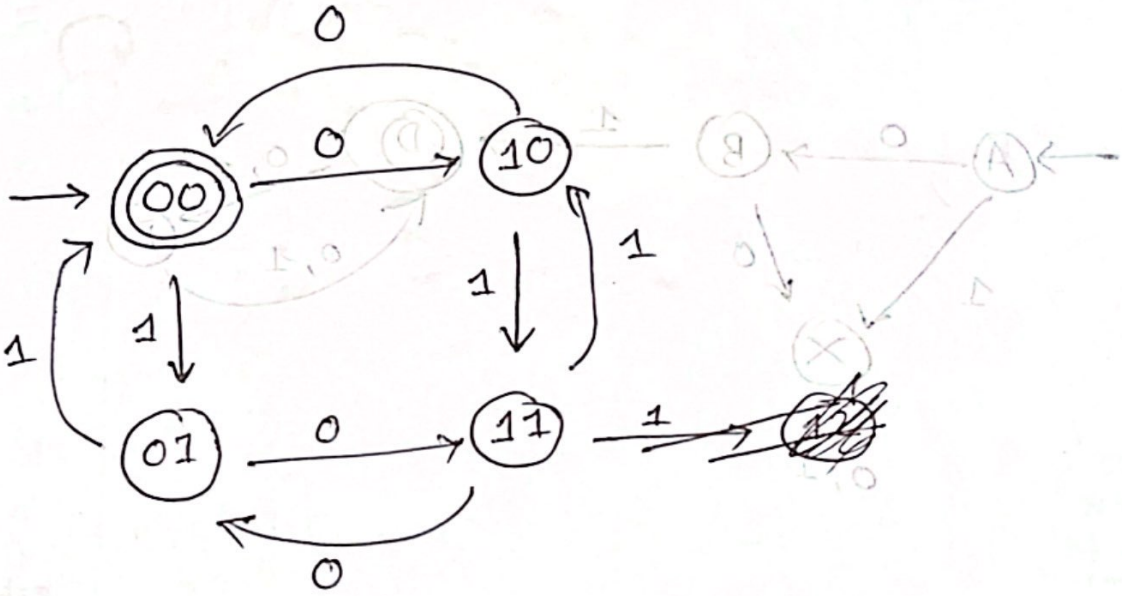
Date...../ ...../ 20 .....

1b

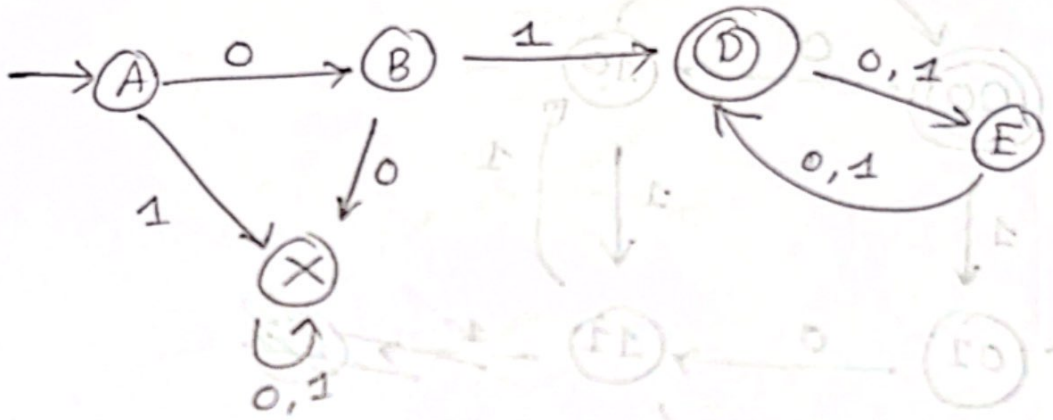
(i) $L = \{ w \mid w$ has both an even number of 0 & even number of 1 $\}$.



Provati

Date...../ ...../ 20 .....

(ii) $L = \{w \mid w$ is of even length & begins with $01\}$.

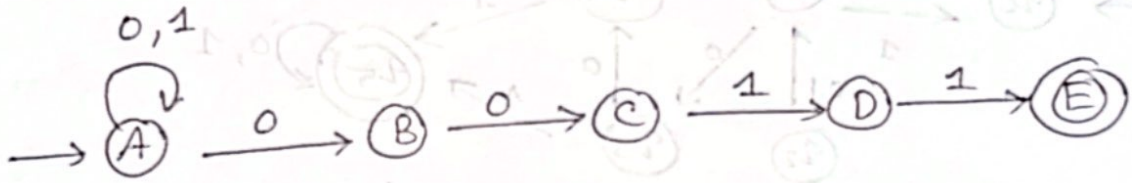


Provati

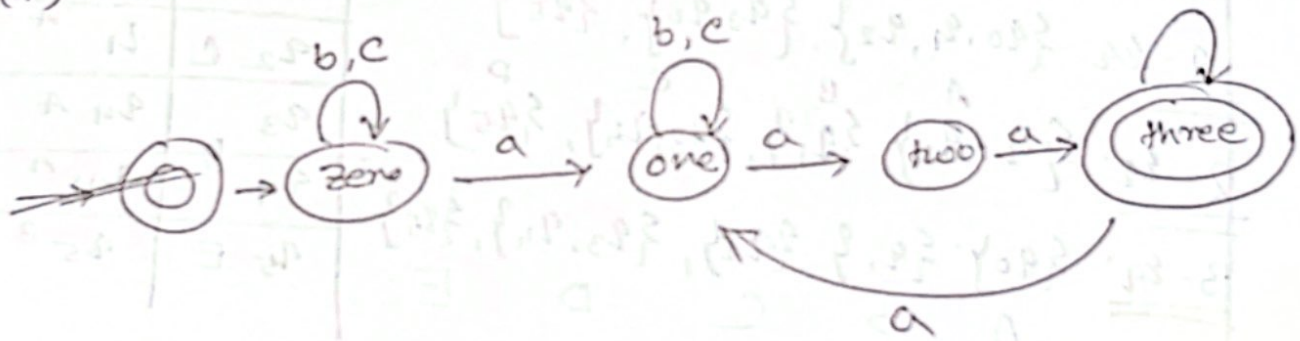
Date...../ ...../ 20 .....

1c

(i) ends with 0011



(ii) number of occurence of a is multiple of 3.



Provati

Theory of Computation

Date 18 / 10 / 2025

2) Prove that for any transition function ~~for~~ δ for any input string x and y :-

$$\delta(q, xy) = \delta(\delta(q,x), y) \quad \text{——— (1)}$$

Proof. By method of induction on $|y|$, i.e length of y

Basis: Let $|y| = 1$, i.e $y = a \in \Sigma$

From 1: LHS of 1 $= \delta(q, xa)$.

$= \delta(\delta(q,x), a)$ From property 2

$=$ RHS of 1.

Assume result, i.e (1) for all string x and all string y where length of y, i.e $|y| = n$. Let length of y be $n+1$.

Write $y = y_1 a$ where $|y_1| = n$

From LHS of 1 $= \delta(q, xy_1 a)$.

$= \delta(q, x_1 a)$, where $x_1 = xy_1$

$= \delta(q, \delta(\delta(q, x_1), a))$ by property-2

$= \delta(\delta(q, xy_1), a)$

$= \delta(\delta(\delta(q, x), y_1), a)$ By inductive Hypothesis

RHS of 1 $= \delta(\delta(q, x), y_1 a)$

$= \delta(\delta(\delta(q, x), y_1), a)$ By property-2

Hence LHS = RHS This proves 1 for string of y of length $n+1$. By principle of induction 1 is true for all string

2B → Lecture 2 → Slide - 12

Date...../...../20.....

2C

If $D = \{Q_D, \Sigma, \delta_D, \{q_0\}, F_D\}$ is a DFA constructed of $N = \{Q_N, \Sigma, \delta_N, q_0, F_N\}$, a NFA by subset construction method, then we need to show that $L(D) = L(N)$.

