ending with ab

$L = \{ ab, abab, aabbb, \dots \}$

DFA for this language will be



But for the complement of it will



then only the difference is of final state.

HH
57

on

ev

Quiz

- 1 D. Trackable problem ✓
- 2 A. ~~Trackable~~ problem undecide intractable problem ✓
- 3 D. 4 ✓
- 4 D. none of these ✓
- 5 C. Backus Normal form ✓
- 6 e. both (a) & (b) ✓
- 7 e. recursion ✓
- 8 D. None of these ✓
- 9 D. non-deterministic ✓
- 10 A. True ✓

01
10
Im

Part A

1 A.

Given:-
Context free languages are closed under Union, Reversal and Concatenation.

Context free language can be defined as

$$\{V, \Sigma, R, S\}$$

where V are the variable in the language
 Σ is the transition all the productions
 R is the
 S is the start state

i)

To prove that Context free languages are closed under Union.

Consider two grammars G_1 and G_2 such that

$$L(G_1) \cup L(G_2)$$

To prove this we have $\{V, \Sigma, R, S\}$
which can be seen as

Consider two grammars G_1 and G_2

$$G_1 = \{V_1, \Sigma_1, R_1, S_1\}$$

$$G_2 = \{V_2, \Sigma_2, R_2, S_2\}$$

such that $V = V_1 \cup V_2 \cup \{S\}$

$$\Sigma = \Sigma_1 \cup \Sigma_2$$

$$R = R_1 \cup R_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\}$$

$$S = \text{start state}$$

hence the considered grammars are in union.

(ii) Context free language closed Under Revers
 Consider grammar $X \rightarrow ABC$
 $X \rightarrow a$

So to prove $L(G) = L(G)^R$
 where G is the grammar and G^R is the
 reverse of language of G .

- $X \rightarrow BC$ and reverse will be $X \rightarrow CB$
- $X \rightarrow a$ and reverse will be $X \rightarrow a$

Hence the language is closed under Revers

(iii) Context free language closed under Concatenation
 Consider the language

$L(G_1) \cup L(G_2)$ which is proved
 such that G_1 and G_2 are the two grammars
 $\{V_1, \Sigma_1, R_1, S_1\}$ and $\{V_2, \Sigma_2, R_2, S_2\}$

and $V = V_1 \cup V_2 \cup \{S\}$

$\Sigma = \Sigma_1 \cup \Sigma_2$

$R = R_1 \cup R_2 \cup \{S \rightarrow S_1 S_2\}$

$S = \text{start state}$

So the grammar will be $L(G_1, G_2)$ hence
 it is proved that ~~concatenation~~ language
 closed under concatenation is context
 free.

26.

Left Recursion :: If the grammar is such that it consist of non terminal that it is in left Recursion.

$$A \rightarrow Aa$$

where a is non terminal

To eliminate the left Recursion

1) we have to see that there should be no non terminal variable in the language if it is then the left recursion will not be applicable.

2) then we have to consider the

$$A \rightarrow \alpha_i A \mid \beta_i$$

We have compare the given grammar with above given sentence.

then put in the form

$$A \rightarrow \alpha_i \beta_i A' \mid \epsilon$$

$$A' \rightarrow \alpha_i A' \mid \epsilon$$

Consider the grammar

$$E \rightarrow E + T$$

$$E \rightarrow T * F$$

$$E \rightarrow E \mid id$$

grammar

$$E \rightarrow E + T \text{ where}$$

27.

PDA for the following grammar.

$$\Gamma = \{ \epsilon, q_0, F, \delta, T, P \}$$

Grammar given is

$$S \rightarrow aABC$$

$$A \rightarrow aB/a$$

$$B \rightarrow bA/b$$

$$C \rightarrow a$$

Consider the formula $\delta(q_1, a, A) = (q_1, \epsilon)$
when a is the terminal

Production	Transition
$S \rightarrow aABC$	$\delta(q_1, a, S) = (q_1, ABC)$
$A \rightarrow aB$	$\delta(q_1, a, A) = (q_1, B)$
$A \rightarrow a$	$\delta(q_1, a, A) = (q_1, \epsilon)$
$B \rightarrow bA$	$\delta(q_1, b, B) = (q_1, A)$
$B \rightarrow b$	$\delta(q_1, b, B) = (q_1, \epsilon)$
$C \rightarrow a$	$\delta(q_1, a, C) = (q_1, \epsilon)$

Here ~~the~~ start state is ~~q_0~~ $= \{q_0\}$
and ~~final state is~~

Now we have the grammar $S \rightarrow aABC$
which should be converted into all
nonterminals.

