UNIT- II

PROPERTIES OF CONTEXT-FREE LANGUAGES.

Normal Forms for CFG- Pumping Lemma for CFL- Closure Properties of CFL- Turing Machine - Programming Technique for TM.

Normal Forms of Context Free Grammars:

Every CFL is generated by a CFG in which all productions are of the form

$$A \rightarrow BC$$
$$A \rightarrow a$$

where, A, B, C - Variables
a - Terminals.

This form of CFG is called as Chomsky Normal Form.

Inorder to find CNF, we need to perform the following operations.

* Eliminate Useless Symbols (unreachable symbols).
* Eliminate Unit Production ($A \rightarrow B$)
* Eliminate ϵ- production ($A \rightarrow \epsilon$).

(Only one variable in Right side)

The variables (or) terminals that do not appear in any derivation

Eliminating Useless Symbols:

1. Consider the grammar

$S \rightarrow AB \mid a$

$A \rightarrow b$

① B does not generate any string. So Replace B production & its Combination.

$S \rightarrow a$

$A \rightarrow b$

② As A is not reachable.

$S \rightarrow a$

2. $S \rightarrow AB \mid CA$

$B \rightarrow BC \mid AB$

$A \rightarrow a$

$C \rightarrow aB \mid b$

① Check whether the String is generated, B is not generate any string. Replace B.

$S \rightarrow CA$

$A \rightarrow a$

$C \rightarrow b$

Eliminating ε-production:

$S \to AB$
$A \to aAA \mid \epsilon$
$B \to bBB \mid \epsilon$

$S \to AB$
$S \to A$ [$B = \epsilon$)
$S \to B$ [$A = \epsilon$]

$B \to bBB$
$B \to bB$ [$B \to \epsilon$]
$B \to bB$ [$B \to \epsilon$]
$B \to b$ [$BB \to \epsilon$]

$A \to aAA$
$A \to aA$ [$A = \epsilon$] $A \to aA$ [$A \to \epsilon$]
$A \to a$ [$AA = \epsilon$]

$S \to AB \mid A \mid B$
$A \to aAA \mid aA \mid a$
$B \to bBB \mid bB \mid b.$

Eliminating unit Productions:

$S \to 0A \mid 1B \mid 011$
$A \to 0S \mid 00$
$B \to 1 \mid A$
$C \to 01$

C is not in Start S.

$Z \to 0$
$W \to 1$

$S \to ZA \mid WB \mid ZWW$ Sub, if more than one symbol.
$A \to ZS \mid ZZ$
$B \to 1 \mid A$
$Z \to 0$
$W \to 1$

CFG to CNF:

$A \to BC$ (or) $NT \to NT\ NT$

$A \to a$ $NT \to T$.

(MAY/JUNE 2014)

$S \to bA \mid aB$

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

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

There is no ε- & unit productions & useless production.

$S \to bA$	$A \to bAA$	$B \to aBB$
$S \to D_bA$	$A \to D_1A$	$B \to D_2B$
$D_b \to b$	$D_1 \to bA$	$D_2 \to aB$
$S \to aB$	$D_1 \to D_bA$	$D_2 \to D_aB$
$S \to D_aB$	$A \to aS$	$B \to bS$
$D_a \to a$	$A \to D_aS$	$B \to D_bS$
	$A \to a$	$B \to b$

$S \to D_bA \mid D_aB$

$A \to D_1A \mid D_aS \mid a$

$B \to D_2B \mid D_bS \mid b$.

$D_a \to a$

$D_b \to b$

$D_1 \to D_bA$

$D_2 \to D_aB$.

2. $S \rightarrow 0A \mid 1B \mid 011$

$A \rightarrow 0S \mid 00$

$B \rightarrow 1 \mid 0S \mid 00$

$C \rightarrow 01$

Soln:

In this there is a unit Production. C

∴ Eliminate the unit production by introducing new Variable.

$Z \rightarrow 0$

$W \rightarrow 1$

Replace in the grammar.

$S \rightarrow ZA \mid WB \mid \underline{ZWW}$

$A \rightarrow ZS \mid ZZ$

$B \rightarrow 1 \mid ZS \mid ZZ$

$C \rightarrow ZW$

$Z \rightarrow 0$

$W \rightarrow 1$

Break the bodies of length having 3 (or) more.

$S \rightarrow ZWW$

$A_1 \rightarrow ZW$

∴ $S \rightarrow ZA \mid WB \mid A_1W$

$A \rightarrow ZS \mid ZZ$

$B \rightarrow 1 \mid ZS \mid ZZ$

$C \rightarrow ZW$

$Z \rightarrow 0$

$W \rightarrow 1$

$A_1 \rightarrow ZW$ //

3. $S \to AAC$

$A \to aAb \mid \epsilon$

$C \to aC \mid a.$

Soln:

Step 1: Eliminate ϵ- production

$S \to AAC$ $A \to aAb$

$S \to AC$ $[\because A \to \epsilon]$ $A \to ab$ $[\because A \to \epsilon]$

$S \to C$ $[\because AA \to \epsilon]$

$S \to AAC \mid AC \mid C$

$A \to aAb \mid ab$

$C \to aC \mid a$

Step 2: Eliminate Unit production.

As There is no string terminal for A, remove A:

No Unit production & No useless Symbol.

$S \to AAC \mid AC \mid aC \mid a$

$A \to aAb \mid ab$

$C \to aC \mid a.$

Check for CNF, and convert into CNF.

$S \to AAC \mid AC \mid aC \mid a$

$A \to aAb \mid ab$

$C \to aC \mid a.$

$S \to AAC$

$\boxed{S \to D_1 C}$

$\boxed{D_1 \to AA}$

$S \to AC$

$S \to aC$

$\boxed{S \to D_a C}$

$\boxed{D_a \to a}$

$\boxed{S \to a}$

$A \to aAb$

$A \to D_a A b$

$\boxed{A \to D_a D_2}$

$D_2 \to Ab$

$D_2 \to A D_b$

$\boxed{D_b \to b}$

$A \to ab$

$\boxed{A \to D_a D_b}$

$C \to aC$

$\boxed{C \to D_a C}$

$\boxed{C \to a}$

$S \to D_1 C \mid AC \mid D_a C \mid a$

$A \to D_a D_2 \mid D_a D_b$

$C \to D_a C \mid a$

$D_1 \to AA$

$D_2 \to A D_b$

$D_a \to a$

$D_b \to b$.

(NOV | DEC 2016)

Solve 4. $S \to aAbB$

$A \to aA \mid a$

$B \to bB \mid B$

Soln: There is no ϵ- Unit production & useless Symbol.

Check for CNF.

$A \to a$ $B \to b$ are already in CNF.