Proof: What we actually proof, by induction on $|w|$ that,

$$\hat{\delta}_D(\{q_0\}, w) = \hat{\delta}_N(q_0, w)$$

But the δ function, $\hat{\delta}_D$ interprets it as a

Notice that the δ function returns a set from Q_N, where $\hat{\delta}_D$ interprets this as a ~~set~~ state of Q_D (which itself is a power set of Q_N), while $\hat{\delta}_N$ interprets as a subset of Q_N

Basis: When $|w| = 1$, i.e $w = \epsilon$ then by basis hypothesis of $\hat{\delta}$ function for DFA and NFA, $\hat{\delta}_D(\{q_0\}, w) = \delta_N(q_0, w) = q_0$

Induction: Let w be a string of length $n+1$, assume the statement for length n. Break w as $w = xa$ where a is final alphabet of w. By inductive hypothesis, $\hat{\delta}_D(\{q_0\}, x) = \hat{\delta}_N(q_0, x)$. Let both these sets of N's state be $\{P_0, P_1, P_2, \ldots P_k\}$

Provati

Scanned with CamScanner

Date...../...../20.....

Then by inductive hypothesis ~~on~~ for $\hat{\delta}$ for NFA,

$$\hat{\delta}_N(q_0, w) = \bigcup_{i=1}^{k} \delta_N(P_i, a) \quad \text{——— (i)}$$

and by subset construction method for DFA,

$$\hat{\delta}_D(\{P_1, P_2, P_3 \ldots P_k\}, a) = \bigcup_{i=1}^{k} \delta_N(P_i, a) \quad \text{——— (ii)}$$

Now, using (ii) and ~~$\hat{\delta}_D(\{P_1, P_2, P_3, \ldots P_k\}, x)$~~

$$\delta_D(\{q_0\}, x) = \{P_1, P_2, \ldots P_k\}$$

$$\therefore \hat{\delta}_D(\{q_0\}, wa) = \hat{\delta}_D(\hat{\delta}_D(\{q_0\}, x), a)$$

$$= \hat{\delta}_D(\{P_1, P_2, \ldots P_k\}, a)$$

$$= \bigcup_{i=1}^{k} \delta_N(P_i, a) \quad \text{——— (iii)}$$

From ~~(i)(ii)~~, (iii), $\hat{\delta}_D(\{q_0\}, w) = \delta_N(q_0, w)$.

We see that D & N both accept w, iff $\hat{\delta}_D(\{q_0\}, w)$ & or $\hat{\delta}_N(q_0, w)$ respectively contain a state in F_N, completing the proof $L(D) = L(N)$.

Provati

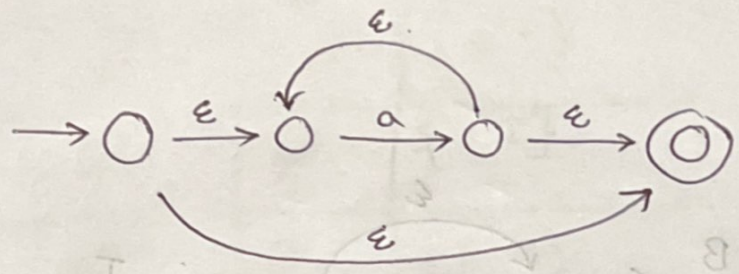
Scanned with CamScanner

36

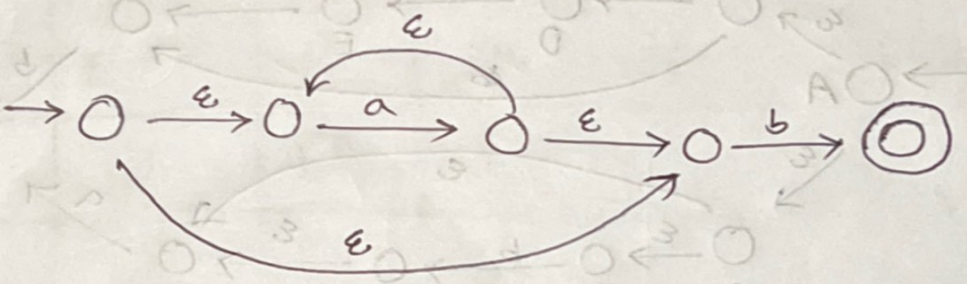
Date...../ ...../ 20 .....

Given R.E: $a^*b \mid b^*a$.

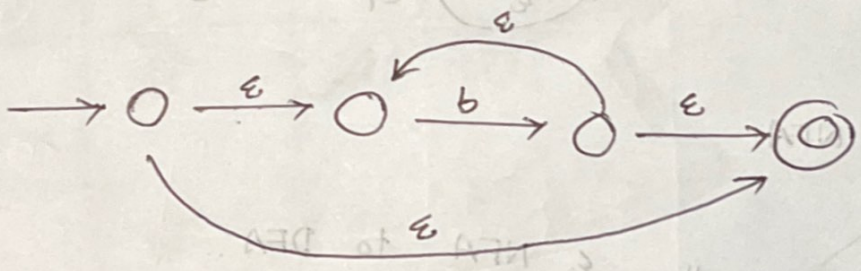
a^*:



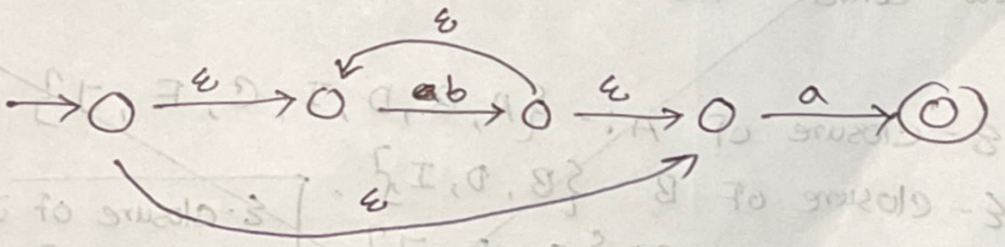
a^*b:



b^*:



b^*a:



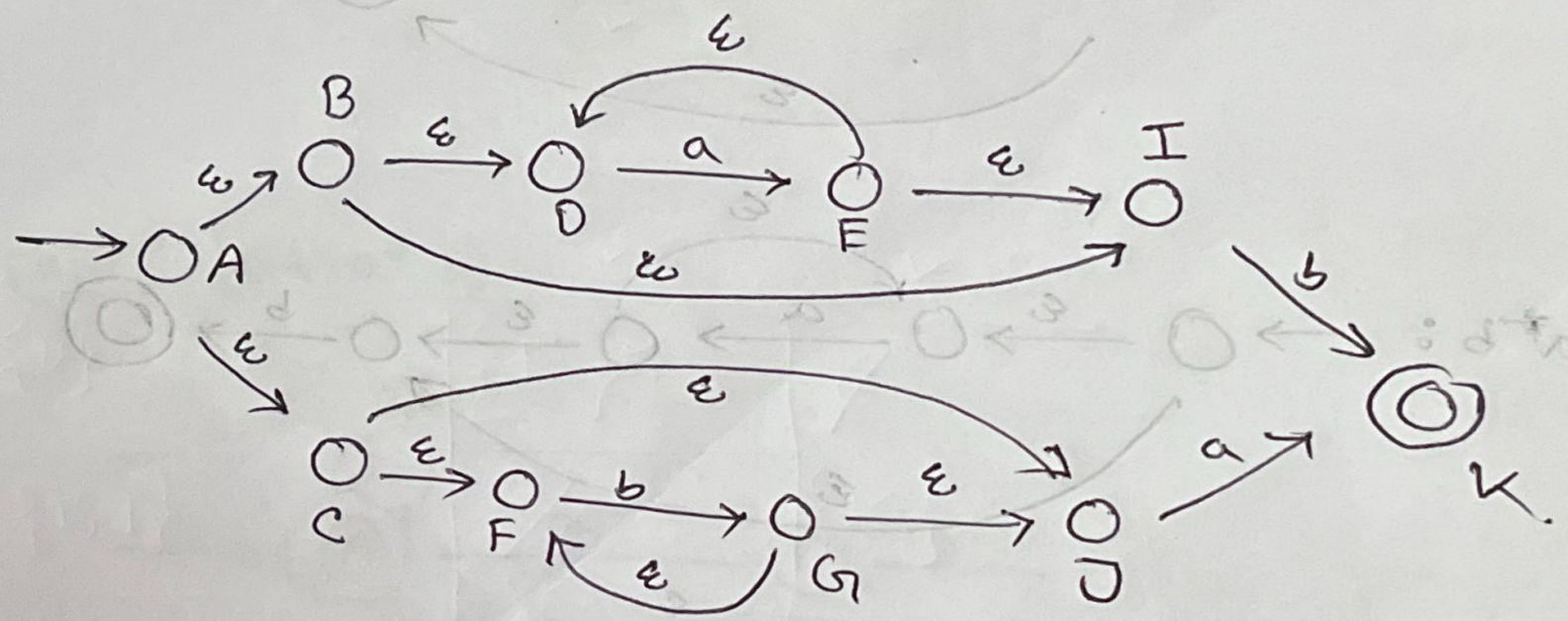
Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

