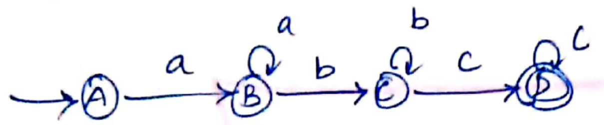
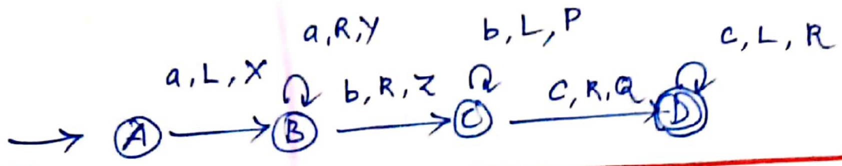
mood-book

② Construct a TM for $L = \{a^n b^n c^n \mid n \geq 1\}$

May 2015, May 2017 → 5 marks
Nov 2016
Dec 2009 – 16 marks





$q_0 \to A$ $\quad B \to B$

$F \to D$ $\quad \Gamma \to \{a, b, c\}$

$\Sigma \to \{a, b, c\}$

$Q \to \{A, B, C, D\}$

$\delta \to \delta(A, a) = (B, L, X)$

$\delta(B, a) = (B, R, Y)$

$\delta(B, b) = (C, R, Z)$

$\delta(C, b) = (C, L, P)$

$\delta(C, c) = (D, R, Q)$

$\delta(D, c) = (D, L, R)$

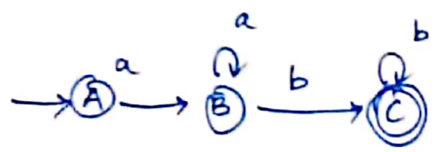
State	O/t State	Tape movement	O/t locations
A	B	L	X
B	B	R	Y
B	C	R	Z
C	C	L	P
C	D	R	Q
D	D	L	R

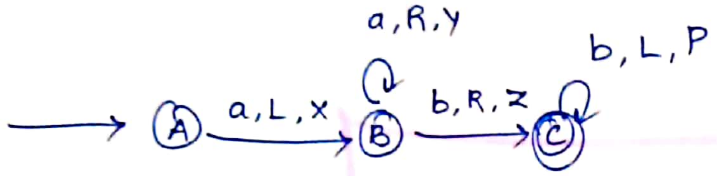
Scanned with CamScanner

Turing Machine.

Construct Turing machine for the Language, $L = \{a^n b^n\}$ Where $n \geq 1$.

JNTUH.





∴ n value is 1 or more then 1. So we are taking $n = 2$

$a^2 b^2$

$= aabb$.

$\delta(A, a) = (B, L, X)$

$\delta(B, a) = (B, R, Y)$

$\delta(B, b) = (C, R, Z)$

$\delta(C, b) = (C, L, P)$

States	O/t State	Tape Directions	O/t.
A	B	L	X
B	B	R	Y
B	C	R	Z
C	C	L	P

Tuples.

$q_0 \rightarrow A$

$F \rightarrow C$

$Q \rightarrow \{A, B, C\}$

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

δ

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

$B \rightarrow B$.

3 Construct TM to add Two given Integers.

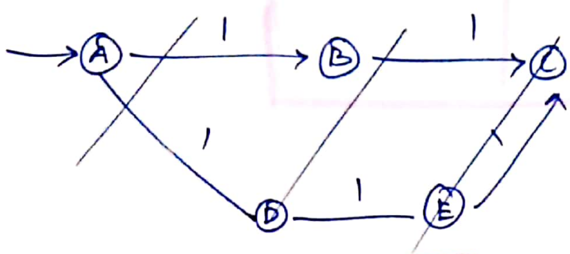
Nov 2008 – 8marks

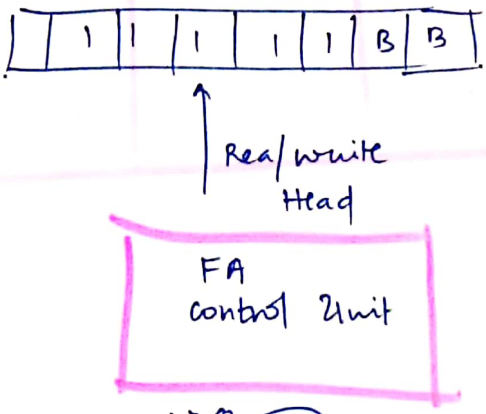
$a = 2$

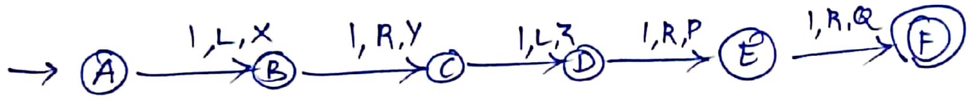
$b = 3$

$2 + 3 = 5$

$11 + 111 = 11111$







1) $q_0 \rightarrow A$

2) $F \rightarrow F$

3) $Q \rightarrow \{A, B, C, D, E, F\}$

4) $\delta \rightarrow (A,1) = (B, L, X)$

$\delta(B,1) = (C, R, Y)$

$\delta(C,1) = (D, L, Z)$

$\delta(D,1) = (E, R, P)$

$\delta(E,1) = (F, R, Q)$

5) $\Sigma = \{1\}$

6) $\Gamma = 1, B$

7) $B \rightarrow B$

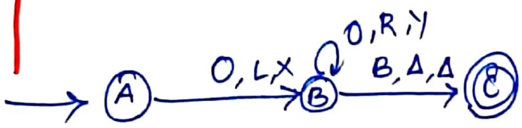
State	I/p symbol	O/t	Tape movement
A	1	X	L
B	1	Y	R
C	1	Z	L
D	1	P	R
E	1	Q	R

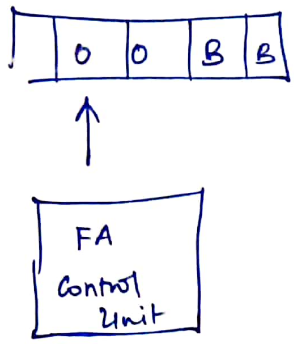
L → Left

R → Right.

④ Given $\Sigma = \{0,1\}$, Design a Turing machine That accepts the Language Denoted by Regular Expression 00^*

Feb - 2008 - 8marks
May - 2017 - 5marks





$\delta(A, 0) = (B, L, X)$

$\delta(B, 0) = (B, R, Y)$

$\delta(B, B) = (C, \Delta, \Delta)$

$q_0 \rightarrow A$

$F \rightarrow C$

$Q \rightarrow \{A, B, C\}$

$\Sigma \rightarrow \{0\}$

$\Gamma \rightarrow 0, B$

$B \rightarrow B$

8 Design a TM that recognizes the set $\{0^{2n}1^n \mid n>=0\}$

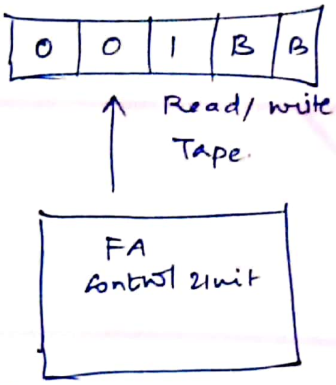
Nov-2008 8marks

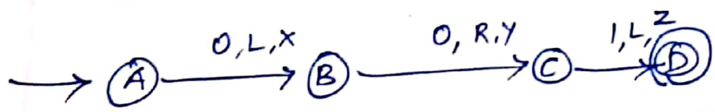
$n = 1$

$= \{0^{2(1)}1^1$

$= 0^2 1^1$

$= 001$





State	I/P Symbol	O/P State	Tape movement
A	0	X	L
B	0	Y	R
C	1	Z	L
D	B		

* Tuples of Turing Machine.