$S \to aAbB$

$S \to D_a A D_b B$

$\boxed{S \to D_a D_1}$

$D_1 \to A D_2$

$D_2 \to D_b B$

$D_a \to a$

$D_b \to b$

$A \to aA$

$\boxed{A \to D_a A}$

$B \to bB$

$\boxed{B \to D_b B}$

$\therefore$ $S \to D_a D_1$

$D_1 \to A D_2$

$D_2 \to D_b B$

$A \to D_a A \mid a$

$B \to D_b B \mid b$

$D_a \to a$

$D_b \to b$

5\. $S \to aAa \mid bBb \mid \epsilon$

Ans $A \to C \mid a$

$B \to C \mid b$

$C \to CDE \mid \epsilon$

$D \to A \mid B \mid ab$

Soln: Eliminate ϵ-production

$C \to \epsilon$. is the ϵ- production

$\therefore A \to \epsilon,\ B \to \epsilon,\ D \to \epsilon$

$S \to aAa \mid bBb \mid aa \mid bb$

$A \to C \mid a$

$B \to C \mid b$

$C \to CDE \mid DE \mid CE \mid E$ $\quad D \to A \mid B \mid ab$

Eliminate Unit Production :

C is the Unit production.

$A \to C \mid a$

$A \to CDE \mid DE \mid CE \mid a$

$B \to C \mid b$

$B \to CDE \mid CE \mid DE \mid b$

$C \to CE \mid DE \mid CDE \mid E$

$C \to CE \mid DE \mid CDE$.

$D \to A \mid B \mid ab$

$D \to a \mid b \mid ab$ $[\because A \to a,\ B \to b]$

$\therefore S \to aAa \mid aa \mid bBb \mid bb$

$A \to a \mid CDE \mid DE \mid CE$

$B \to b \mid CDE \mid CE \mid DE$

$C \to CE \mid DE \mid CDE$

$D \to a \mid b \mid ab$

Eliminate Useless Symbol:

C, E.

$S \to aAa \mid aa \mid bBb \mid bb$

$A \to a$

$B \to b$

$D \to a \mid b \mid ab$.

Eliminate 'D' as the Start S does not Contain D.

$\therefore S \to aAa \mid aa \mid bBb \mid bb$

$A \to a,\ B \to b$.

$S \to aAa \mid bBb \mid aa \mid bb$

$A \to a,$

$B \to b$

$S \to aAa$ $\quad$ $S \to bBb$ $\quad$ $S \to aa$ $\quad$ $S \to bb$

$S \to D_a A a$ $\quad$ $S \to D_b B b$ $\quad$ $S \to D_a D_a$ $\quad$ $S \to D_b D_b$

$\boxed{S \to D_a D_1}$ $\quad$ $\boxed{S \to D_b D_2}$

$\boxed{D_1 \to A D_a}$ $\quad$ $\boxed{D_2 \to B D_b}$

$D_a \to a$ $\quad$ $D_b \to b$

$\therefore$ $S \to D_a D_1 \mid D_a D_a \mid D_b D_b$

$S \to D_b D_2$

$D_1 \to A D_a$

$D_2 \to B D_b$

$D_a \to a$

$D_b \to b$

$A \to a$

$B \to b$ //

Greibach Normal Form:

Every Context-free language L without ε can be generated by a grammar for which every production is of the form $A \to a\alpha$, where A is a Variable, a is the terminal and α is a string of Variable.

$$GNF: NT \to T.NT$$
$$NT \to T$$

Conditions:

① $A_K \to A_j\alpha$ is a production with $j < k$, generate a new set of production by Substituting for A_j until $j \geq k$.

$A_j \to \beta$ add production $A_K \to \beta\alpha$ and remove $A_K \to A_j\alpha$

② $A_K \to A_K\alpha$, add productions $B_K \to \alpha$ and $B_K \to \alpha B_K$ and remove $A_K \to A_K\alpha$.

③ $A_K \to \beta$, where β does not begin with A_K, add production $A_K \to \beta B_K$.

Convert the following production into GNF:

$S \to AA \mid a$

$A \to SS \mid b$

Soln: The Given production are in CNF

To Convert into GNF,

Replace S by A_1 and A by A_2

$$\underset{k}{A_1} \to \underset{j}{A_2}A_2 \mid a \quad — ①$$

$$\underset{k}{A_2} \to \underset{j}{A_1}A_1 \mid b. \quad — ②$$

step 1: In (1) $j > k$, no need to change

In (2) $j < k$, $\therefore$ Sub A_1 in (2)

$A_1 \to A_2 A_2 \mid a$

$A_2 \to A_1 A_1 \mid b$

$A_2 \to A_2 A_2 A_1 \mid \overset{a}{\cancel{b}} A_1 \mid b$

$\therefore$

$A_1 \to A_2 A_2 \mid a$

$A_2 \to A_2 A_2 A_1 \mid a A_1 \mid b$

step 2:

$$A_2 \to \underbrace{A_2}_{A_K} \underbrace{A_2 A_1}_{\alpha} \mid \underbrace{a A_1 \mid b}_{\beta}$$

Introduce new symbol B_2

$B_2 = A_2 A_1 \mid A_2 A_1 B_2$

$\therefore$

$A_1 \to A_2 A_2 \mid a$

$A_2 \to a A_1 \mid b$

$B_2 \to A_2 A_1 \mid A_2 A_1 B_2$

step 3:

$A_2 \to a A_1 \mid b$

For each production $A_K \to \beta$ add

$A_K \to \beta \mid \beta B_K$

$A_2 \to a A_1 \mid b \mid a A_1 B_2 \mid b B_2$

$\therefore$ The productions are,

$A_1 \to A_2 A_2 \mid a$

$A_2 \to a A_1 \mid b \mid a A_1 B_2 \mid b B_2$

$B_2 \to A_2 A_1 \mid A_2 A_1 B_2$

$A_1 \rightarrow A_2 A_2 \mid a$ Sub A_2 in (A_1)

$A_1 \rightarrow aA_1A_2 \mid bA_2 \mid aA_1B_2A_2 \mid bB_2A_2 \mid a$

$A_2 \rightarrow aA_1 \mid b \mid aA_1B_2 \mid bB_2$

$B_2 \rightarrow aA_1A_1 \mid bA_1 \mid aA_1B_2A_1 \mid bB_2A_1 \mid aA_1A_1B_2$

$aA_1B_2A_1B_2 \mid bA_1B_2 \mid bB_2A_1B_2.$

$\therefore$ The Equivalent GNF is given by,

$A_1 \rightarrow aA_1A_2 \mid bA_2 \mid aA_1B_2A_2 \mid bB_2A_2 \mid a$

$A_2 \rightarrow aA_1 \mid b \mid aA_1B_2 \mid bB_2$

$A_3 \rightarrow aA_1A_1 \mid bA_1 \mid aA_1B_2A_1 \mid bB_2A_1 \mid aA_1A_1B_2$

$aA_1B_2A_1B_2 \mid bA_1B_2 \mid bB_2A_1B_2.$

2. (NOV/DEC 2014)

$A_1 \rightarrow A_2A_3$		$S \rightarrow AB$	$A_1 \rightarrow S$
$A_2 \rightarrow A_3A_1 \mid b$	(or)	$A \rightarrow BS \mid b$	$A_2 \rightarrow A$
$A_3 \rightarrow A_1A_2 \mid a$		$B \rightarrow SA \mid a$	$A_3 \rightarrow B$

Soln:

$\underset{k}{A_1} \rightarrow \underset{j}{A_2} A_3$ —— (1)

$\underset{k}{A_2} \rightarrow \underset{j}{A_3} A_1 \mid b$ —— (2)

$\underset{k}{A_3} \rightarrow \underset{j}{A_1} A_2 \mid a$ —— (3)

step 1: In (1) & (2) $j > k$ $\therefore$ no need to change.