$S \rightarrow aABC$
 $S \rightarrow a aBBC$
 $S \rightarrow a a bBC$
 $S \rightarrow a a b bC$
 $S \rightarrow a a b b a$

we have grammar known as $aabba$

Now we have the grammar.

$(q_0, \epsilon, z_0) = (q_1, S z_0)$

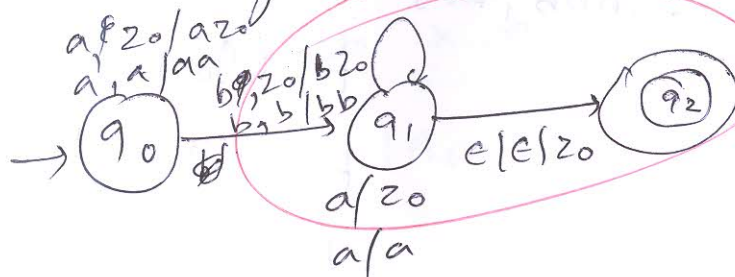
$$\begin{aligned}
 & (q_0, aabba, z_0) \vdash (q_1, \cancel{a}abba, \cancel{z_0}) \\
 & \vdash (q_1, \cancel{a}abba, AB(z_0)) \\
 & \vdash (q_1, \cancel{a}abba, BB(z_0)) \\
 & \vdash (q_1, \cancel{a}abba, BCB(z_0)) \\
 & \vdash (q_1, \cancel{a}abba, \cancel{z_0}) \\
 & \vdash (q_1, \cancel{a}abba, z_0) \\
 & \vdash (q_1, \cancel{a}abba, z_0)
 \end{aligned}$$

Part - B

3A) $L = \{a^n b^n \mid n \geq 1\}$
for converting into PDA we have language
say $L = \{ab, aabb, aaabbb, \dots\}$

$$\begin{aligned}
 \text{So } & \cancel{q_0}. S(q_0, a, a) = S(q_0, aa) \\
 & S(q_0, \cancel{a}, z_0) = S(q_0, az_0) \\
 & S(\cancel{q_1}, b, z_0) = S(q_1, bz_0) \\
 & S(\cancel{q_1}, b, b) = S(q_1, bb)
 \end{aligned}$$

Transition Diagram will be



Here $Q = \{q_0, T, F, S, \Gamma, \epsilon, Z_0\}$

q_0 is the start state $= \{q_0\}$

T is the terminal variable $= \{a, b\}$

Q is the ~~set~~ $\{q_0, q_1, q_2\}$

F is the final state $\{q_2\}$

Z_0 = it is initial symbol in the stack

$\Gamma = \{a, b\}$

for the accepting string we have

$\{q_0, aabbbb, Z_0\} \vdash \{q_0, aabbbb, aZ_0\}$
 $\vdash \{q_0, abbbb, aaZ_0\}$
 $\vdash \{q_0, bbb, aaaZ_0\}$
 $\vdash \{q_0, bbb, aaaZ_0\}$
 $\vdash \{q_1, bb, aaZ_0\}$
 $\vdash \{q_1, b, aZ_0\}$
 $\vdash \{q_1, \epsilon, Z_0\}$

Rejecting state, consider

$\{q_0, aabbbb, Z_0\} \vdash \{q_0, abbbb, aZ_0\}$
 $\vdash \{q_0, bbb, aaaZ_0\}$
 $\vdash \{q_0, bb, aaZ_0\}$
 $\vdash \{q_0, b, aZ_0\}$

now it is ~~not~~ in rejecting state.

Eliminate the Left Recursion

$$\begin{aligned} S &\rightarrow A/B \\ A &\rightarrow ABC/Acd/a/aa \\ B &\rightarrow Bce/b \end{aligned}$$

grammar	formal all	New grammar
$S \rightarrow A/B$	Not applicable	$S \rightarrow A/B$
$A \rightarrow ABC/Acd/a/aa$	$A \rightarrow \alpha_i A / \beta_i$ $\alpha_1 = BC$ $\alpha_2 = cd$ $\beta = a/aa$ $A' \rightarrow A\alpha_1 / A\alpha_2 / \beta$	$A \rightarrow \beta_i A'$ $A' \rightarrow \alpha_i A' / E$ $A \rightarrow a/aa/A'$ $A' \rightarrow BC A' / E$ $A \rightarrow a/aa/A'$ is the new grammar
$B \rightarrow Bce/b$	$\alpha_1 = ce$ $\beta_1 = b$ $B \rightarrow B\alpha_1 / \beta_1$	$A' \rightarrow b A'$ $A' \rightarrow ce A' / E$

3B.