$q_0 \rightarrow A$

$Q \rightarrow \{A, B, C, D\}$

$F \rightarrow D$

$\Sigma \rightarrow \{0, 1\}$

$B \rightarrow B$

$\Gamma \rightarrow \{0, 1, B\}$

$\delta \rightarrow \quad \delta(A, 0) = (B, L, X)$

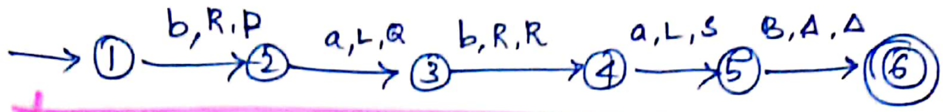
$\delta(B, 0) = (C, R, Y)$

$\delta(C, 1) = (D, L, Z)$

⑤ Construct a TM For a language Having equal number of a's and b's in it over the Input set $\Sigma = \{a, b\}^*$

JNTU-H

baba



$\delta(1, b) = (2, R, P)$

$\delta(2, a) = (3, L, Q)$

$\delta(3, b) = (4, R, R)$

$\delta(4, a) = (5, L, S)$

$\delta(5, B) = (6, \Delta, \Delta)$

$q_0 \rightarrow 1$

$F \rightarrow 6$

δ

$Q \rightarrow \{1, 2, 3, 4, 5, 6\}$

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

$B \rightarrow \{B\}$

$\Gamma \rightarrow \{a, b, B\}$

State	I/P Symbol	Tape movement	O/t. Values
1	b	R	P
2	a	L	Q
3	b	R	R
4	a	L	S
5	B	Δ	Δ

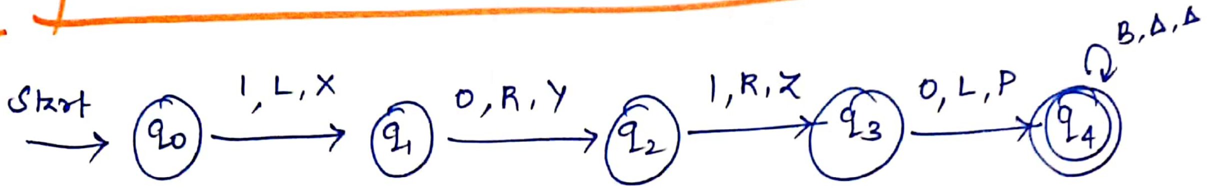
Scanned with CamScanner

⑦ Construct TM for obtaining two's complement of a given binary numbers.

JNTU H

Ex:

	0110	
	1001	1's complement
	+ 1	adding 1
	1010	



$\delta(q_0, 1) = (q_1, L, X)$

$\delta(q_1, 0) = (q_2, R, Y)$

$\delta(q_2, 1) = (q_3, R, Z)$

$\delta(q_3, 0) = (q_4, L, P)$

$q_0 \rightarrow q_0$

$F \rightarrow q_4$

$B \rightarrow B,$

$\Gamma \rightarrow 0, 1, B$

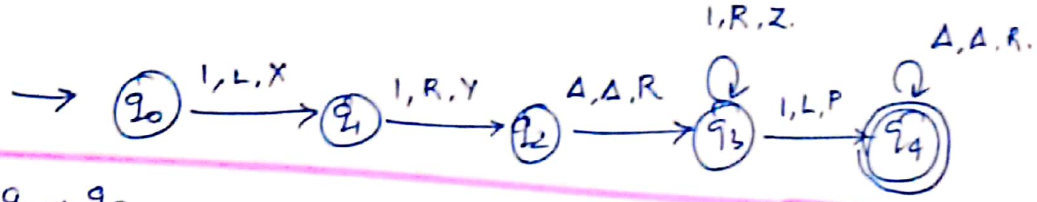
$Q \rightarrow \{q_0, q_1, q_2, q_3, q_4\}$

$\Sigma \rightarrow \{0, 1, B\}$

Input State	Input Symbol	Tape movement	O/P.
q_0	1	L	X
q_1	0	R	Y
q_2	1	R	Z
q_3	0	L	P
q_4	B	Δ	Δ

⑥ construct TM for copying the Input ~~binary~~ unary. string on the Tape. JNTU – H

1 1 1 1



① $q_0 \rightarrow q_0$

② $F \rightarrow q_4$

③ $Q \rightarrow \{q_0, q_1, q_2, q_3, q_4\}$

④ $\delta \rightarrow \delta(q_0, 1) = (q_1, L, X)$

$\delta(q_1, 1) = (q_2, R, Y)$

$\delta(q_2, \Delta) = (q_3, \Delta, R)$

$\delta(q_3, 1) = (q_3, R, Z)$

$\delta(q_3, 1) = (q_4, L, P)$

$\delta(q_4, \Delta) = (q_4, \Delta, R)$

⑤ $\Sigma \rightarrow \{1\}$

$\Gamma \rightarrow \{1\}$

$B \rightarrow B$

State	I/P	Tape movement	O/t
q_0	1	L	X
q_1	1	R	Y
q_2	Δ	Δ	Z.
q_3	1	R	Z ~~q_3~~
	1	L	P
q_4	Δ	Δ	R

(1) Construct TM for the Function $f(x) = x+3$.

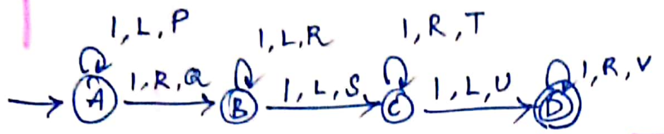
may 2013 - 8 marks

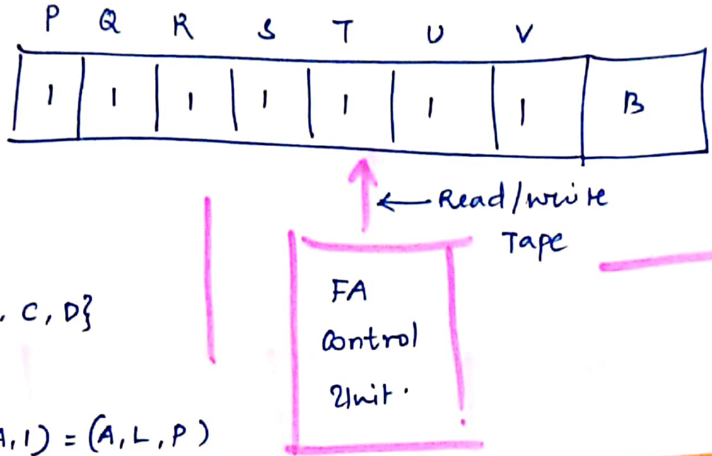
Consider x is a unary number.

$x = 4 \rightarrow 1111$

$4 + 3 = 7$

$1111 + 111 = 1111111$





$q_0 \rightarrow A$

$F \rightarrow D$

$Q \rightarrow \{A, B, C, D\}$

$\Sigma \rightarrow \{1\}$

$\delta \rightarrow \delta(A,1) = (A, L, P)$

$\delta(A,1) = (B, R, Q)$

$\delta(B,1) = (B, L, R)$

$\delta(B,1) = (C, L, S)$

$\delta(C,1) = (C, R, T)$

$\delta(C,1) = (D, L, U)$

$\delta(D,1) = (D, R, V)$

$\Gamma \rightarrow \{1\}$

$B \rightarrow B$

I/P State	O/t State	Tape movement	O/t Value
A	(1, A)	P L	P
A	(1, B)	Q R	Q
B	(1, B)	R L	R
B	(1, C)	S L	S
C	(1, C)	T R	T
C	(D, 1)	U L	U
D	(1, D)	V R	V

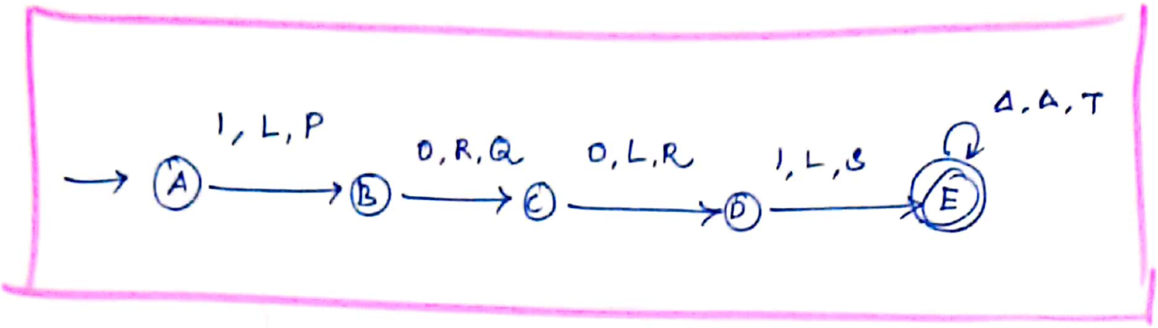
Design Turing machine over {0,1}

April - 2011 - 11 marks

L = { w | |w| is a multiple of 3 }.