In (3) $j < k$ $\therefore$ modify

$A_1 \rightarrow A_2 A_3$

$A_2 \rightarrow A_3 A_1 \mid b$

$A_3 \rightarrow A_1 A_2 \mid a$

$A_3 \rightarrow A_2 A_3 A_2 \mid a$

$A_3 \rightarrow A_3 A_1 A_3 A_2 \mid bA_3 A_2 \mid a$

$\bar{k} \quad \bar{j}$

$\therefore$

$A_1 \rightarrow A_2 A_3$

$A_2 \rightarrow A_3 A_1 \mid b$

$A_3 \rightarrow \underline{A_3} \underbrace{A_1 A_3 A_2}_{\alpha} \mid \underbrace{b A_3 A_2 \mid a}_{\beta}$

Step 2:

$\underbrace{A_3}_{A_k} \rightarrow \underbrace{A_3}_{A_k} \underbrace{A_1 A_3 A_2}_{\alpha}$

$A_k \rightarrow A_k \alpha$, then

Introduce new Symbol B_3

$B_k \rightarrow \alpha \mid \alpha B_k$

$B_3 \rightarrow A_1 A_3 A_2 \mid A_1 A_3 A_2 B_3$

$\therefore$ The resulting Set is

$A_1 \rightarrow A_2 A_3$

$A_2 \rightarrow A_3 A_1 \mid b$

$A_3 \rightarrow bA_3 A_2 \mid a$

$B_3 \rightarrow A_1 A_3 A_2 \mid A_1 A_3 A_2 B_3$

Step 3: For each production $A_k \to \beta$ add $A_k \to \beta | \beta B_k$

$A_3 \to bA_3A_2 | a$

$A_3 \to bA_3A_2 | a | bA_3A_2B_3 | aB_3$.

$\therefore$ The productions are

$A_1 \to A_2A_3$

$A_2 \to A_3A_1 | b$

$A_3 \to bA_3A_2 | a | bA_3A_2B_3 | aB_3$

$B_3 \to A_1A_3A_2 | A_1A_3A_2B_3$.

By, Substituting A_3.

$A_1 \to A_2A_3$

$A_1 \to A_3A_1A_3 | bA_3$

$A_1 \to bA_3A_2A_1A_3 | aA_1A_3 | bA_3A_2B_3A_1A_3 | aB_3A_1A_3$
$| bA_3$

$A_2 \to A_3A_1 | b$

$A_2 \to bA_3A_2A_1 | aA_1 | bA_3A_2B_3 | aB_3A_1 | b$

$B_3 \to A_1A_3A_2 | A_1A_3A_2B_3$

$\to bA_3A_2A_1A_3A_3A_2 | aA_1A_3A_3A_2 | bA_3A_2B_3A_1A_3A_3A_2$
$| aB_3A_1A_3A_3A_2 | bA_3A_3A_2$

$bA_3A_2A_1A_3A_3A_2B_3 | aA_1A_3A_2B_3 | bA_3A_2B_3A_3A_2B_3,$
$aB_3A_1A_3A_2B_3 | bA_3A_2B_3$.

Solve, $S \to AA \mid 0$ $\quad$ $S \to aSb$

$A \to SS \mid 1.$ $\quad$ $S \to ab$

Soln:

$S \to aSb$

$S \to ab$

There is no Unit production, ϵ- production.

Convert into CNF.

$S \to aSb$

$S \to ab$

$S \to D_a S D_b$ $\quad$ $S \to ab$

$\boxed{S \to D_a D_1}$ $\quad$ $\boxed{S \to D_a D_b}$

$D_1 \to S D_b$ $\quad$ $D_a \to a, D_b \to b$

$S \to D_a D_1$

$S \to D_a D_b$

$S \to A_1$, $D_a \to A_2$ $D_b \to A_3$ $D_1 \to A_4$

$A_1 \to A_2 A_4$

$A_1 \to A_2 A_3$

$A_2 \to a$

$A_3 \to b$

$A_1 \to a A_4$

$A_1 \to a A_3$

$A_2 \to a$

$A_3 \to b$

$\therefore$ $A_1 \to aA_4 \mid aA_3$

$A_2 \to a$

$A_3 \to b.$

Pumping Lemma For CFL:

Let L be any CFL, Then there is a Constant n, depending only on L, such that if z is in L and $|z| \geq n$, then we can write $z = uvwxy$ such that

i) $|vx| \geq 1$

ii) $|vwx| \leq n$

iii) $uv^i wx^i y \in L$, for all $i \geq 0$.

Application:

Used to show that a Language L is not Context free.

1. Show that $L = \{a^i b^i c^i \mid i \geq \phi\}$ is not Context Free.

Soln: $i = 1$ $\quad$ $i = 2$

$L = abc$ $\quad$ $L = aabbcc$

$z = abc$

$u = ab$

$v = \epsilon$

$w = \epsilon$

$x = c$

$y = \epsilon$.

i) $|vx| \geq 1$

$|\epsilon c| \geq 1$

$|c| \geq 1$

$\therefore$ It is true.

ii) $|vwx| \leq n$

$|\epsilon \epsilon c| \leq 1$

$|c| \leq 1$

$\therefore$ It is true.

iii) $uv^i wx^i y$

$ab\epsilon^0 \epsilon c^0 \epsilon$

$abc^0 \Rightarrow ab \notin L$ $\therefore$ L is not Context-Free.

8. Show that $L = \{a^p \mid p \text{ is prime}\}$ is not a CFL. (NOV/DEC)

Soln: prime = 2, 3, 5, 7...

$P = 2$

$L = a^2 = aa \quad a^3 = aaa \ldots$

$n = 2$

$L = aa$

$z = aa$

$u = a$

$v = \epsilon$

$w = \epsilon$

$x = a$

$y = \epsilon$

i) $|vx| \geq 1$

$|a| \geq 1$

$\therefore$ It is true

ii) $|vwx| \leq n$

$|\epsilon \epsilon a| \leq 2$

$|a| \leq 2$

$\therefore$ It is true.

iii) uv^iwx^iy

$= a\epsilon^0\epsilon a^0\epsilon \quad i = 0$

$= a$

$= a \notin L$.

$\therefore$ It is not a CFL.

3. Solve $L = \{ 0^{2^i} \mid i \geq 1 \}$, $L = \{ a^n b^n c^n \mid n \geq 1 \}$

4. Show that $L = \{ a^n b^n c^n \mid n \geq 1 \}$ is not Context Free.

5. " " $L = \{ a^k b^j c^k d^j \mid k \geq 1 \,\&\, j \geq 1 \}$ is Not CFG.

$L = \{ b^{2n} : n \geq 1 \}$

Closure Properties for CFL: (NOV/DEC 2017)

* CFL is closed under Union
* CFL is closed under Concatenation
* CFL is closed under Kleene closure.
* CFL is not closed under intersection.
* CFL is not closed under Complementation
* Every Regular Language is Context Free

Theorem 1:

CFL is closed Under Union.