Therefore;

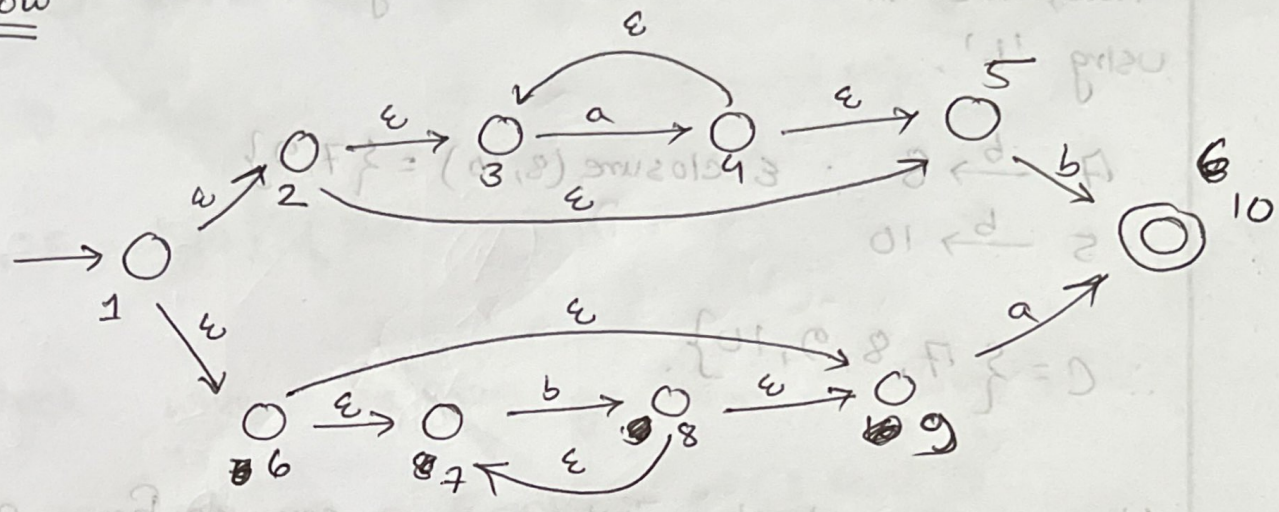
$a^*b \mid b^*a$:



3c

Date...../ ...../ 20 .....

Now



NF ε- closures:

~~A: 1. {1, 2, 3, 5, 6, 7,~~

A: {1, 2, 6, 3, 5, 7, 9}. [From '1' who accepts ε].

~~B: {4, 11}. [A থেকে কারা কারা a দিয়ে কোথায় যাওয়া যায়]~~

Now, we find where we go from A with 'a' input.

∴ 3 $\xrightarrow{a}$ 4

9 $\xrightarrow{a}$ 10 (final state.)

Now, ε-closure of (4, 10) = 3, 5 [10 has no ε-transition]

∴ ~~B = {3, 4, 9,~~ B = {3, 4, 5, 10}. [Final State Because of 10]

Provati

Date...../ ...../ 20 .....

Now, we find where we can go from A using 'b'.

$7 \xrightarrow{b} 8$ $\therefore$ ε-closure $(8, 10) = \{7, 9\}$.

$5 \xrightarrow{b} 10$

$\therefore C = \{7, 8, 9, 10\}$.

Now, we check where ~~to~~ can go from B using 'a'

$\therefore 3 \xrightarrow{a} 4$. $\therefore$ ε-closure $(4) = \{3, 5\}$.

$\therefore D = \{3, 4, 5\}$.

Check where we can go from B using 'b'

$5 \xrightarrow{b} 10$ ~~Final State~~

ε-closure of 10 : 10.

$\therefore E = \{10\}$ Final State.

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

Check where we can go from C using 'a':

$9 \xrightarrow{a} 10$. Eclosure $(10) = \{10\}$.

$\therefore F = \{10\}$. (Final State).

Check where we can go from C using 'b':

$7 \xrightarrow{b} 8$. E-closure of $8 = 7, 9$.

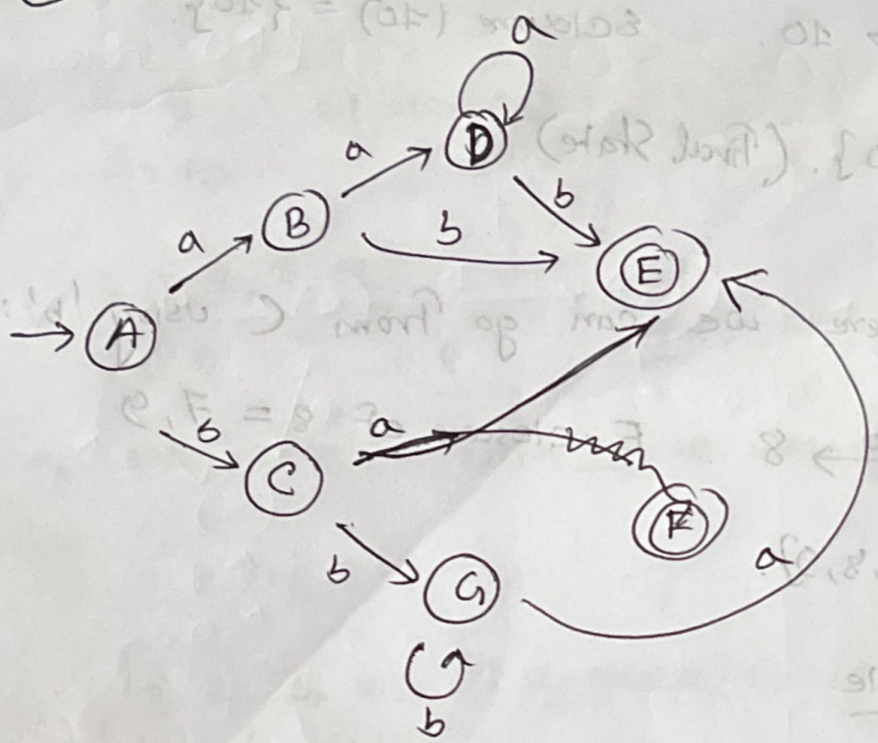
$\therefore G = \{7, 8, 9\}$.

$\therefore$ DFA Table:

State	a	b
A	B	C
B	D	E*
C	E*	G
D	D	E
E	$\emptyset$	$\emptyset$
~~F~~	~~$\emptyset$~~	~~$\emptyset$~~
G	E	G

Date...../ ...../ 20 .....

Now drawing the DFA:



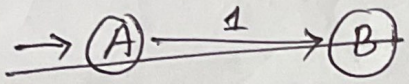
Provati

Scanned with CamScanner

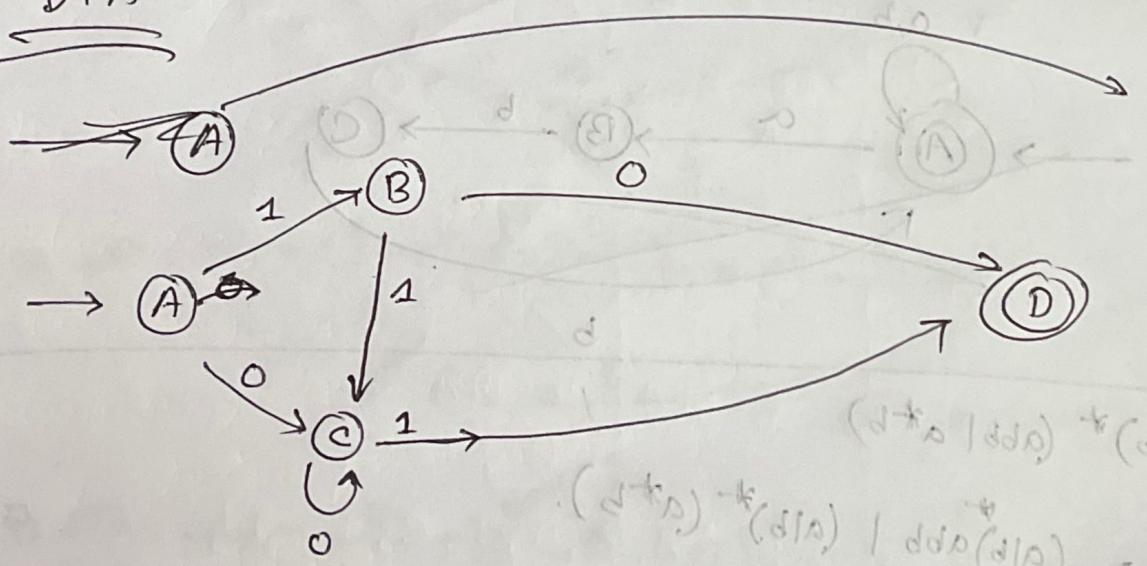
Date...../ ...../ 20 .....