Turing Machine

$$L = \{a^n b^n \mid n \geq 1\}$$

Take the language $L = \{ab, aabb, aaabbb, \dots\}$
 which will be converted into x and y .

Consider language where $a^k b^k$ will be converted into x and y .

$$\begin{array}{c} x \alpha a a b b b \\ \uparrow \\ q_0 \end{array}$$

$$\begin{array}{c} x a a b b b \\ \uparrow \\ q_0 \end{array}$$

$$(q_0, a) = (q_1, x, R)$$

$$(q_1, a) = (q_1, a, R)$$

$$(q_2, y) = (q_2, y, R)$$

~~x~~ a a y b b
 $\uparrow$
 q_1
 x x a y b b
 $\uparrow$
 q_2
~~x x a y b b~~
 x x ~~x~~ y b b
 $\uparrow$
 q_2
 x x x y y b
 $\uparrow$
 q_3
 x x x y y y
 $\uparrow$
 q_1

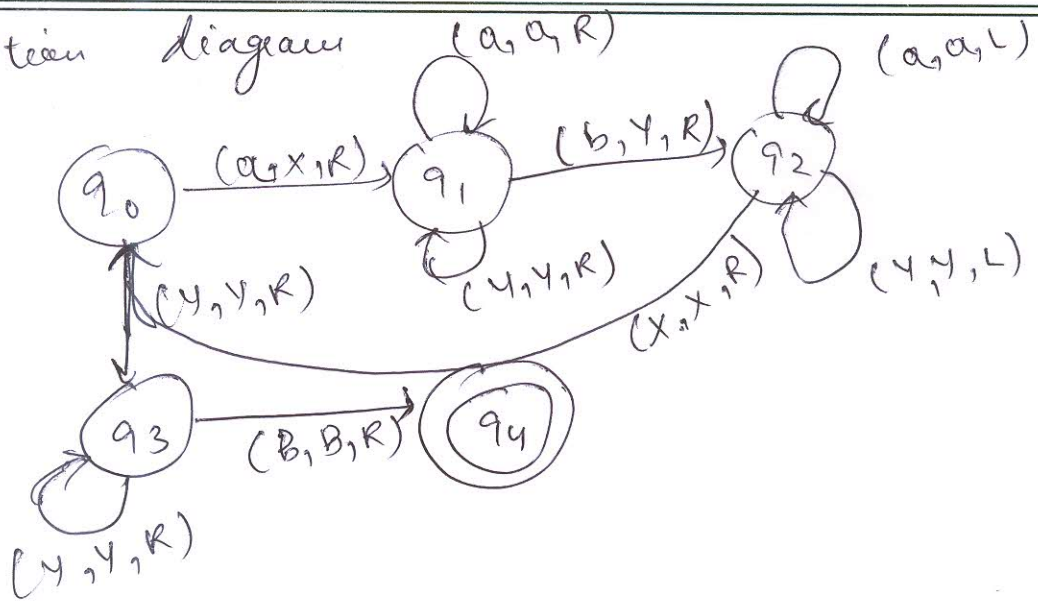
x x x y y y

$(q_2, \emptyset) = (q_2, L, Y)$
 $(q_2, X) = (q_0, R, X)$
 $(q_3, Y) = (q_3, R, Y)$
 $(q_2, a) = (q_2, L, a)$
 $(q_0, y) = (q_3, R, y)$
 $(q_1, y) = (q_1, R, y)$
 ~~$(q_2, y) = (q_2, L, y)$~~
 $(q_4, B) = (q_4, R, B)$

Transition Table
Present state

	a	b	x	y	B
Start state q_0	xRq_1	—	—	yRq_3	—
q_1	aRq_1	yLq_2	—	yRq_1	—
q_2	aLq_2	—	xRq_0	yLq_2	—
q_3	—	—	—	yRq_3	—
q_4 final state	—	—	—	$\emptyset$	BRq_4

Transition diagram



~~Transitions~~

⊕

77
50

Q