Proof:

Given two CFG's G_1 and G_2 that generates the Language L_1 and L_2 respectively, we can Construct the CFG that generates $L_1 \cup L_2$

Let $G_1 = (V_1, \Sigma_1, P_1, S_1)$

$G_2 = (V_2, \Sigma_2, P_2, S_2)$

where S_1 & S_2 are the Starting Symbols of G_1 & G_2 respectively.

$L_1 \cup L_2$ is given by

$G = (V_1 \cup V_2, \Sigma_1 \cup \Sigma_2, P_3, S)$ where $S \rightarrow S_1$ or S_2.

2. Show that $L = \{a^p \mid p \text{ is prime}\}$ is not a CFL. (NOV/DEC)

Soln: prime = 2, 3, 5, 7...

$P = 2$

$L = a^2 = aa \quad a^3 = aaa \ldots$

$n = 2$

$L = aa$

$z = aa$

$u = a$

$v = \epsilon$

$w = \epsilon$

$x = a$

$y = \epsilon$

i) $|vx| \geq 1$

$|a| \geq 1$

∴ It is true

ii) $|vwx| \leq n$

$|\epsilon \epsilon a| \leq 2$

$|a| \leq 2$

∴ It is true.

iii) uv^iwx^iy

$= a \epsilon^0 \epsilon a^0 \epsilon \quad i = 0$

$= a$

$= a \notin L$.

∴ It is not a CFL.

3. Solve $L = \{ 0^{2^i} \mid i \geq 1 \}$, $L = \{ a^n b^n c^n \mid n \geq 1 \}$

4. Show that $L = \{ a^n b^n c^n \mid n \geq 1 \}$ is not Context Free.

5. " " $L = \{ a^k b^j c^k d^j \mid k \geq 1 \,\&\, j \geq 1 \}$ is Not CFG.

$L = \{ b^{2^n} : n \geq 1 \}$

Closure Properties for CFL: (NOV/DEC 2017)

* CFL is closed under Union
* CFL is closed under Concatenation
* CFL is closed under Kleene closure.
* CFL is not closed under intersection
* CFL is not closed under Complementation
* CFL is Context Free
* Every Regular Language is Context Free

Theorem 1:

CFL is closed under Union.

Proof:

Given two CFG's G_1 and G_2 that generates the Language L_1 and L_2 respectively, we can Construct the CFG that generates $L_1 \cup L_2$

Let $G_1 = (V_1, \Sigma_1, P_1, S_1)$

$G_2 = (V_2, \Sigma_2, P_2, S_2)$

where S_1 & S_2 are the Starting Symbols of G_1 & G_2 respectively.

$L_1 \cup L_2$. is given by

$G = (V_1 \cup V_2, \Sigma_1 \cup \Sigma_2, P_3, S)$ where $S \rightarrow S_1$ or S_2.

2. Show that $L = \{a^p \mid p \text{ is prime}\}$ is not a CFL.

(NOV/DEC 2017)

Soln: prime = 2, 3, 5, 7...

$P = 2$

$L = a^2 = aa \quad a^3 = aaa \ldots$

$n = 2$

$L = aa$

$z = aa$

$u = a$

$v = \epsilon$

$w = \epsilon$

$x = a$

$y = \epsilon$

i) $|vx| \geq 1$

$|a| \geq 1$

$\therefore$ It is true

ii) $|vwx| \leq n$

$|\epsilon \epsilon a| \leq 2$

$|a| \leq 2$

$\therefore$ It is true.

iii) uv^iwx^iy

$= a\epsilon^0\epsilon a^0\epsilon \quad i = 0$

$= a$

$= a \notin L$.

$\therefore$ It is not a CFL.

3. Solve $L = \{ 0^{2^i} \mid i \geq 1 \}$, $L = \{ a^n b^n c^n \mid n \geq 1 \}$

4. Show that $L = \{ a^n b^n c^n \mid n \geq 1 \}$ is not Context Free.

5. " " $L = \{ a^k b^j c^k d^j \mid k \geq 1 \,\&\, j \geq 1 \}$ is NOT CFG.

$L = \{ b^{2n} : n \geq 1 \}$

Closure Properties for CFL: (NOV/DEC 2017)

* CFL is closed under Union
* CFL is closed under Concatenation
* CFL is closed under Kleene closure.
* CFL is not closed under intersection.
* CFL is not closed under Complementation
* Every Regular Language is Context Free

Theorem 1:

CFL is closed Under Union.

Proof:

Given two CFG's G_1 and G_2 that generates the Language L_1 and L_2 respectively, we can Construct the CFG that generates $L_1 \cup L_2$

Let $G_1 = (V_1, \Sigma_1, P_1, S_1)$

$G_2 = (V_2, \Sigma_2, P_2, S_2)$

Where S_1 & S_2 are the Starting Symbols of G_1 & G_2 respectively.

$L_1 \cup L_2$ is given by

$G = (V_1 \cup V_2, \Sigma_1 \cup \Sigma_2, P_3, S)$ where $S \rightarrow S_1$ (or) S_2.

Theorem 2:

CFL is closed under Concatenation.

Proof:

Given two CFGs G_1 and G_2, that generates the languages L_1 and L_2 respectively, we can construct the CFG that generates $L_1 L_2$.

Let $G_1 = (V_1, \Sigma_1, P_1, S_1)$

$G_2 = (V_2, \Sigma_2, P_2, S_2)$

$L_1 L_2$ is given by

$G = (V_1 \cup V_2, \Sigma_1 \Sigma_2, P_3, S)$ where $S \rightarrow S_1 . S_2$.

Theorem 3:

CFL is closed under kleene closure.

Proof:

If G_1 is a CFG generating L_1 then we have to prove L_1 is a CFL.

Let G_1 be a CFG generating the language L_1 with starting symbols S_1. Then we can construct a grammar G Generating L_1 where G Containing all production of G_1 and an additional production $S \rightarrow S_1 S | \epsilon$

Start state

New state.

Theorem 4:

CFL is not closed under intersection.

Proof:

Let L_1 and L_2 be two CFLs

Consider

$$L_1 = \{a^i b^i c^i \mid i \geq 1\}$$

$$L_2 = \{a^i b^i c^i \mid i \geq 1\}$$

$$L = L_1 \cap L_2 = \{a^i b^i c^i \mid i \geq 1\}$$

Ex: $L_1 = abc$ when $i = 1$

$L_2 = aabbcc$ when $i = 2$.

$\therefore$ It is not a ~~Content~~ ~~Free~~ closed under Intersection.

Theorem 5:

CFL is not closed under Complementation.

Proof:

Let L_1 and L_2 be two CFL's.

Assume the CFL is closed under Complementation, then $\overline{L_1}$, $\overline{L_2}$ and $\overline{L_1} \cup \overline{L_2}$ is Content Free.

$L_1 \cap L_2$ Can be Expressed as

$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$

$$L_1 \cap L_2 = L_1 \cup L_2.$$

$L_1 \cap L_2$ is Content Free which is a Contradiction to the theorem. $\therefore$ Our Assumption is false.

$\therefore$ CFL is not Closed under Complementation.