4(a):

i. $10 + (0+1)0^*1$. (RE)



DFA:

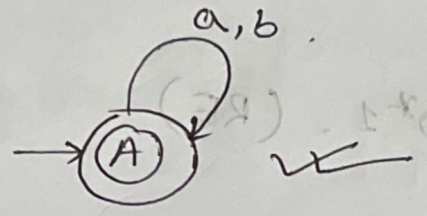


Provati

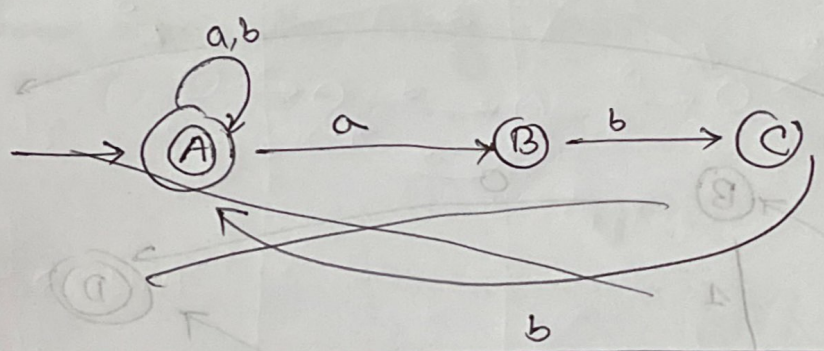
Date...../ ...../ 20 .....

ii. $(a|b)^*(abb|a^*b)$.

$(a|b)^*$:

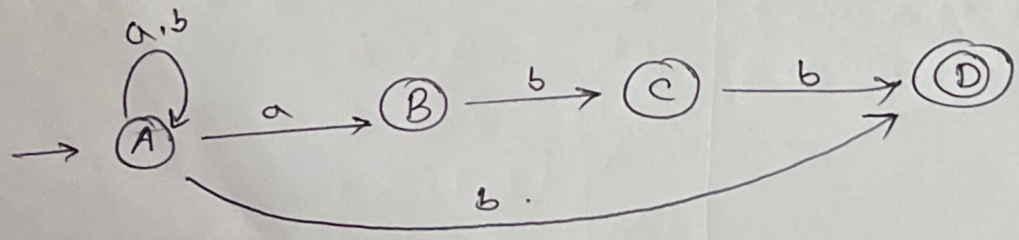


$(a|b)^*(abb|a^*b)$:



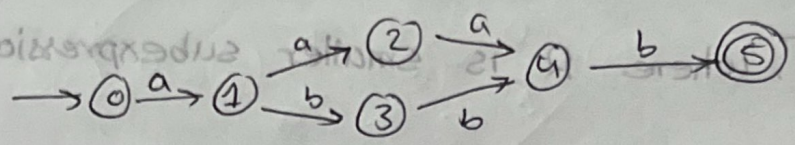
$(a|b)^*(abb|a^*b)$

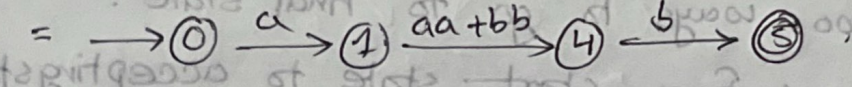
$= (a|b)^*abb \mid (a|b)^*(a^*b)$.

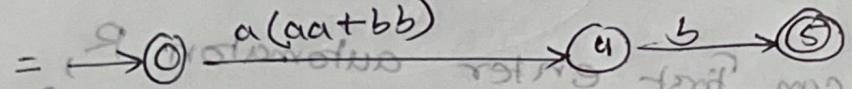


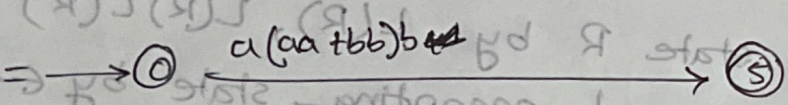
Provati

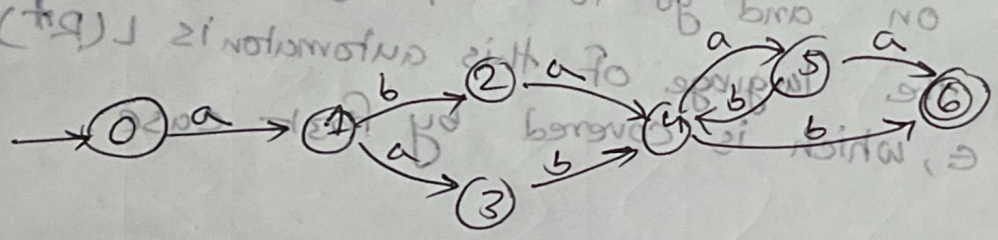
Date...../...../20.....