3, 6, 9, 12 . . . .

1001 → 9.



$q_0 \to A$

$F \to E$

$\Sigma \to \{0, 1\}$

$Q \to \{A, B, C, D, E\}$

$\delta \to \delta(A, 1) = (B, L, P)$

$\delta(B, 0) = (C, R, Q)$

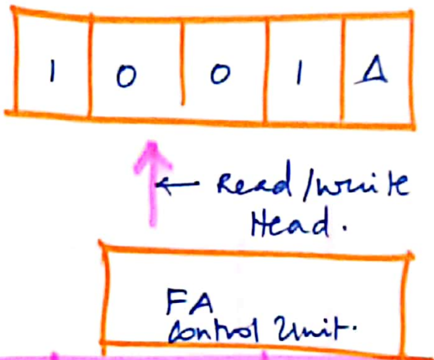
$\delta(C, 0) = (D, L, R)$

$\delta(D, 1) = (E, L, S)$

$\delta(E, \Delta) = (E, \Delta, T)$

$\Gamma \to \{0, 1\}$

$B \to B$



State	O/P State	Tape movement	O/P Value
A	(1, B)	L	P
B	(0, C)	R	Q
C	(0, D)	L	R
D	(1, E)	L	S
E	(Δ, E)	Δ	T

Design a Turing machine to find square of a given Integer.

March-2006 - 10 marks

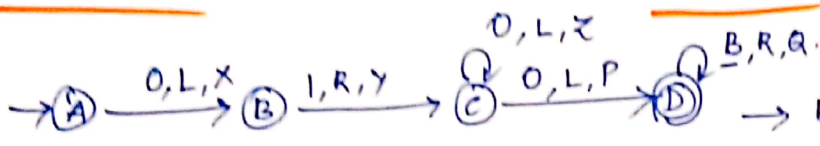
Ex: 2

: $2^2 \rightarrow 4$.

If you want to take unary number, 1111

If you want to take binary number, 0100

Now we can construct for 0100



→ Blank symbol.

→ Right Direction.

→ Q memory location.

$q_0 \rightarrow A$

$F \rightarrow D$

$Q \rightarrow \{A, B, C, D\}$

$\Sigma \rightarrow \{0,1\}$

$\delta \rightarrow \delta(A,0) = (B,L,X)$

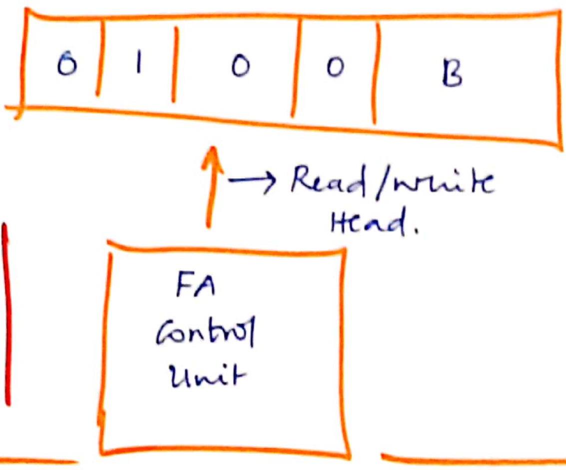
$\delta(B,1) = (C,R,Y)$

$\delta(C,0) = (C,L,Z)$

$\delta(C,0) = (D,L,P)$

$\Gamma \rightarrow \{0,1\}$

$B \rightarrow B$



State	O/t State	Tape movement	O/t values
A	(0, B)	L	X
B	(1, C)	R	Y
C	(0, C)	L	Z
C	(0, D)	L	P

(*) Design a Turing machine to find out GCD of two given numbers. (8, 4)

Dec 2009 - 16 marks

2 | 8 , 4
2 | 4 , 2
2 , 1

$8 \rightarrow \underline{2 \times 2} \times 2$

$4 \rightarrow \underline{2 \times 2}$

$\boxed{GCD = 4.}$

2 | 8 , 16
2 | 4 , 8
2 | 2 , 4
1 , 2

$\Rightarrow \cancel{8}$.

$8 \rightarrow \underline{2 \times 2} \times 2$

$4 \rightarrow \underline{2 \times 2}$.

$= 4.$

Ex:

$20 \rightarrow \underline{2 \times 2} \times 5$

$8 \rightarrow \underline{2 \times 2} \times 2$

$\rightarrow 2 \times 2$

$= 4.$

Ex:

$8 \rightarrow \underline{2 \times 2 \times 2}$

$16 \rightarrow \underline{2 \times 2 \times 2} \times 2$

$\Rightarrow 8.$

Ex:

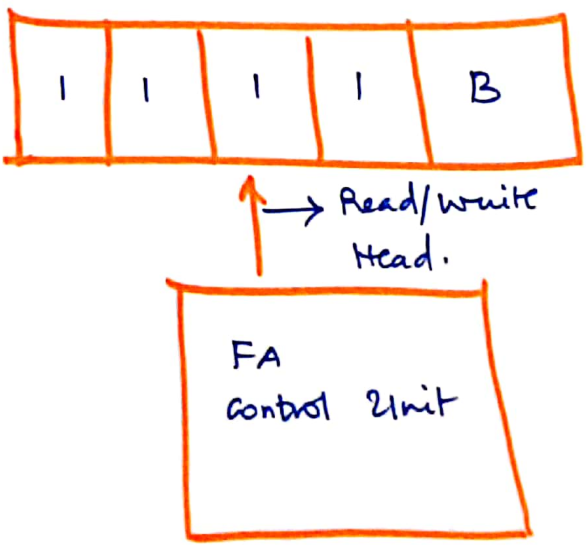
$40 \rightarrow \underline{2 \times 2 \times 2} \times 5$

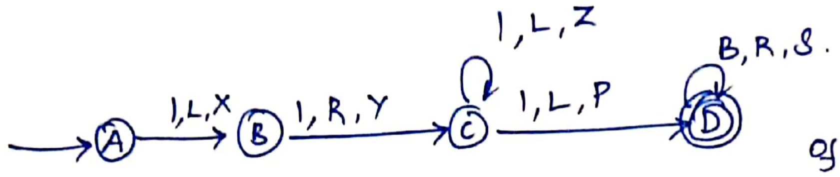
$16 \rightarrow \underline{2 \times 2 \times 2} \times 2$

$= 8.$

If it is unary number, then,

$1111 \rightarrow 4.$





∴ There are two numbers 8 and 4. after Finding the GCD of Two numbers, 4 is The Answer, considering unary number, 4 is 1111.

So we are constructing the Turning machine for 4 is "1111".

$q_0 \to A$

$F \to D$

$Q \to \{A, B, C, D\}$ It is

$\Sigma \to \{1\} \to$ because unary number.

$\delta \to \delta(A,1) = (B, L, X)$

$\delta(B,1) = (C, R, Y)$

$\delta(C,1) = (C, L, Z)$

$\delta(C,1) = (D, L, P)$

$\delta(D,B) = (D, R, S)$

$\Gamma \to \{1\}$

$B \to B$

State	O/P State	Tape movement	output values
A	(1,B)	Left	X
B	(1,C)	Right	Y
C	(1,C)	Left	Z
C	(1,D)	Left	P
D	(B,D)	Right	S

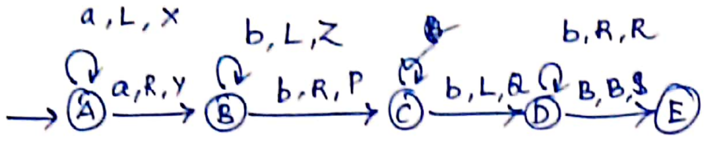
Construct the TM for the Following. $L = \{a^n b^{2n} \mid n \geq 1\}$

$n \geq 1$ means, the Values are,

1, 2, 3, 4 . . . . .