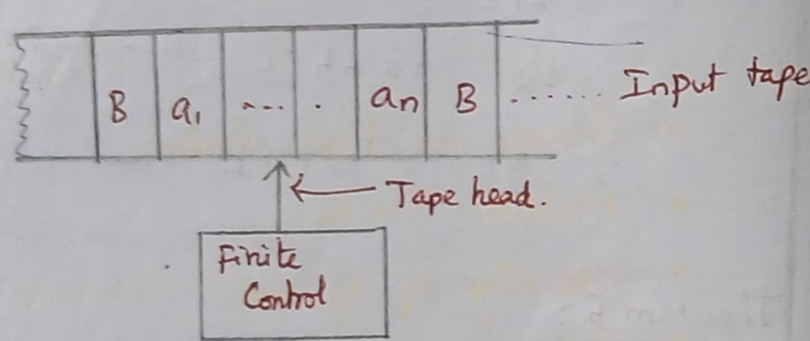
Turing Machine:

Alan Turing introduced a new mathematical model called Turing Machine during the year of 1936.

It is mostly used to define languages and to compute integer functions.

Model of a Turing Machine:

The basic model has a finite control, an i/p tape that is divided into cells, and a tape head that scans one cell of the tape at a time.



Definition:

A turing Machine M is a 7-tuple.

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

where,

- Q - Finite set of states
- Σ - Finite set of i/p symbols.
- Γ - Finite set of tape symbols.
- δ - Mapping (or) transition function.
- q_0 - Initial State
- B - Blank symbol.
- F - Final State.

Instantaneous Description for Turing Machines:

An ID of a Turing Machine is a string $\alpha\beta\gamma$, where β is the present state of M, the entire i/p string is split as $\alpha\gamma$, the first symbol of γ is the current symbol 'a' under R/W head and γ has all the subsequent symbols of the i/p string and the string α is the substring of the i/p string formed by all the symbols to the left of a.

Programming Techniques for Turing Machine: (NOV/DEC 2016) (NOV/DEC 2017)

There are different techniques which is used to construction of turing machine with help of some tools. They are,

* Storage in Finite Control.
* Multiple Tracks.
* Checking of Symbols.
* Subroutines.

Storage in Finite Control:

The finite control can also be used to hold a finite amount of information along with the task of representing a position in the program.

Ex:

Construct a turing Machine M for $\Sigma = \{a, b\}$, $L = \{aba\}$ which will convert lowercase language to upper case language?

Soln:

$L = \{a, b, a, \Delta\}$

$\delta(q_0, a) = (q_1, A, R)$

$\delta(q_1, b) = (q_2, B, R)$

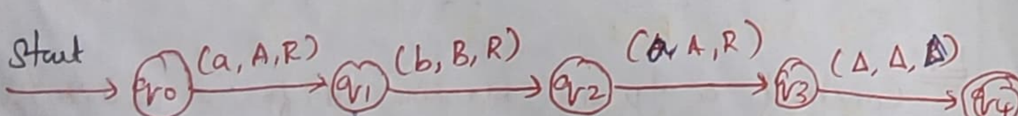
$\delta(q_2, a) = (q_3, A, R)$

$\delta(q_3, \Delta) = (q_4, \Delta, \Delta)$

Transition table:

	a	b	Δ
q_0	(q_1, A, R)		
q_1		(q_2, B, R)	
q_2	(q_3, A, R)		
q_3	-	-	(q_4, Δ, Δ)
q_4			

Transition Diagram:



Multiple Tracks:

The Turing Machine's i/p can be divided into several tracks. Each track can hold one symbol and the tape alphabet of the TM consists of tuples with one component for each track.

#	1	1	1	1	\$	. . . .
B	B	B	B	1	B	. . . .
B	1	1	1	B	B	. . .

↑
Finite Control

Ex: Design a turing m/c to check whether the given i/p is prime or not.

Soln:

(i) 10.

I track: 10, II ": 2, III ": 10 → 10, 2, 8 → 10, 2, 6 → 10, 2, 4 → 10, 2, 2 → 10, 2, 0

$10-2=8$ $8-2=6$ $6-2=4$ $4-2=2$ $2-2=0$

It is not a prime

(ii) 7.

7, 2, 7 → 7, 2, 5 → 7, 2, 3 → 7, 2, 1

$7-2=5$ $5-2=3$ $3-2=1$

$\therefore$ It is a prime number.

5, 4, 5   5, 4, 1   increase by 1

5, 5, 5   II = III

$\therefore$ It is prime.

iii) 5

5, 2, 5   5, 2, 3   5, 2, 1   Increase the 2nd track value

5, 3, 5   5, 3, 2   5, 3, 1   Incre by 1

Checking of Symbols:

Checking of Symbols is an effective way of recognizing the language by TM. The Symbols are to be placed on i/p tape.

The symbol which is read is Marked by any Special Character. The tape

Ex: Construct a turing machine which recognizes the language. $L = \{ wcw \mid w \in (a+b)^* \}$

Soln: $L = \{abcab\}$

q_0 abcabΔ

$\delta(q_0, a) = (q_1, *, R)$	$* b c a b \Delta$
$\delta(q_1, b) = (q_1, b, R)$	$* b c a b \Delta$
$\delta(q_1, c) = (q_1, c, R)$	
$\delta(q_1, a) = (q_2, *, L)$	$* b c * b \Delta$
$\delta(q_2, c) = (q_2, c, L)$	$* b c * b \Delta$
$\delta(q_2, b) = (q_2, b, L)$	$* b c * b \Delta$
$\delta(q_2, *) = (q_2, *, R)$	$* b c * b \Delta$
$\delta(q_2, b) = (q_3, \$, R)$	$* \$ c * b \Delta$
$\delta(q_3, c) = (q_3, c, R)$	$* \$ c * b \Delta$
$\delta(q_3, *) = (q_3, *, R)$	$* \$ c * b \Delta$
$\delta(q_3, b) = (q_4, \$, L)$	$* \$ c * \$ \Delta$
$\delta(q_4, *) = (q_4, *, L)$	$* \$ c * \$ \Delta$

$\delta(q_4, c) = (q_4, c, L)$ * $ c * $ Δ

$\delta(q_4, \$) = (q_4, \$, L)$ * $ c * $ Δ

$\delta(q_4, *) = (q_4, *, R)$ * $ c * $ Δ

$\delta(q_4, \$) = (q_4, \$, R)$ * $ c * $ Δ

$\delta(q_4, c) = (q_4, c, R)$ * $ c * $ Δ

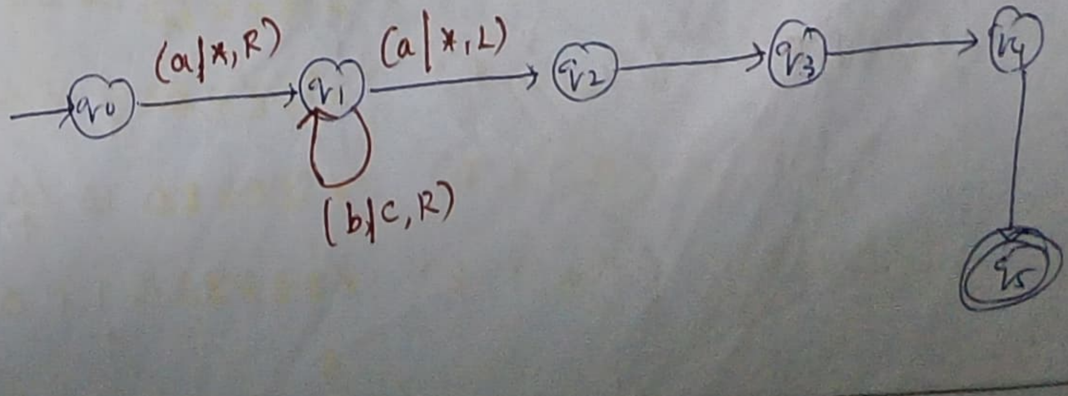
$\delta(q_4, *) = (q_4, *, R)$ * $ c * $ Δ

$\delta(q_4, \$) = (q_4, \$, R)$ * $ c * $ Δ

$\delta(q_4, \Delta) = (q_5, \Delta, \text{Halt})$.