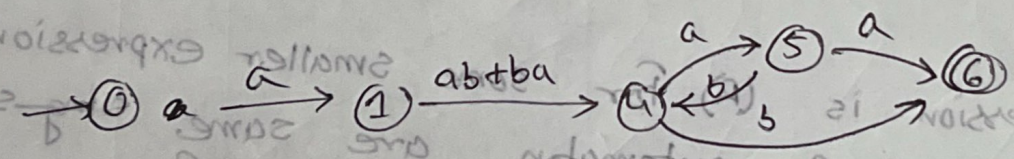


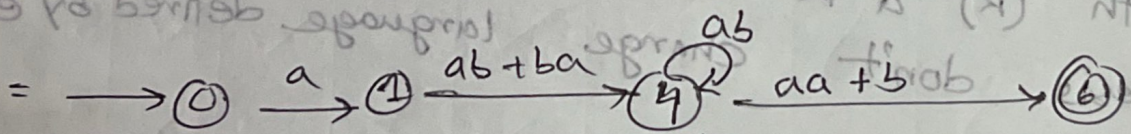


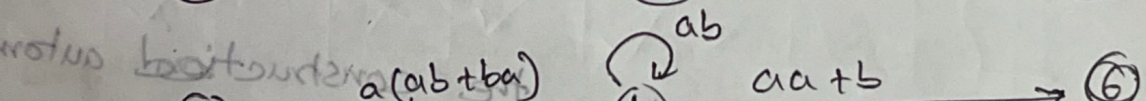


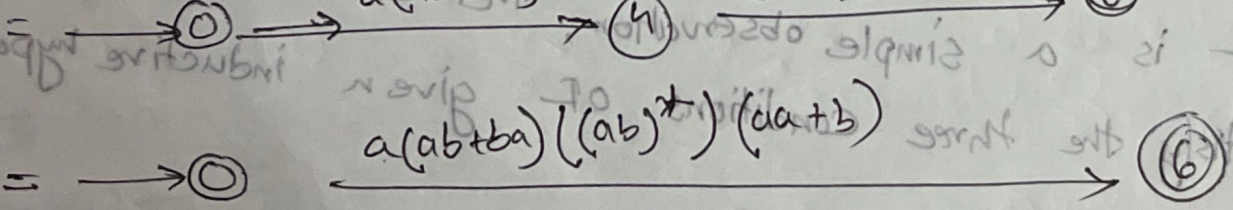












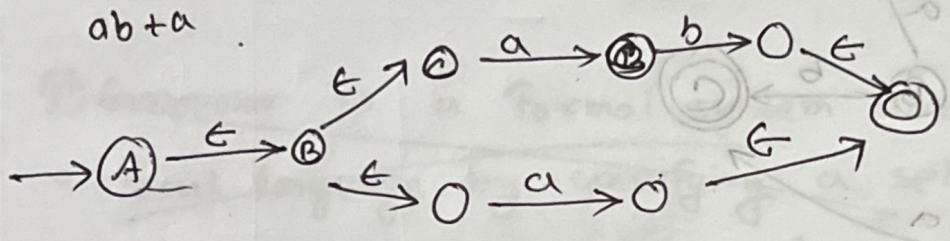
Provati

UC

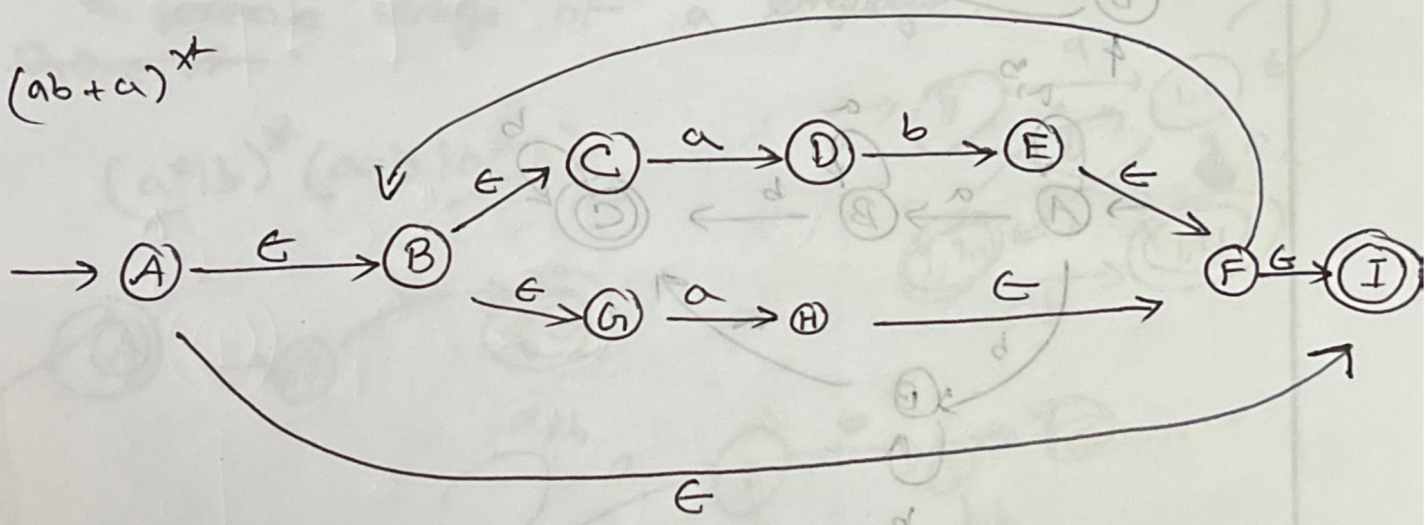
Date...../...../20.....

$R = (ab+a)^*$

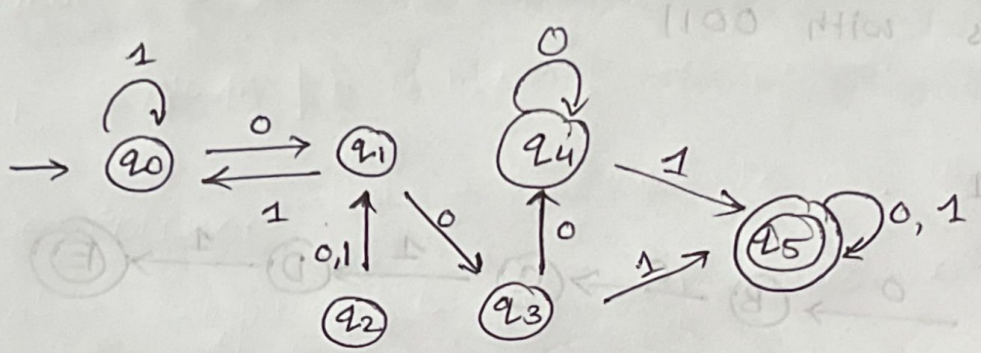
$ab+a$



$(ab+a)^*$



5a



0 - Eq: $\{q_0, q_1, q_2, q_3, q_4\}^{A}$, $\{q_5\}^{B}$.

1 - Eq: $\{q_0, q_1, q_2\}^{A}$, $\{q_3, q_4\}^{B}$, $\{q_5\}^{C}$

2 - Eq: $\{q_0, q_2\}^{A}$, $\{q_1\}^{B}$, $\{q_3, q_4\}^{C}$, $\{q_5\}^{D}$

3 - Eq: $\{q_0\}^{A}$ $\{q_1\}^{B}$ $\{q_2\}^{C}$, $\{q_3, q_4\}^{D}$, $\{q_5\}^{E}$

	0	1
q_0 A	q_1 A	q_0 A
q_1 B	q_3 A	q_0 A
q_2 C	q_1 A	q_1 A
q_3 D	q_4 A	q_5 B
q_4 D	q_4 A	q_5 B
q_5 E	q_5 B	q_5 B

q_0	q_1 B	q_0 A
q_1	q_3 C	q_0 A
q_2	q_1 B	q_1 B
q_3	q_4 C	q_5 D
q_4	q_4 C	q_5 D
q_5	q_5 D	q_5 D

q_0	q_1 A	q_0 A
q_1	q_3 B	q_0 A
q_2	q_1 A	q_1 A
q_3	q_4 B	q_5 C
q_4	q_4 B	q_5 C
q_5	q_5 C	q_5 C

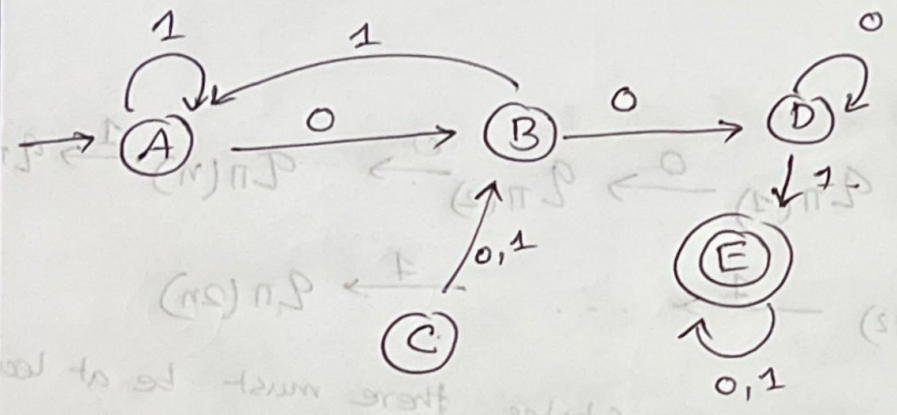
Provati

Scanned with CamScanner

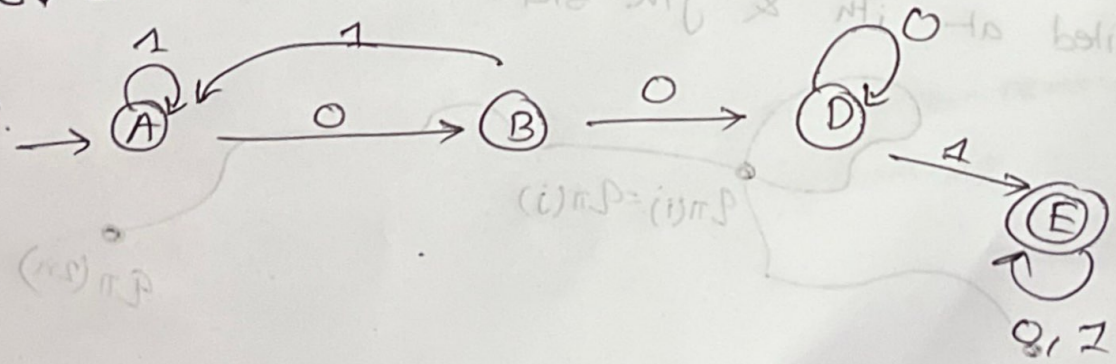
Date...../...../20.....

Now minimized DFA:

State	Inputs 0	1
A	B	A
B	D	A
C	B	B
D	D	E
E	E	E.



C_r can be removed as it has no entry condition.



Statement

$L = \{0^n 1^n \mid n \geq 0\}$ is not Regular language

Date...../...../20.....