So n Value is 2.

$L = \{a^2 b^{2 \times 2} = aabbbb$



$\delta(A, a) = (A, L, x)$

$\delta(A, a) = (B, R, Y)$

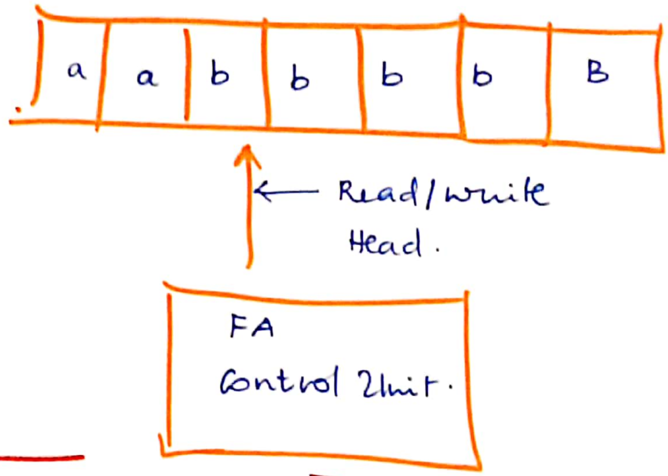
$\delta(B, b) = (B, L, Z)$

$\delta(B, b) = (C, R, P)$

$\delta(C, b) = (D, L, Q)$

$\delta(D, b) = (D, R, R)$

$\delta(D, B) = (E, B, S)$



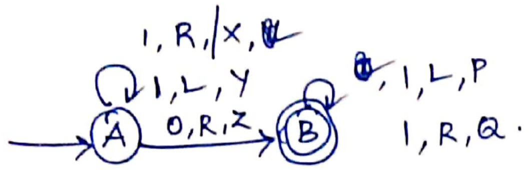
State	I/P State	Tape movement	output Values
A	(a, A)	L	x
A	(a, B)	R	Y
B	(b, B)	L	Z
B	(b, C)	R	P
C	(b, D)	L	Q
D	(b, D)	R	R
D	(B, E)	B	S

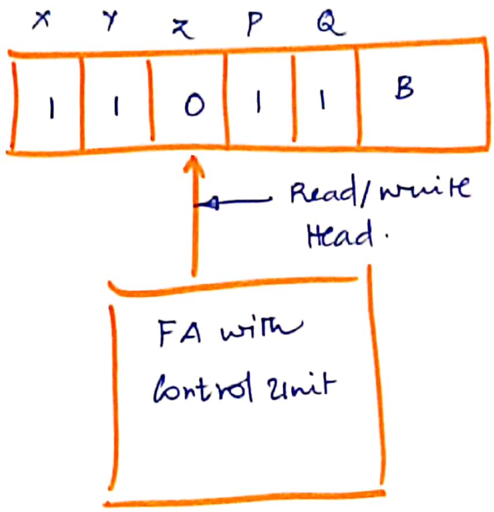
(*) Construct a Turing machine that will accept the Language consists of all palindromes of 0's and 1's.

April 2018 - 5marks

Ex: 11011

If u reverse the string, you will get same string.





$\delta(A,1) = (A,R,X)$

$\delta(A,1) = (A,L,Y)$

$\delta(A,0) = (B,R,Z)$

$\delta(B,1) = (B,L,P)$

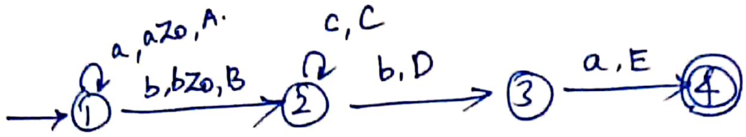
$\delta(B,1) = (B,R,Q)$

$q_0 \rightarrow A$ $Q \rightarrow \{A,B\}$ $\Gamma \rightarrow 0,1$

$F \rightarrow B$ $\Sigma \rightarrow \{0,1\}$ $B \rightarrow B$

State	I/P	O/t State	Tapemovement	O/t memory
A	1	A	R	X
A	1	A	L	Y
A	0	B	R	Z
B	1	B	L	P
B	1	B	R	Q

wCw^R.



abCba.

$\vdash$

Elimination of useless symbols.

Null productions.

$S \to BAAB$

$A \to 0A2 \mid 2A0 \mid \epsilon$

$B \to AB \mid 1B \mid \epsilon$

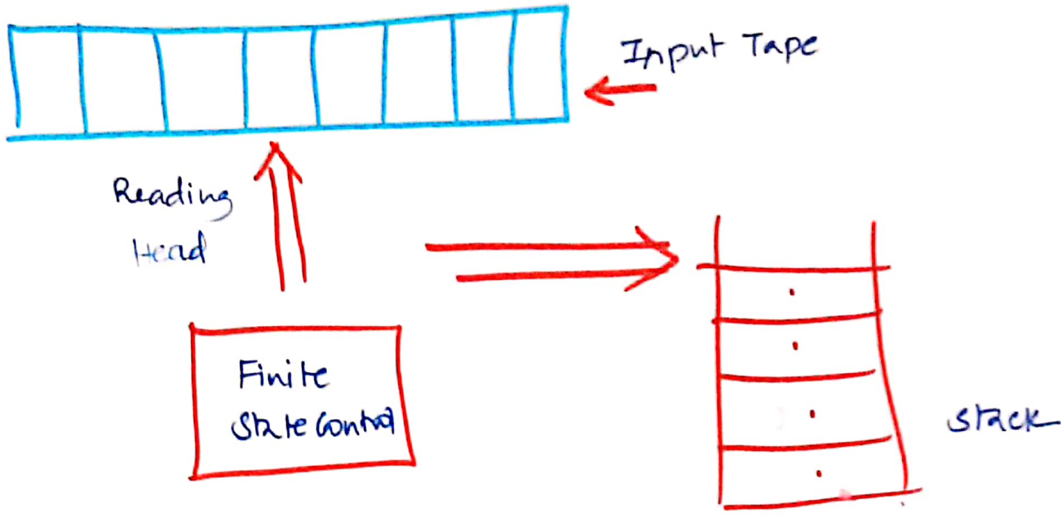
$S \to BAAB \mid ABAAB \mid 1BAAB \mid AAB \mid$

$A \to 0A2 \mid 02 \mid 20 \mid 2A0$

$B \to AB \mid 1B \mid A \mid 1$

Scanned with CamScanner

Push Down Automata.



Stack Symbol. Γ input symbols in the stack.

* consists of Finite Tape

* LIFO Fashion.

7 Tuples.

q_0

Σ

F

Q

$\delta \to Q \times \Gamma^*$

is the starting

$Z_0 \to$ Stack symbol.

$\Gamma \to$ set of pushdown symbol.

The Input which is read from the stack.

UNIT-3

CONTEXT FREE GRAMMARS

Definition. G is a grammar, it consists of components,

$G = (T, N.T, P, S)$

T → Terminal → Σ

N.T → states - Non-Terminal

P → Productions

S → start symbol.

Generating Variables :

Variables means Input symbols.(or) Terminals.

variables can be derived,

Ex: $S \to aB \mid bB \mid \epsilon.$

$S \to aAB \mid AB$

Derivations :

The generation of language using specific Rules is called Derivation.

Ex: $S \to aA$

$S \to aAB$

$S \to a$

$S \to \epsilon.$

$S \to AB$

Left most Derivations :

Strings are Derived from the left side of Non - Terminal is called.

$$S \to S+S \mid S*S \mid a \mid b$$

$S \to \underline{S} * S$

$\to a * \underline{S}$

$\to a * \underline{S} + S$

$\to a * a + \underline{S}$

$\to a * a + b$

$S \to a$

$S \to S+S$

$S \to a$

$S \to b$

Right most Derivation.

$S \to S * \underline{S}$

$\to S * S + \underline{S}$

$\to S * \underline{S} + b$

$\to \underline{S} * a + b$

$\to a * a + b$

$S \to S+S$

$S \to b$

$S \to a$

$S \to a$

Consider a CFG $S \to bA \mid aB$

$A \to aS \mid aAA \mid a$

$B \to bS \mid aBB \mid b$

Find Leftmost Derivation and Right most Derivations for

W = aaabbabbba.