Transition table:

	a	b	c	*	$	Δ
q_0	$(q_1, *, R)$					
q_1	$(q_2, *, L)$	(q_1, c, R)				
q_2		(q_2, b, L) $(q_3, \$, R)$	(q_2, c, L)	$(q_2, *, R)$		
q_3		$(q_4, \$, L)$	(q_3, c, R)	$(q_3, *, R)$		
q_4			(q_4, c, L) (q_4, c, R)	$(q_4, *, L)$ $(q_4, *, R)$	$(q_4, \$, L)$ $(q_4, \$, R)$	q_5
q_5						



2\. Check for the language $L = \{0^n 1^n \mid n \geq 1\}$ [NOV/DEC 2015]

$L = \overset{n=2}{(0011)}$ $\quad$ $a^n b^n \mid n \geq 1$.

$q_0\ 0011B$. $\quad$ $L = \{a^i b^i c^i \mid i \geq 0\}$

$\delta(q_0, 0) = (q_1, X, R)$ $\quad$ $0011 \rightarrow X011B$

$\delta(q_1, 0) = (q_1, 0, R)$ $\quad$ $X011B$

$\delta(q_1, 1) = (q_2, Y, L)$ $\quad$ $X0Y1B$

$\delta(q_2, 0) = (q_2, X, L)$ $\quad$ ~~X0Y1B~~ $\quad$ $XXY1B$

$\delta(q_2, X) = (q_2, X, R)$ $\quad$ $XXY1B$

$\delta(q_2, Y) = (q_2, Y, R)$ $\quad$ $XXY1B$

$\delta(q_2, 1) = (q_3, Y, L)$

$\delta(q_3, Y) = (q_3, Y, L)$

$\delta(q_3, X) = (q_3, X, L)$

$\delta(q_3, X) = (q_3, X, R)$

$\delta(q_3, X) = (q_3, X, R)$

$\delta(q_3, Y, 0) = (q_3, Y, R)$

$\delta(q_3, Y) = (q_3, Y, R)$

$\delta(q_3, B) = (q_4, B, \text{Half})$

Equal no of 0's & 1's.

$L = 1001$

$= 1100$

Sub routine:

A problem with same tasks to be repeated for many number of lines can be programmed using subroutines.

A turing m/c with sub routine is a set of states that perform some careful process.

Ex: Construct a TM for the subroutine $F(a,b) = a*b$ where a and b are many numbers?