5b

Proof: We want to prove that

$L = \{0^n 1^n \mid n = 0, 1, 2 \ldots\}$ is not regular.

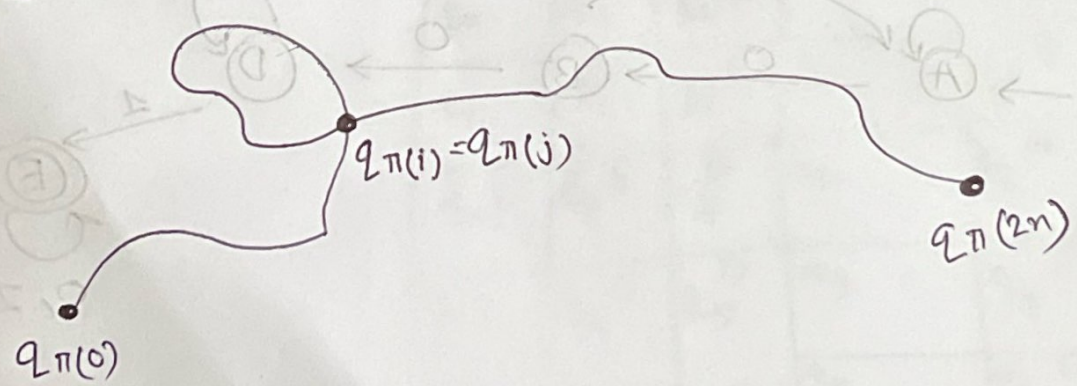
We consider that there exists a DFA M that recognizes L. Let M have n states.

Consider path followed by input string $0^n 1^n$ in M:

$$q_\pi(0) \xrightarrow{0} q_\pi(1) \xrightarrow{0} q_\pi(2) \xrightarrow{0} \cdots q_\pi(n) \xrightarrow{1} q_\pi(n+1)$$

$$\xrightarrow{1} q_\pi(n+2) \xrightarrow{1} \cdots \xrightarrow{1} q_\pi(2n)$$

Since M has only n states, there must be at least one state visited twice. Let this state be visited at ith & jth steps where $j > i$.



Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

By skipping the loop, $a^{n-(j-1)}b^n$ should be accepted, but this is contradictory as $a^{n-(j-i)}.b^n \notin L$.

And hence we found a contradiction, thus proving that L is not a regular language.

6(a) Why we remove null value from a grammar:

1. Parsing loops:

   Consider $A \rightarrow BA$; $B \rightarrow \varepsilon$.

   A naive parser gets stuck in infinite loop as it tries to match ε and keeps calling A, while never reading an input.

2. Ambiguity
3. Algorithms like GNF, CYK require ~~a~~ ε-removal.
4. Causes parsing conflict in bottom up parses used in compilers.

Provati

Date...../ ...../ 20 .....

6(b): Given Grammar:

$S \rightarrow AB \mid BC$

$A \rightarrow BA \mid a$

$B \rightarrow CC \mid b$

$C \rightarrow AB \mid a.$

1,5	S,A,C				
1,4	$\emptyset$	S,A,C	2,5		
1,3	$\emptyset$	{B} 2,4	{B} 3,5		
1,2	{A,S}	{B} 2,3	{S,C} 3,4	{A,S} 4,5	
	{B}	{A,C}	{A,C}	{B}	{A,C} 5,5
	1,1	2,2	3,3	4,4	5,5
	b	a	a	b	a

Now, $(1,2) = (11)(22)$

$= B(A,C) = BA, BC$

Now, ~~BA~~ $A \rightarrow BA$ and $S \rightarrow BC$.

$\therefore (1,2) = 'A,S'$

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

Similarly, $(2,3) = (22)(33)$.

$= (A,C) \times (A,C)$

$= AA, AC, CA, CC$.

Now, the only grammar which yields CC is; $B \to CC$.

$\therefore (2,3) = B$.

For $(3,4) = (33)(44)$.

$= (A,C) \times (B)$

$= AB, CB$

The only grammar ~~y~~ which yield is $S \to AB$ and $C \to AB$ $\therefore (3,4) = S, C$.

For $X_{4,5} = (X_{4,4}, X_{5,5}) = B(A,C) = BA, BC$

Production rules which yields $A \to BA$.

$\therefore X_{4,5} = \{A\}$.

Now, $X_{1,3} = (X_{1,1}, X_{2,3}), (X_{1,2}, X_{3,3})$.

$\therefore (X_{1,1}, X_{2,3}) = \{B\} \times \{B\} = \{B\}$

$(X_{1,2}, X_{3,3}) = \{A,S\} \times \{A,C\} = \{AA, AC, SA, SC\}$.

$\therefore$ ~~No rules~~ y $\therefore$ ~~C~~ $X_{1,3} = \{B\} \cup \{AA, AC, SA, SC\}$

$Y = \{B, AA, AC, SA, SC\}$.

No PR yields Y. Thus $X_{1,3} = \phi$.

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

For $X_{2,4} = (X_{2,2}, X_{3,4}) \cup (X_{2,3}, X_{4,4})$.

Now, $(X_{2,2}, X_{3,4}) = \{\{A,C\} \times \{S,C\}$.

$= \{AS, AC, CS, CC\}$.

and $(X_{2,3}, X_{4,4}) = \{B\} \times \{B\} = \{B\}$.

$\therefore Y = \{B\} \cup \{AS, AC, CS, CC\}$.

$= \{B, AS, AC, CS, CC\}$.

Now, the only production rule, $B \to CC$.

$\therefore X_{2,4} = B$.

For $X_{3,5} = (X_{3,3}, X_{4,5}) \cup (X_{3,4}, X_{5,5})$.

Now, $(X_{3,3}, X_{4,5}) = \{A,C\} \times \{A\} = \{AA, CA\}$.

$(X_{3,4}, X_{5,5}) = \{S,C\} \times \{A\} = \cancel{\{SA, CA\}}$. $\{SA, SC, CA, CC\}$.

~~$\therefore Y = \{AA, CA\} \cup \{SA, CA\} = \{AA, CA, SA\}$.~~

~~Now, Y is produced by the production rules: $\emptyset$.~~

$\therefore X_{3,5} =$

$\therefore Y = \{AA, CA\} \cup \{SA, SC, CA, CC\} = \{AA, CA, SA, SC, CC\}$.

Now Y is produced by only one PR : $B \to CC$.

$\therefore X_{3,5} = B$

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

For $X_{1,4} = (X_{1,1}, X_{2,4}) \cup (X_{1,2}, X_{3,4}) \cup (X_{1,3}, X_{4,4})$

$\therefore (X_{1,1}, X_{2,4}) = \{B\} \times \{B\} = \{B\}$.

$(X_{1,2}, X_{3,4}) = \{A, S\} \times \{S, C\} = \{AS, AC, SS, SC\}$.

$(X_{1,3}, X_{4,4}) = \{\emptyset\} \times \{B\} = \emptyset$.

$\therefore Y = \{B\} \cup \{AS, AC, SS, SC\} = \{B, AS, AC, SS, SC\}$.

Now, Y is not yield by any production rule.

$\therefore X_{1,4} = \emptyset$.

For $X_{2,5} = (X_{2,2}, X_{3,5}) \cup (X_{2,3}, X_{4,5}) \cup (X_{2,4}, X_{5,5})$.

$\therefore (X_{2,2}, X_{3,5}) = \{A, C\} \times \{B\} = \{AB, CB\}$.

$(X_{2,3}, X_{4,5}) = \{B\} \times \{A, S\} = \{BA, BS\}$.

$(X_{2,4}, X_{5,5}) = \{B\} \times \{A, C\} = \{BA, BC\}$.

$\therefore Y = \{AB, CB\} \cup \{BA, BS\} \cup \{BA, BC\} = \{AB, CB, BA, BS, BC\}$.

$\therefore Y$ is yield by PR : S, C, A.

$\therefore X_{2,5} = S, A, C$.

For $X_{1,5} = (X_{1,1}, X_{2,5}) \cup (X_{1,2}, X_{3,5}) \cup (X_{1,3}, X_{4,5}) \cup (X_{1,4}, X_{5,5})$

$\Rightarrow (X_{1,1}, X_{2,5}) = \{B\} \times \{S, A, C\} = \{BS, BA, BC\}$.

$\Rightarrow (X_{1,2}, X_{3,5}) = \{A, S\} \times \{B\} = \{AB, SB\}$.