Left most Derivation

$S \to a\underline{B}$ aBB

$\to aa\underline{B}B$

$\to aaa\underline{B}BB$ $B \to aBB$

$\to aaab\underline{B}B$ $B \to b$

$\to aaabb\underline{B}$ $B \to b$

$B \to aBB$

$\to aaabba\underline{B}B$ $B \to b$

$\cancel{B} \to bS$ $B \to bS$

$\to aaabbab\underline{b}S$

$S \cancel{\to} bA$

$\to aaabbabb\underline{A}B\,\underline{S}$ $S \to bA$

$\to aaabbabbb\underline{A}$ $A \to a$

$$\boxed{S \to aaabbabbba.}$$

Right most Derivation:

aaabbabbba.

$S \to aB$ $\qquad B \to aBB$

$\to aaBB$

$\to aaBbS$ $\qquad B \to bS$

$\to aaBbbA$ $\qquad S \to bA$

$\to aaBbba$ $\qquad A \to a.$

$\qquad B \to aBB$

$\to aaaBBbba$ $\qquad B \to b$

$\to aaaBbbba$ $\qquad B \to bS$

$\to aaabSbbba.$ $\qquad S \to bA$

$\to aaabbAbbba$ $\qquad A \to a$

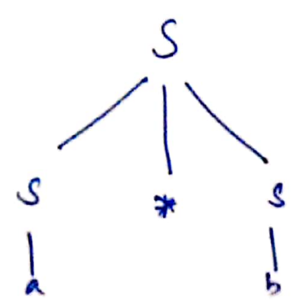
$\boxed{S \to aaabbabbba.}$

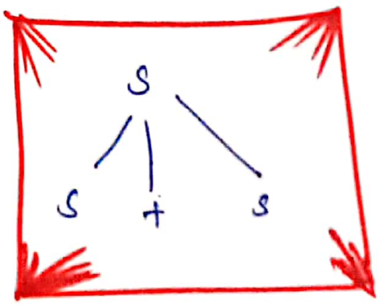
Derivation Trees:

A → nodes

1. The collections of children from left to Right.
2. The Non-Terminals always in Root.

Ex: $S \to S+S \mid S*S \mid a \mid b$





Derivation Tree (or) Parse Tree.

1. The Root node is starting Symbol.
2. (States) Node should be in Right or left node.
3. operations are intermediate nodes.

Consider the grammar

$S \to S+S \mid S*S \mid a \mid b$. Construct derivation Tree for string $w = a*b+a$.

$S \to \underline{S}*S$

$S \to a$

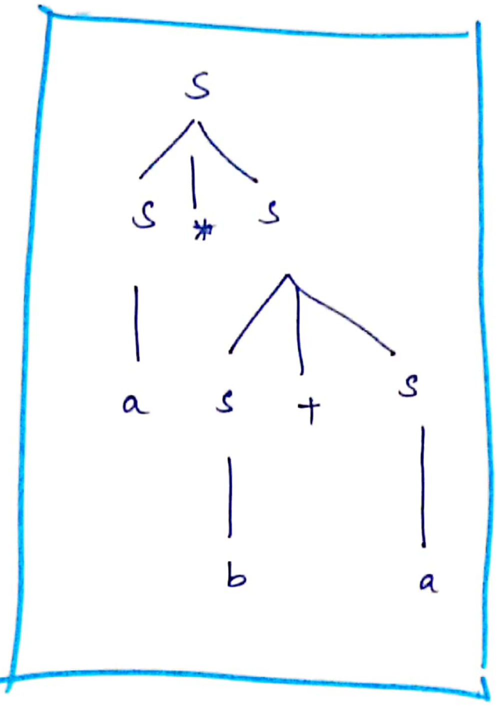
$\to a * S$ ~~$*S$~~

$S \to S+S$

$\to a * \underline{S}+S$

$\to a * b + \underline{S}$

$$\boxed{S \to a*b+a}$$



Consider a grammar G Having productions

$S \rightarrow aAS \mid a$

$A \rightarrow SbA \mid SS \mid ba$

show that $S \stackrel{*}{\Rightarrow} aabbaa$. and construct a Derivation Tree whose yield is aabbaa.

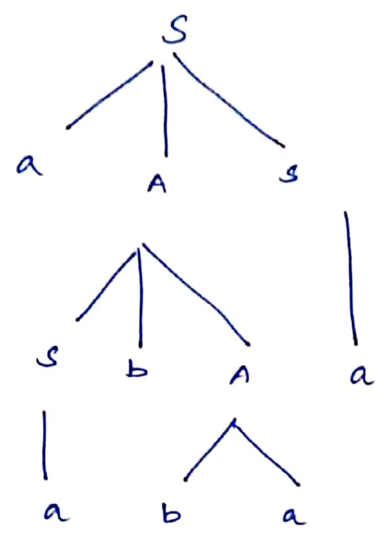
$S \rightarrow a\underline{A}S$

$A \rightarrow SbA$ $\quad \rightarrow a\underline{S}bAS$

$S \rightarrow a$ $\quad \rightarrow aab\underline{A}S$

$A \rightarrow ba$ $\quad \rightarrow aabba\underline{S}$

$S \rightarrow a$ $\quad \boxed{S \rightarrow aabbaa}$



Consider the grammar G whose products are

$S \rightarrow 0B \mid 1A$

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

$B \rightarrow 1 \mid 1S \mid 0BB$. Find (a) leftmost (b) Rightmost Derivation for string 00110101., and construct Derivation tree also.

Scanned with CamScanner

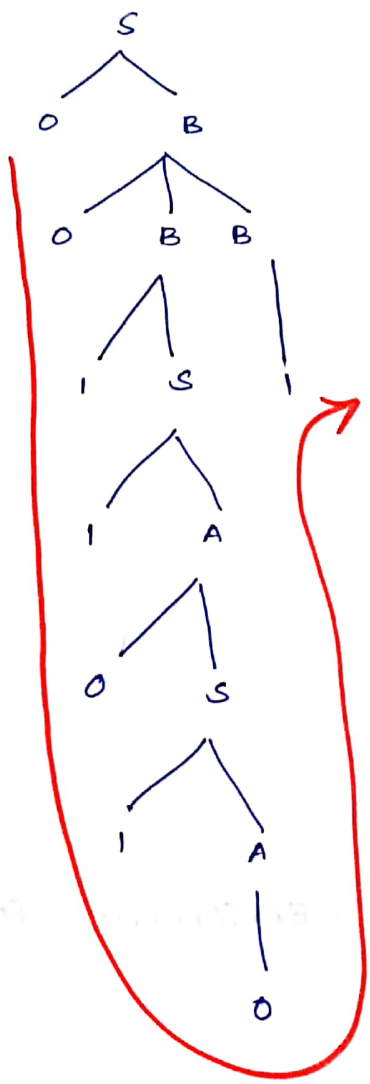
5   4

00110101

(a) Leftmost Derivation.

S → 0B

S → 00BB	B → 0BB
S → 001SB	B → 1S
	S → 1A
S → 0011AB	A → 0S
S → 00110SB	S → 1A
S → 001101AB	A → 0
S → 0011010B	B → 1
$\boxed{S → 00110101}$	



(b) Right most Derivation.

S → 0B	B → 0BB
→ 00BB	
	B → 1S
→ 00B1S	
	S → 0B
→ 00B10B	
→ 00B101S	B → 1S
→ 00B1010B	S → 0B
	B → 1
→ 00B10101	
	B → 1
$\boxed{S → 00110101}$	

sentential forms.

$G = \{T, NT, P, S\}$

$A \to \beta$

any strings from Terminals and Non-Terminals.

$a_1, \ldots\ldots\ a_m$.

$a_1 \overset{*}{\underset{G}{\Rightarrow}} a_m$.

$*$ – no of Derivations steps.

$L \Rightarrow W = abb$

$W^R = bba$.

$L = abbbba$.

$$\to A \xrightarrow{ab} B \xrightarrow{bb} C \xrightarrow{ba} (D)$$

$A \to aBB$.