$a = 2, \quad b = 3$

$F = a * b$

$= 2 * 3$

$= 6.$

11 \$ 111 ΔΔΔΔΔ

$\delta(q_0, 1) = (q_1, X, R)$ X 1 \$ \$ 1 1 ΔΔΔΔΔΔ

$\delta(q_1, 1) = (q_1, 1, R)$ X 1 \$ 1 1 1 ΔΔΔΔΔΔ

$\delta(q_1, \$) = (q_1, \$, R)$ X 1 \$ 1 1 1 ΔΔΔΔΔΔ

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

$(q_2, 1) = (q_2, 1, R)$ X 1 \$ Y 1 1

$(q_2, 1) = (q_2, 1, R)$ X 1 \$ Y 1 1 ΔΔΔΔΔΔ

$(q_2, \Delta) = (q_2, \Delta, R)$ X 1 \$ Y 1 1 ΔΔΔΔΔΔΔ

$2 * 3 = 6$

11 $ 111 Δ Δ Δ Δ Δ Δ

$\delta(q_0, 1) = (q_1, X, R)$ X 1 $ 111 Δ Δ Δ Δ Δ Δ Δ

$\delta(q_1, 1) = (q_1, 1, R)$ X 1 $ 111 Δ Δ Δ Δ Δ Δ Δ

$\delta(q_1, \$) = (q_1, \$, R)$ X 1 $ 111 Δ Δ Δ Δ Δ Δ Δ

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

$\delta(q_2, 1) = (q_2, 1, R)$ X 1 $ Y 1 1 Δ Δ Δ Δ Δ Δ Δ

$\delta(q_2, 1) = (q_2, 1, R)$ X 1 $ Y 1 1 Δ Δ Δ Δ Δ Δ Δ

$\delta(q_2, \Delta) = (q_2, \Delta, R)$ X 1 $ Y 1 1 Δ Δ Δ Δ Δ Δ

$\delta(q_2, \Delta) = (q_3, 1, L)$ X 1 $ Y 1 1 Δ 1 Δ Δ Δ Δ

$\delta(q_3, \Delta) = (q_3, \Delta, L)$ X 1 $ Y 1 1 Δ 1 Δ Δ Δ Δ

$\delta(q_3, 1) = (q_3, 1, L)$ X 1 $ Y 1 1 Δ 1 Δ Δ Δ Δ Δ

$\delta(q_3, 1) = (q_3, 1, L)$ X 1 $ Y 1 1 Δ 1 Δ Δ Δ Δ Δ

$\delta(q_3, Y) = (q_3, Y, R)$ X 1 $ Y 1 1 Δ 1 Δ Δ Δ Δ Δ

$\delta(q_3, 1) = (q_4, Y, R)$ X 1 $ Y Y 1 Δ 1 Δ Δ Δ Δ Δ

$\delta(q_4, 1) = (q_4, 1, R)$ X 1 $ Y Y 1 Δ 1 Δ Δ Δ Δ

$\delta(q_4, \Delta) = (q_4, \Delta, R)$ X 1 $ Y Y 1 Δ 1 Δ Δ Δ Δ

$\delta(q_4, 1) = (q_4, 1, R)$ X 1 $ Y Y 1 Δ 1 Δ Δ Δ Δ

$\delta(q_4, \Delta) = (q_5, 1, L)$ X 1 $ Y Y 1 Δ 1 1 Δ Δ

$\delta(q_5, 1, 8) = (q_5, 1, L)$ x $ y y 1 Δ 1 1 Δ Δ

$\delta(q_5, \Delta) = (q_5, \Delta, L)$ x $ y y 1 Δ 1 1 Δ Δ

$\delta(q_5, 1) = (q_5, 1, L)$ x $ y y 1 Δ 1 1 Δ Δ

$\delta(q_5, y) = (q_5, y, R)$ x $ y y 1 Δ 1 1 Δ Δ

$\delta(q_5, 1) = (q_6, y, R)$ x $ y y y Δ 1 1 Δ Δ

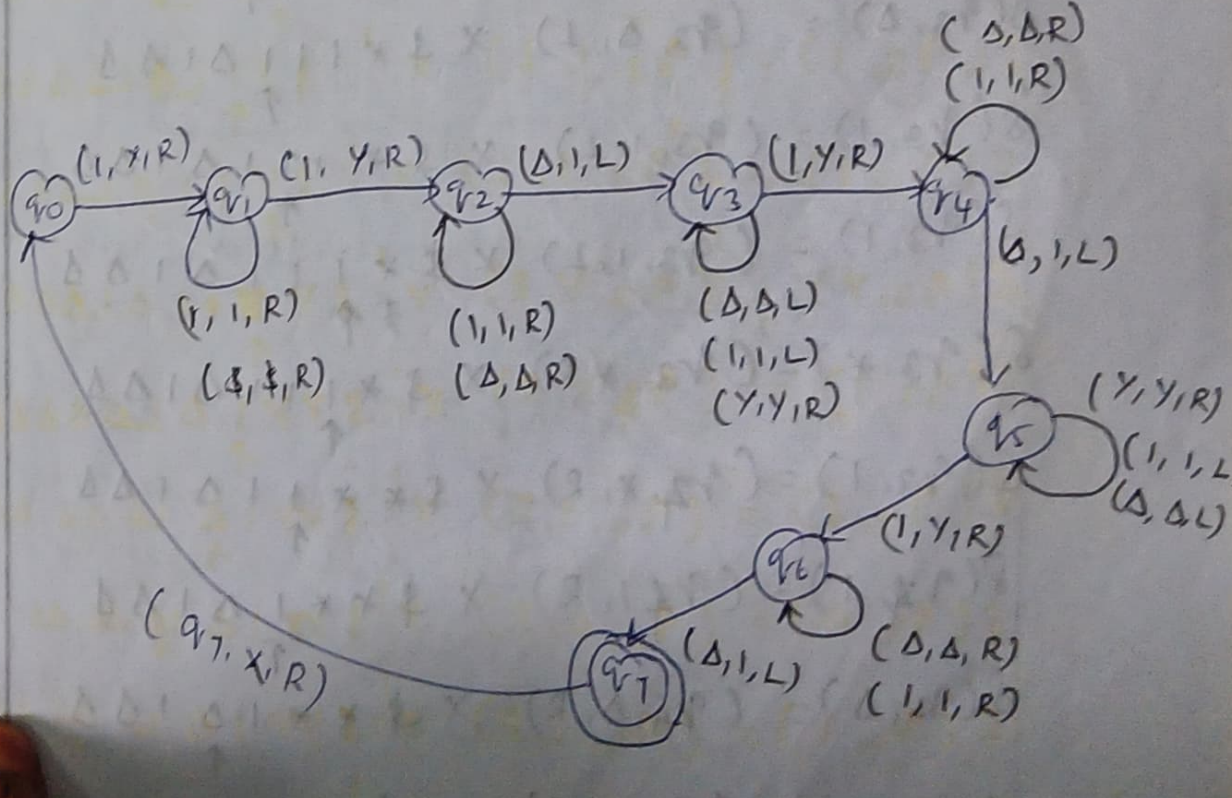
$\delta(q_6, \Delta) = (q_6, \Delta, R)$ x $ y y y Δ 1 1 Δ Δ

$\delta(q_6, 1) = (q_6, 1, R)$ x $ y y y Δ 1 1 Δ Δ

$\delta(q_6, 1) = \delta(q_6, 1, R)$ x $ y y y Δ 1 1 Δ Δ

$\delta(q_6, \Delta) = \delta(q_7, 1, L)$ x $ y y y Δ 1 1 1

call subroutine for other x and y operations.



Solve, $F(a,b) = a * b$, where $a = 1$, $b = 3$.

$\therefore \quad F = a * b$

$= 1 * 3 = 3$

$1 \, \$ \, 1 1 1 \, \Delta \, \Delta \Delta \Delta$

$\delta(q_0, 1) = (q_1, x, R)$ $\quad 1 \, \$ \, 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_1, \$) = (q_1, \$, R)$ $\quad x \, \$ \, 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_1, 1) = (q_2, *, R)$ $\quad x \, \$ * 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_2, 1) = (q_2, 1, R)$ $\quad x \, \$ * 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_2, 1) = (q_2, 1, R)$ $\quad x \, \$ * 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_2, \Delta) = (q_2, \Delta, R)$ $\quad x \, \$ * 1 1 1 \, \Delta \Delta \Delta \Delta$

$\delta(q_2, \Delta) = (q_3, 1, L)$ $\quad x \, \$ * 1 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_3, \Delta) = (q_3, \Delta, L)$ $\quad x \, \$ * 1 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_3, 1) = (q_3, 1, L)$ $\quad x \, \$ * 1 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_3, 1) = (q_3, 1, L)$ $\quad x \, \$ * 1 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_3, *) = (q_3, *, R)$ $\quad x \, \$ * 1 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_3, 1) = (q_4, *, R)$ $\quad x \, \$ * * 1 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_4, 1) = (q_4, 1, R)$ $\quad x \, \$ * * 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_4, \Delta) = (q_4, \Delta, R)$ $\quad x \, \$ * * 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_4, 1) = (q_4, 1, R)$ $\quad x \, \$ * * 1 \, \Delta \, 1 \, \Delta \Delta$

$\delta(q_4, \Delta) = (q_5, 1, L)$ x $ x x 1 Δ 1 1 Δ

$\delta(q_5, 1) = (q_5, 1, L)$ x $ x x 1 Δ 1 1 Δ

$\delta(q_5, \Delta) = (q_5, \Delta, L)$ x $ x x 1 Δ 1 1 Δ

$\delta(q_5, 1) = (q_5, 1, L)$ x $ x x 1 Δ 1 1 Δ

$\delta(q_5, x) = (q_5, x, R)$ x $ x x 1 Δ 1 1 Δ

$\delta(q_5, 1) = (q_6, x, R)$ x $ x x x Δ 1 1 Δ

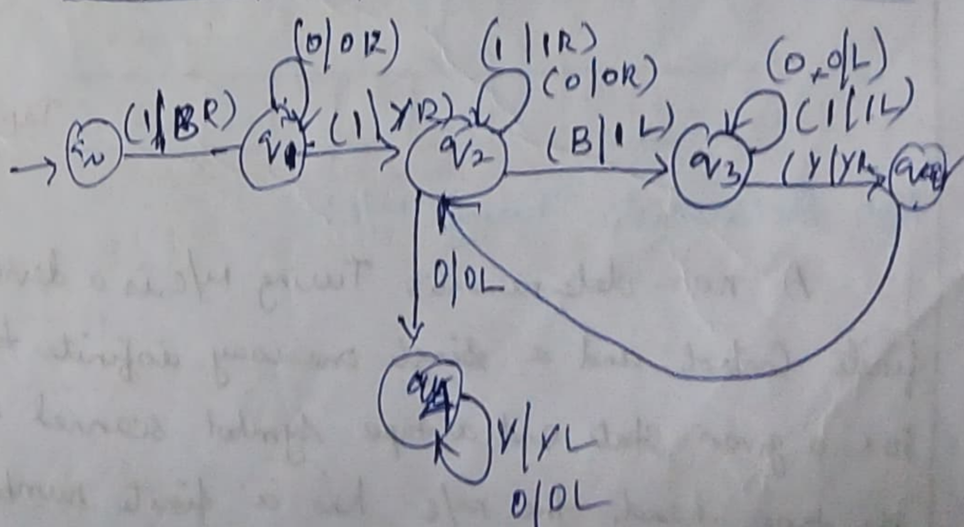
$\delta(q_6, \Delta) = (q_6, \Delta, R)$ x $ x x x Δ 1 1 Δ