$\Rightarrow (X_{1,3}, X_{4,5}) = \{\emptyset\} \times \{A, S\} = \emptyset$.

$\Rightarrow (X_{1,4}, X_{5,5}) = \emptyset$.

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

$X_{1,5} = \{BS, BA, BC\} \cup \{AB, SB\}$

$Y = \{BS, BA, BC, AB, SB\}$.

Y is yielded by PRs: S, AC.

$\therefore X_{1,5} = S, A, C$

Since Cell $X_{1,5}$ has start symbol, thus the given grammar accepts the string 'baaba'.

Provati

Date...../ ...../ 20 .....

6(c)

$S \rightarrow 1A \mid 0B$

$A \rightarrow 1AA \mid 0S \mid 0$

$B \rightarrow 0BB \mid 1S \mid 1$

terminals are 0/1

S, A, B are variables

1 S appears on RHS.

∴ we add $S' \rightarrow S$ rule

Current Production rules:

$S' \rightarrow S$

$S \rightarrow 1A \mid 0B$

$A \rightarrow 1AA \mid 0S \mid 0$

$B \rightarrow 0BB \mid 1S \mid 1.$

2. No null production exists.

3. There exists only one unit production, i.e $S' \rightarrow S$

Thus, replacing $S' \rightarrow S$ with

$S' \rightarrow 1A \mid 0B$

$S \rightarrow 1A \mid 0B$

$A \rightarrow 1AA \mid 0S \mid 0$

$B \rightarrow 0BB \mid 1S \mid 1.$

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

4. Now we need to fix the production rule to CNF Format.

* i. Fix terminal in mixed rule.

Add rule $X \to 1$, $Y \to 0$

$\therefore$ ~~$S' \to XA \mid 0Y$~~

$\therefore S' \to XA \mid YB$

$S \to XA \mid YB$

$A \to XAA \mid YS \mid 0$

$B \to YBB \mid XS \mid 1$.

Now, in A and B, there are ~~two~~ more than two terminals in RHS. $\therefore$ $W \to AA$ and $V \to BB$

$\therefore S' \to XA \mid YB$

$S \to XA \mid YB$

$A \to XW \mid YS \mid 0$

$B \to YV \mid XS \mid 1$.

$W \to AA$

$V \to BB$

$X \to 1$

$Y \to 0$

Final CNF Grammar.

Variables: $\{S', S, A, B, X, Y, W, V\}$

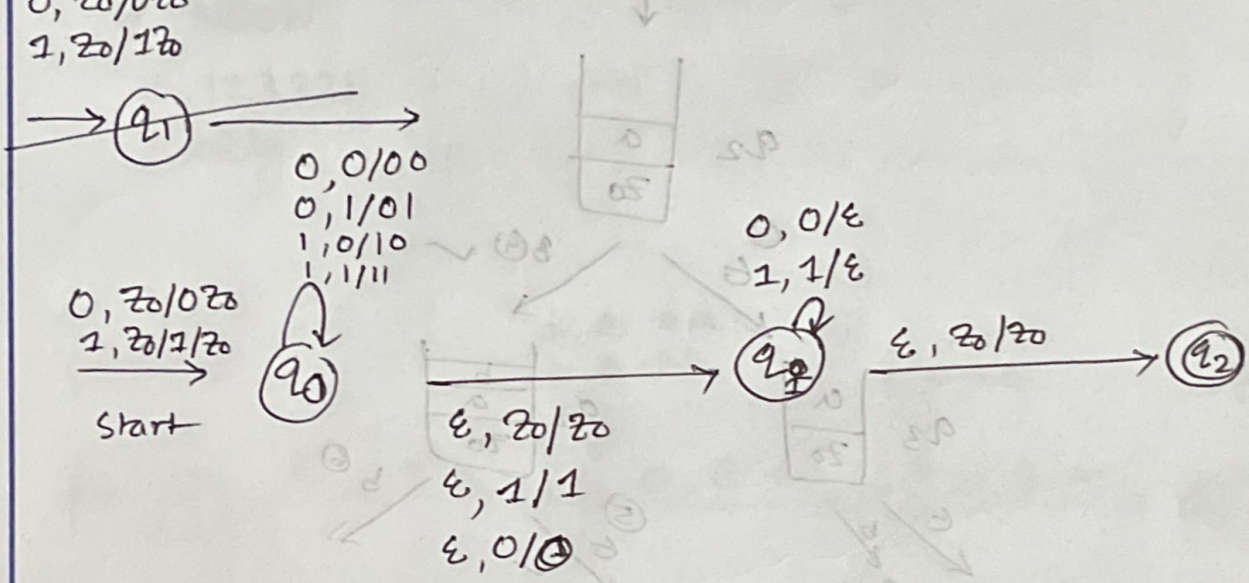
Terminals: $\{0, 1\}$.

Provati

Scanned with CamScanner

7b

$L_{wwr} = \{ ww^R \mid w \in (0+1)^* \}$.



Now, the design of PDA.

1. Start in q_0. In $\emptyset$ q_0, we guess that we are not in middle. Thus, we have not seen the end of string w that is followed by its own reverse. In ~~q_0, we~~ q_0, we read symbols and store them in stack by pushing a copy of input onto stack.

Date...../ ...../ 20 .....

2. At any time, we may guess that we have seen the middle. At this time w will be in stack, with right end of w at top and left end at bottom. We signify this step by spontaneously going to st q_1 state. Since it is a non-deterministic automata, thus we actually make both prediction that we have seen the end of string w and also stay in q_0 and continue to read input & put them onto stack.

3. Once in q_1, we compare input symbol with symbol on top of stack. If the symbols match, then we pop from stack and continue. If they don't match, then the branch dies. Other branches of the non-deterministic automata may survive & lead to acceptance.

4. If we empty the stack, then we have inded seen some input w followed by w^R. This state is represented in q_2.

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

Turing Machines: A type-0 Automaton with infinite memory which accepts Recursively innumerable language.

← Tape head →

→ A tape

a	a	a	a	b	a	b	b	⊔	⊔	⊔

Tape Alphabets: $\Sigma = \{0, 1, a, b, x, z_0\}$.

'⊔' (Blank) is a special symbol.

'⊔' $\notin \Sigma$

Initial configuration:

Tape head

a	a	a	a	b	a	b	b	a	a	a	⊔	⊔	⊔	⊔	....

Input string — Blanks to infinity

Operations:

→ Read/Scan symbol below tape head

→ Update/Write a symbol below tape head

→ Move ~~te~~ tape head one step left

→ Move tape head one step right

Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

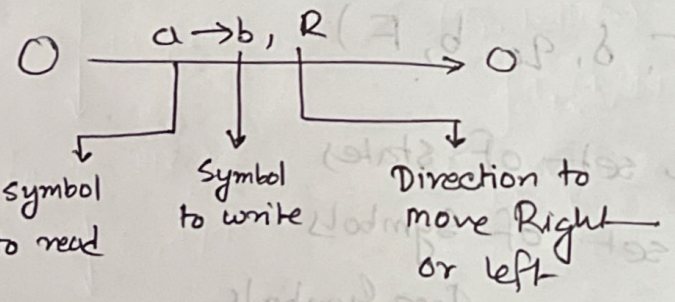
1. A control portion which controls the turing machine.
2. It is deterministic.

Rules of operation:-1

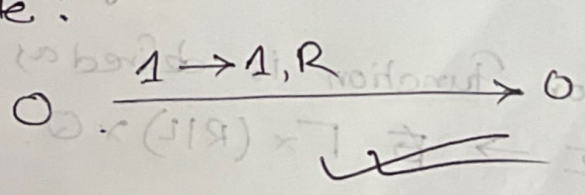
At each state of computation.

1. Read the current symbol
2. Update/Write the same cell.
3. Move exactly ~~left or~~ one cell left or right.

N.B If we are at left end of tape, trying to move left then do not move.



To not update:



Provati

Scanned with CamScanner

2. Control is sort of FSM.

→ Initial State.

→ Final State → Accept state / → Reject state

Computation ~~can~~ can either.

1) HALT & ACCEPT
2) HALT & REJECT
3) LOOP (Fails to Halt)

Formal Definition of Turing Machine.