$B \to bbC$

$C \to baD$

$D \to \epsilon$.

abbbba.

$A \to a\underline{B}B$.

$\to abb\underline{C}B$

$\to abbbaDB$.

SENTENTIAL FORMS:

G consists (V, T, P, S) be the Context free grammar.

If $A \to \beta$

β contains Terminals and Non Terminals includin(g) e.

$$A \overset{*}{\Rightarrow} \beta$$

$*$ → No of Derivation steps. Used.

Construct CFG for the language L which has all the strings which are all palindrome over $\Sigma = \{a, b\}^*$. (May 12)

or

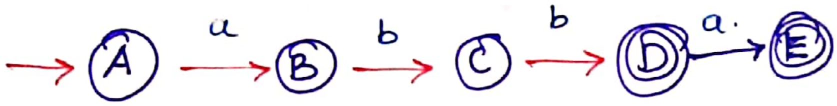
Define CFG. Obtain CFG for the following language.

(i) $L = \{WW^R \mid W \text{ is in } (a,b)^*, W^R \text{ is the Reversal of } W\}$

$W = aba$ (or) $W = ab$

$W^R = aba$ $\quad$ $W^R = ba$

$L = \{WW^R \rightarrow abba$



$A \rightarrow aB$

$B \rightarrow bC$

$C \rightarrow bD$

$D \rightarrow aE$

$E \rightarrow e.$

$A \rightarrow a\underline{B}$ $\qquad$ $B \rightarrow bC$

$\rightarrow ab\underline{C}$ $\qquad$ $C \rightarrow bD$

$\rightarrow abb\underline{D}$ $\qquad$ $D \rightarrow aE$

$\rightarrow abbaE$ $\qquad$ $E \rightarrow e$

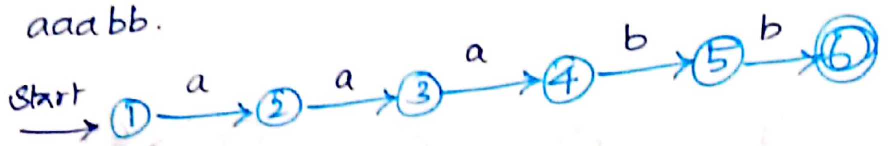
$\rightarrow abbae$

$\boxed{A \rightarrow abba}$

* obtain a CFG to generate unequal number of a's and b's.

1) a no 'b'
2) aab
3) aaabb $\Rightarrow$ CFG.
4) ba
5) bba
6) bbbaa.

aaabb.



$1 \to a2$

$2 \to a3$

$3 \to a4$

$4 \to b5$

$5 \to b6$

$6 \to \epsilon$

$1 \to a\underline{2}$

$\to aa\underline{3}$ $\because 2 \to a3$

$\to aaa\underline{4}$ $\because 3 \to a4$

$\to aaab\underline{5}$ $\because 4 \to b5$

$\to aaabb6$ $\because 5 \to b6$

$\to aaabb\epsilon$ $\because 6 \to \epsilon$.

$\boxed{\to aaabb}$

Terminal $\{a, b\}$

Non-Terminal $\to \{1, 2, 3, 4, 5, 6\}$

$P \to \{1, 2, 3, 4, 5, 6\}$

$S \to 1$

If $G = \{\{S\}, \{0, 1\}, \{S \to 0S1, S \to e\}, S\}$

Find $L(G)$. (Dec - 09, 12)

$S \to 0S1 \mid e$

$S \to 0\underline{S}1$ $\quad$ $S \to 0S1$

$\to 00\underline{S}11$ $\quad$ $S \to 0S1$.

$\to 000S111$ $\quad$ $S \to 0S1$

$\to 0000S1111$

$\vdots$ $\quad$ $S \to \epsilon$.

$\to 0^n 1^n$

$L(G) \Rightarrow \therefore$ any no of 0's followed by any no of 1's.

Define Context Free grammar and write Context Free grammar for the Language,

(i) $L = \{ a^i b^j c^k \mid i + j = k,\ i >= 0,\ j \geq 0$

(ii) $L = \{ a^n b^m c^k \mid n + 2m = k. \}$

$L = \{ a^i b^j c^{i+j}$

$i = 1$

$= a^1 b^1 c^{1+1}$

$\to A \xrightarrow{a} B \xrightarrow{b} C \xrightarrow{c} D \xrightarrow{c} E$

$= abcc$

$= abcc$

$A \to aB$	$A \to a\underline{B}$	$B \to bC$
$B \to bC$	$\to ab\underline{C}$	
$C \to cD$	$\to ab$	$C \to cD$
$D \to cE$	$\to abc\underline{D}$	
$E \to e$	$\to abcc\underline{E}$	$D \to cE$
	$\boxed{A \to abcc}$	$E \to e$

$L = \{ a^n b^m c^k \mid n + 2m = k$

$n = 1$

$= a^1 b^1 c^{1+2(1)}$

$c^3 \Rightarrow ccc$

$= a\, b\, c^{1+2}$

$\boxed{L = abccc}$

$\to A \xrightarrow{a} B \xrightarrow{b} C \xrightarrow{ccc} D$

$A \to aB$

$B \to bC$

$C \to cccD$

$D \to e.$

$A \to a\underline{B}$

$\to ab\underline{C}$

$\to abccc\underline{D}$

$\to abccce$

$\boxed{A \to abccc}$

PARSE TREE

* Graphical Representation in which the Derivation of the given Production Rules is present.

* To show how the derivation can be done to obtain some string from given set of Production Rules.

* The derivation Tree is also called Parse Tree.

Properties of Parse Tree :

1. Root Node → Start symbol
2. Derivation is Read from left to Right.
3. The leaf nodes are always Terminal Nodes.
4. The interior Nodes are always the Non-Terminal Nodes.

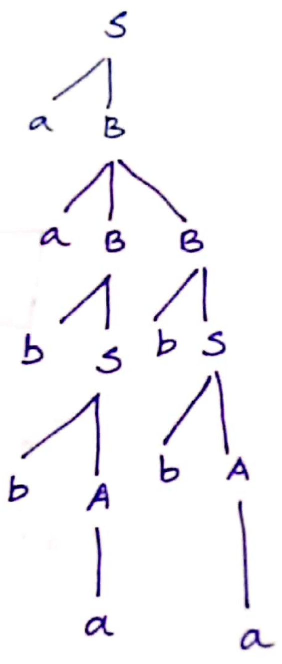
Ex:

Construct the derivation Tree for the String 'aabbabba' from the context free grammar given by,

$S \to aB \mid bA$

$A \to a \mid aS \mid bAA$

$B \to b \mid bS \mid aBB$



$S \to a\underline{B}$	
$\to aa\underline{B}B$	$S \to aBB$
$\to aab\underline{S}B$	$B \to bS$
$\to aabb\underline{A}B$	$S \to bA$
	$A \to a$
$\to aabba\underline{B}$	
	$B \to bS$
$\to aabbab\underline{S}$	$S \to bA$
$\to aabbabb\underline{A}$	$A \to a$
$\to aabbabba$	

Left most Derivation Tree. for "aabbabba"

Scanned with CamScanner

Ambiguity in Grammar and Languages.

Ambiguous grammar is a grammar using which if it is possible to construct more than one parse Trees for the same string.