A Turing Machine is defined as a set of 7 Tuples

$$(Q, \Sigma, \Gamma, \delta, q_0, b, F)$$

$Q \rightarrow$ Non-empty set of states

$\Sigma \rightarrow$ " set of symbols

$\Gamma \rightarrow$ " " " Tape symbols

$\delta \rightarrow$ Transition function is defined as

$$Q \times \Sigma \rightarrow \Gamma \times (R|L) \times Q.$$

$q_0 \rightarrow$ initial state

$b \rightarrow$ Blank symbol

$F \rightarrow$ Set of Final

Provati

Scanned with CamScanner

Production Rule:

$\delta(q_0, a) \longrightarrow (q_1, Y, R)$

- R → Goes Right
- Y → Writes Y
- q_1 → Goes to state q_1.

Turing's Thesis: Any computation that can be mechanically performed can be performed by a Turing Machine.

Arguments for accepting the Thesis:

1. Anything that can be done on an existing digital computer can also be done by Turing Machine
2. No problem yet exist for which an algorithm exists for which a Turing Machine Program cannot be written

A Language L & Σ is recursively Enurable, if there exists a Turing Machine that accepts it.

Provati

Scanned with CamScanner

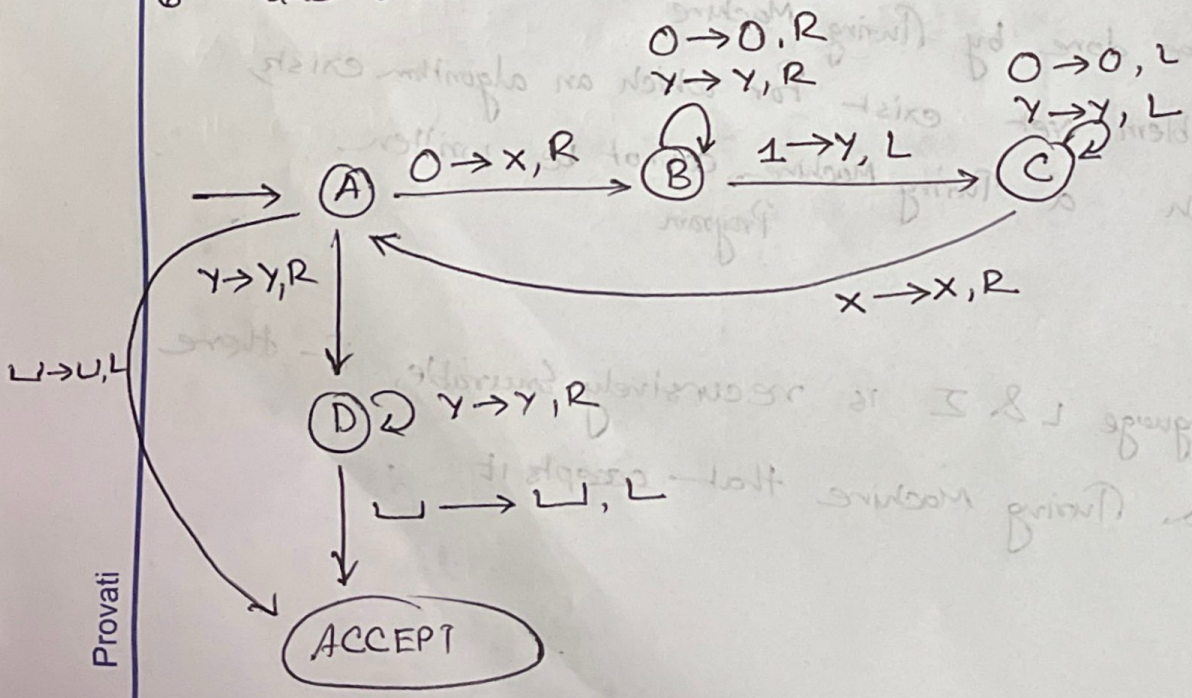
Date...../ ...../ 20 .....

Design TM that accepts $L = 0^N 1^N$

0	0	0	0	1	1	1	1	⊔	⊔

Algorithm:

1. Change 0 to x.
2. Move Right to 1st 1.
   B → IF None: REJECT
3. Change '1' to y
4. Move Left to left-most '0'
5. Repeat 1 to 4 until no-more '0's
6. Make sure no more 1's remains.



Provati

Scanned with CamScanner

Date...../ ...../ 20 .....

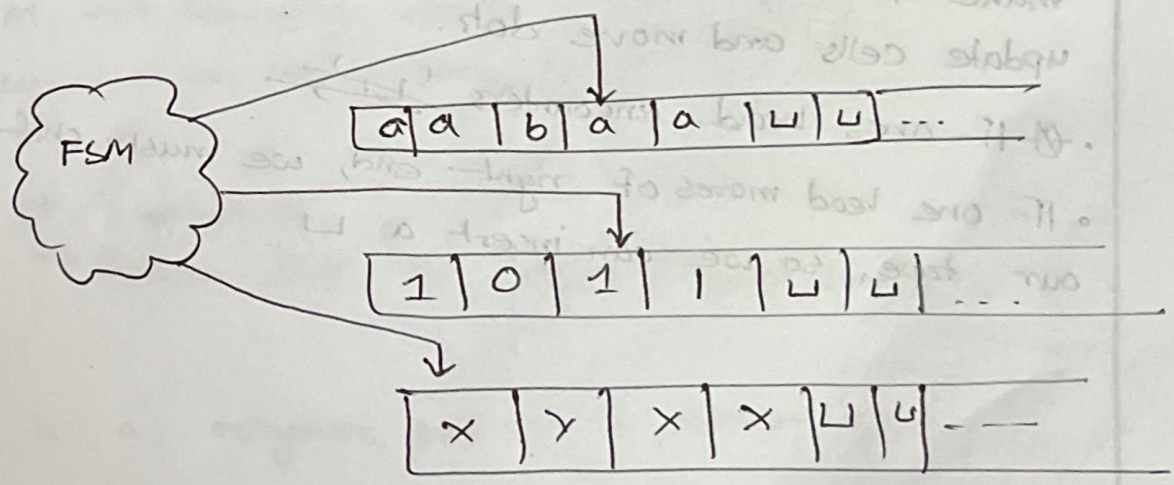
Multi-~~step~~ Tape Turing Machines:

Every multitape Turing Machine has an equivalent Single Tape Turing Machine

Proof: Given a Multitape Turing machine, we need to show how to build a single tape Turing Machine were.

1. we store all tapes on a single tape.
2. Show ~~we~~ how we represent multi-tape heads in a single tape head.
3. ~~Ont~~ One to one mapping of multitape TM moves to one or multiple moves of a single tape TM moves.

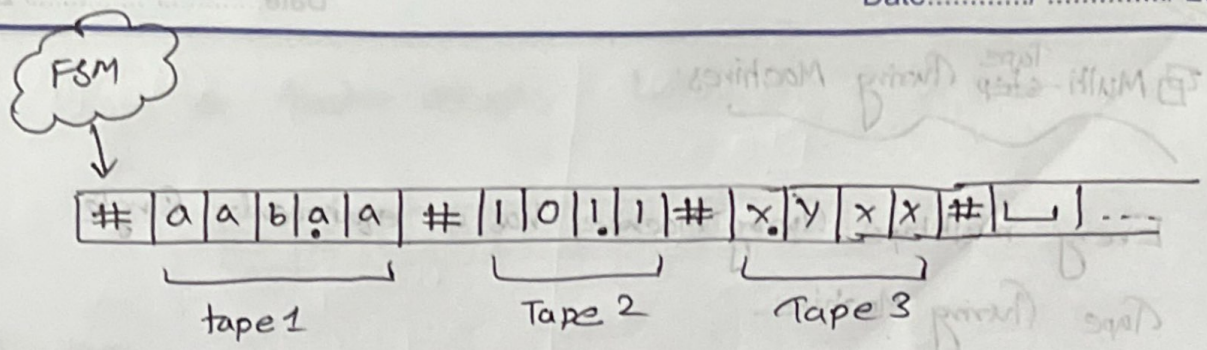
Ex:



Q $\xrightarrow{a1x \to a1y, LLR}$ R

Provati

Scanned with CamScanner



- Add 'dots' to show where head 'k' is.
- To simulate a transition from state Q to state R, we must scan the tape to check which symbol is under the kth Tape head.
- Once we determine the symbols, we are ready to make updates, we scan through entire tape again, update cells and move dots.
- ~~If one head encounters '⊔',~~
- If one head moves of right end, we must shift our tape, so we can insert a ⊔.

Provati

Scanned with CamScanner