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

String ⇒ id * id + id

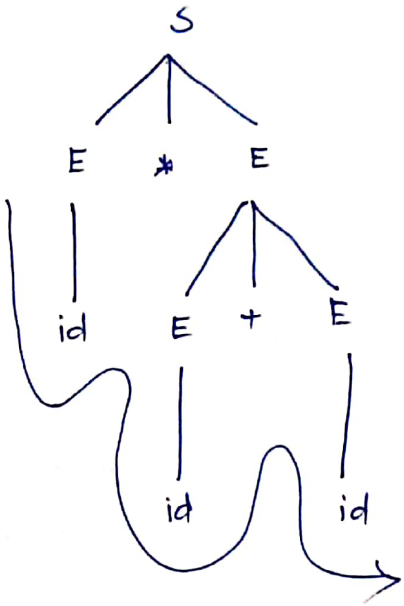
$S \rightarrow E * E$ $E \rightarrow id$

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

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

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

$\boxed{S \rightarrow id * id + id}$



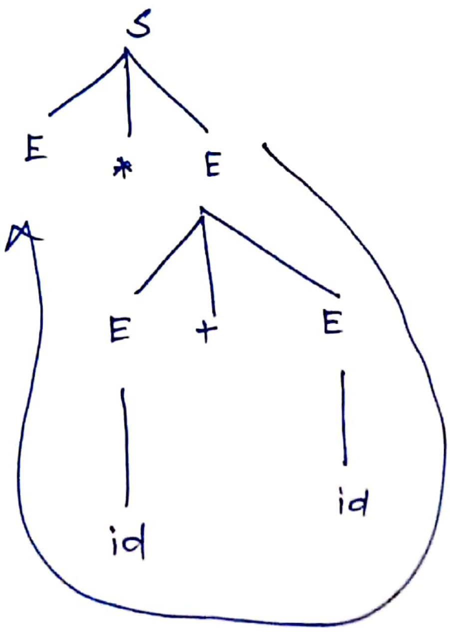
$S \rightarrow E * E$

$\rightarrow E * E + E$

$\rightarrow E * E + id$

$\rightarrow E * id + id$

$\boxed{S \rightarrow id * id + id}$



Unit-3

Sentential Form.

$G = \{ T, N.T, P, S \}$

$\alpha \xrightarrow{*} \beta$

$\alpha \rightarrow$ Non-Terminal.
$\beta \rightarrow$ Terminal with Non Terminal.
$\beta' \rightarrow$ strings.
$* \rightarrow$ no of Derivation steps.

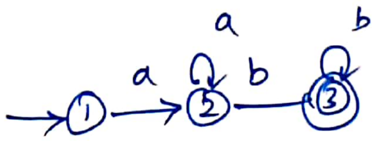
Ambiguity in Grammar and Languages.

Deriving more then one parse Tree in CFL. is Ambiguous.

Ex: When do you say a Language L is unambiguous? Show that the Language.

$L = \{ a^n b^n \mid n >= 1 \}$ is unambiguous.

$L = a^n b^n$



aabb.

$1 \rightarrow 2a \rightarrow a2$
$\rightarrow 2aa \rightarrow a2a$
$\rightarrow$

$1 \rightarrow 2a$
$2 \rightarrow 2a \mid 3b$
$3 \rightarrow 3b \mid \epsilon$

$S \rightarrow aS \mid e$
$B \rightarrow e \mid Bb.$

~~From,~~

~~Dr. V. Sathiyasuntharam~~
~~Professor, CSE Department,~~
~~CMR Engineering College,~~

G_1	G_2
a b f	c d e

$\delta(a,0) = b$
$\delta(b,0) = a$ $\rangle G_1$
$\delta(f,0) = f$

$\delta(a,1) = c$
$\delta(b,1) = d$ $\rangle G_2$
$\delta(f,1) = f. \to G_1$

g

G_1	G_2	G_3
a b	f	c d e

$\delta(a,0) = b$
$\delta(b,0) = a$ $\rangle G_1$

$\delta(a,1) = c$
$\delta(b,1) = d$ $\rangle G_3$

$\delta(f,0) = f$
$\delta(f,1) = f$

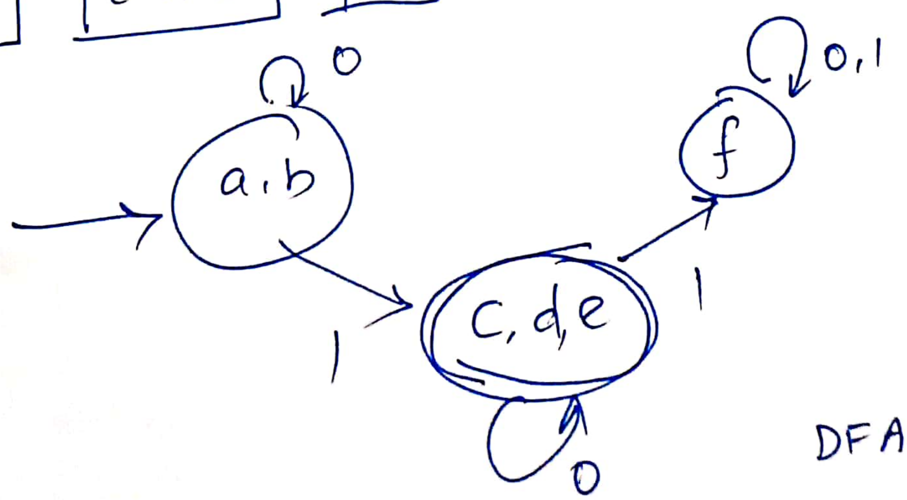
$\delta(c,0) = e$
$\delta(d,0) = e$ $\rangle G_3$
$\delta(e,0) = e$

$\delta(c,1) = f$
$\delta(d,1) = f$ $\rangle \to G_2$
$\delta(e,1) = f$

all are moving into Same Group..

∴ So no partition required.

a, b	c d e	f



DFA.

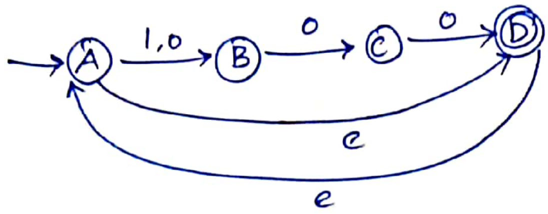
c. write about closure properties of Regular language.

4. a. write the steps in minimization of FA.

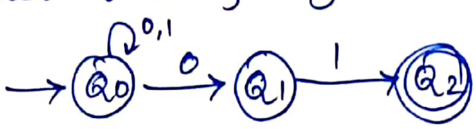
b. construct the minimum State Automata for the Following

	a	b
→A	B	A
B	A	C
C	D	B
D	D	A
E	D	F
F	G	E
G	F	G
H	G	D

c. Construct NFA for given NFA with e-moves



5. a. Construct DFA for given NFA.



b. Show that $L = \{a^n b^n \mid n \geq 0\}$ is not Regular.

c. If a Regular grammar G is given by $S \to aS \mid a$

Find DFA (M) accepting L(G)?

d. Construct a Regular grammar for $L = \{0^n 11 \mid n >= 1\}$

Submission Date: 15-2-19.

1st Assignment Questions.

1. a. Differentiate between NFA and DFA

b. Design DFA for the Following over {a, b}

i) All strings containing not more than three a's.

ii) All strings that has at least two occurrences of b between any two occurrences of a.

iii) Accepting the set of All strings ending with 00?

2. i. Define Regular Expressions? Explain about the properties of Regular Expressions.

ii. Construct a DFA for the Regular Expression $(0+1)^* (00+11) (0+1)^*$

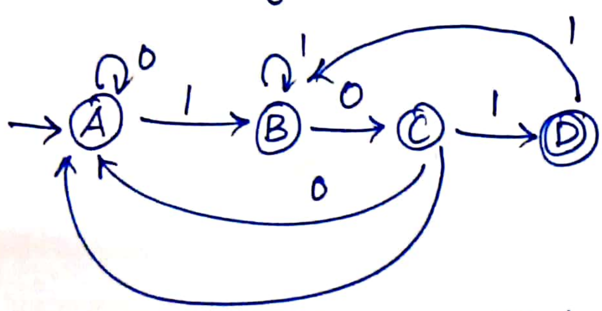
(iii) Design a FA for the Following languages.

(a) $(0^* 1^*)^*$

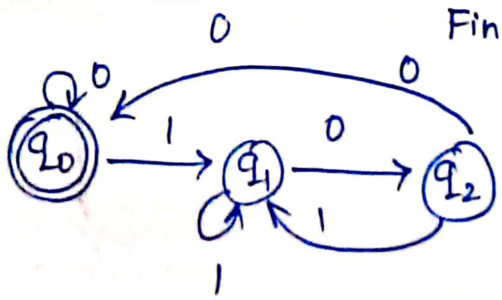
(b) $(0+1)^* 111^*$

(c) $(0^* 11^* + 101)$

3. a. Obtain a regular Expression for the Following FA.



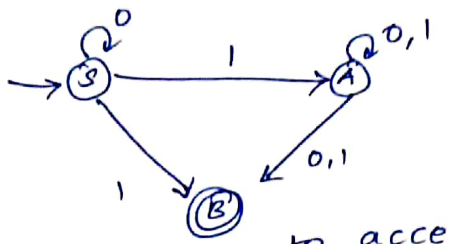
b. Find out Regular Expression from given DFA.



Regular grammar into FA.

(3) $S \to 0S \mid 1A \mid 1$

$A \to 0A \mid 1A \mid 0 \mid 1$

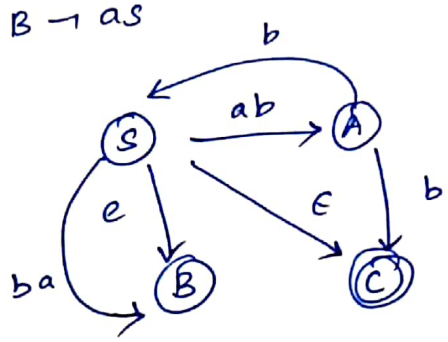


(4) Trace the Transition to accept the string 'abba'

$S \to abA \mid B \mid baB \mid \epsilon$

$A \to bS \mid b$

$B \to aS$



Right linear grammar.

$A \to aB$

$A \to a$

$A \to e$

Left linear.

$A \to Ba$

$A \to a$

$A \to e$.

$L = a^n b^m \quad n \geq 2, m >= 3 \}$

$= a^3 b^3$

$=$ aaa bbb.

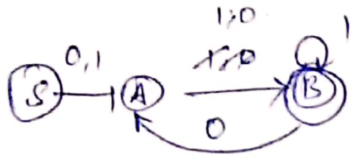
①

S → 01A

A → 10B

B → 0A | 11

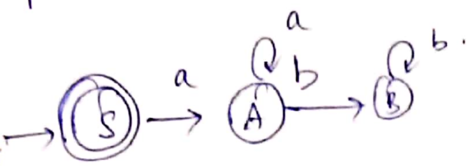
δ(S, 0,1) = A)



S → 01A

A → 1,0B

B → 0A | 11

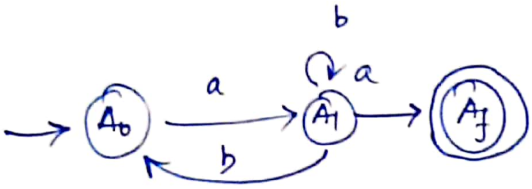


① Regular grammar

TO FA.

$A_0 \rightarrow aA_1$

$A_1 \rightarrow bA_1 \mid a \mid bA_0$

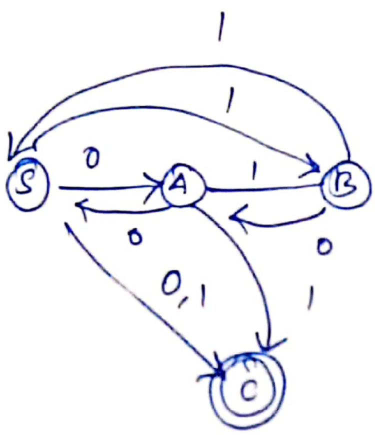


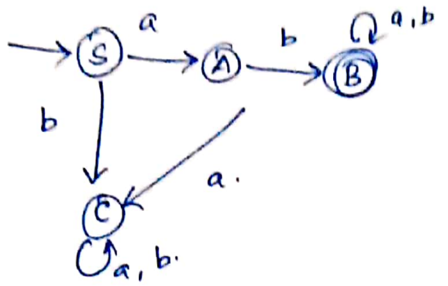
②

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

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

$B \rightarrow 0A \mid 1S$





$S \to aA \mid bB \mid bC$

$A \to bB \mid aC \mid$

$B \to aB \mid bB \mid \epsilon$.

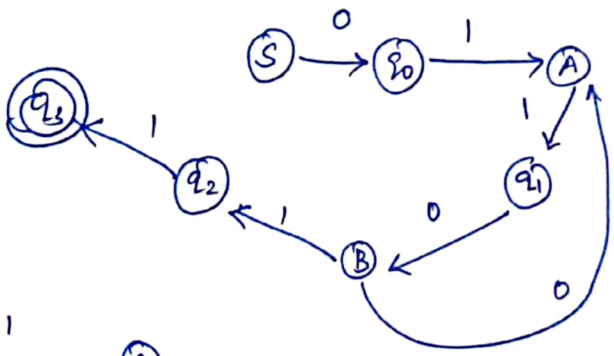
$C \to aC \mid bC$.

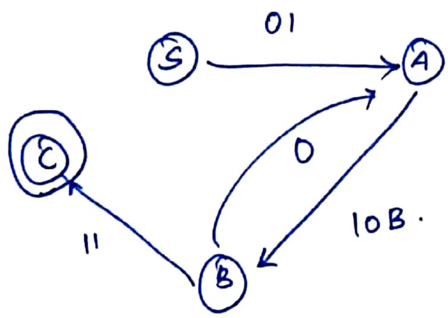
Construct DFA to accept the language generated by the Following grammar.

$S \to 01A$

$A \to 10B$

$B \to 0A \mid 11$

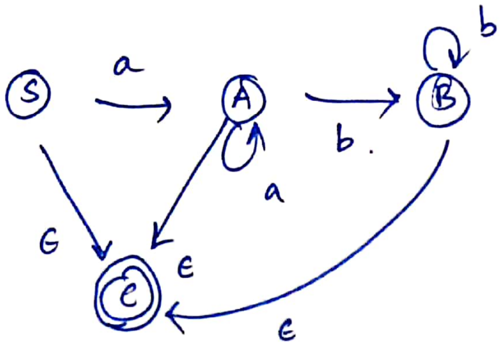




$S \to aA \mid \epsilon$

$A \to aA \mid bB \mid \epsilon$

$B \to bB \mid \epsilon$

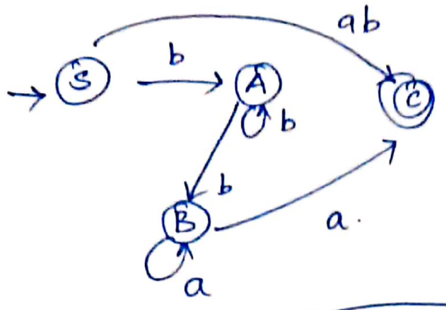


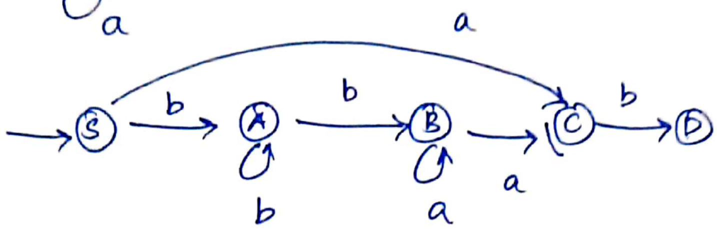
Construct NFA for the Following grammar,

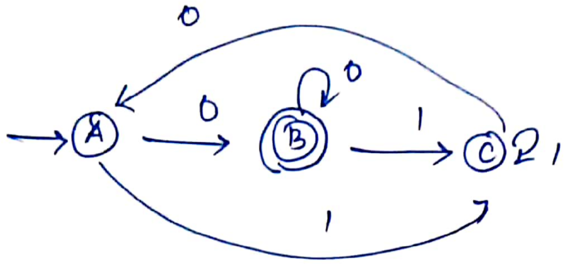
$S \to \cancel{a}\cancel{A}\, Ab \mid ab$

$A \to Ab \mid Bb$

$B \to Ba \mid a$





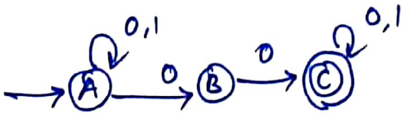


$A \to 0B \mid 1C$

$B \to 0B \mid 1C$

$C \to 0A \mid 1C.$

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



$(A,0) = A$

$(A,1) = A$

$(A,0) = B$

$(B,0) = C$

$A \to A0 \mid A1 \mid B0$

$B \to B0$

$C \to C0 \mid C1$