$\delta(q_6, 1) = (q_6, 1, R)$ x $ x x x Δ 1 1 Δ

$\delta(q_6, 1) = (q_6, 1, R)$ x $ x x x Δ 1 1 Δ

$\delta(q_6, \Delta) = (q_7, 1, L)$ x $ x x x Δ 1 1 1 Add Δ at least.

1	0	1	1	1	0	B	B	B	. . .



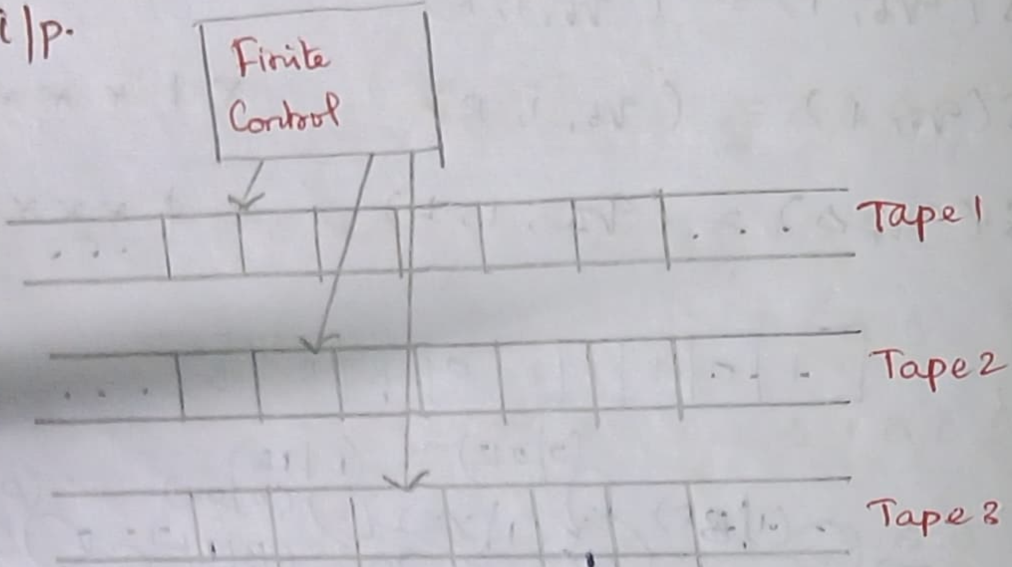
Multitape Turing Machine (or) Two way infinite Tape:

(APR/MAY)

A multitape turing machine has a finite control with some finite number of tapes. Each tape is infinite in both directions.

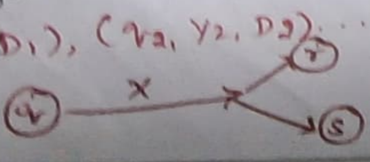
It has its own initial state and accepting state.

* The Finite set of i/p symbols is placed on the first tape.
* All the other cells of all the tapes hold the blank
* The control head of the first tape is at the left end of the i/p.



Non-Deterministic Turing M/c:

A non-deterministic Turing M/c is a device with a finite control and a single one-way infinite tape. For a given state and a tape symbol scanned by the tape head, the m/c has a finite number of choices for the next move. For $\delta(q, x)$, there is a set of tr {$(q_1, Y_1, D_1), (q_2, Y_2, D_2) \ldots (q_K, Y_K, D_K)$} where K is any integer



1. Construct a TM that performs addition Operation.

Soln:

$F(x+y) = x+y$. $x=2, y=3$. $x+y=5$.

$1 1 \$ 1 1 1 \Delta \Delta \Delta \Delta \Delta \Delta$

2. Construct a TM that performs subtraction operation of ~~Soln:~~ two nos. $f(a-b)=c$ where a is always greater than b

$F(a-b) = c$

$a=3, b=2$ $F(3-2)=1$

$111 \$ 11 \Delta \Delta$.

3. Construct a TM for a Successor function for a given many number $f(n)=n+1$?

Soln: If $n=4$ $f(n) = 4+1 = 5$.

$(q_0, 1 1 1 1)$.

$(q_0, 1) = (q_1, X, R)$

⋮

4. Convert the given CFG to GNF.

$S \to CA$

$A \to a$

$C \to aB \mid b$.

~~Ans: $S \to aBA \mid bA$~~

5. Convert the CFG to CNF

$S \to ABA$

$A \to aA \mid \epsilon$

$B \to bB \mid \epsilon$

6. ~~S→~~10. Convert the grammar into CNF

$S \rightarrow aSaA \mid A$

$A \rightarrow abA \mid b.$

7. Find a CNF with no useless symbols equivalent to

$S \rightarrow AB \mid CA$

$B \rightarrow BC \mid AB$

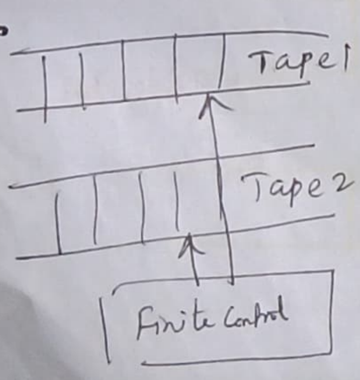
$A \rightarrow a$

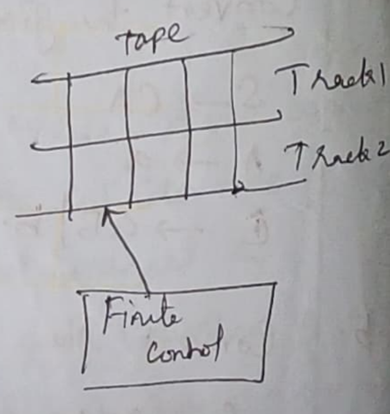
$C \rightarrow aB \mid b.$

8. Construct a TM for the Language $L = \{wcw^R \mid w \text{ is } (0+1)^*\}$

$L = 01C10$

Multitape Turing M/c	Multitrack TM.
i) Many tapes are there with its own initial state & accepting states.	One tape is divided into multiple tracks.
ii) Many finite control heads are there.	Only one finite control is there.





9. Convert the grammar with productions into CNF

$A \to bAB \mid \lambda, \quad B \to BAa \mid \lambda \quad [\because \lambda \equiv \epsilon]$.

10. Find a CNF for	GNF
$S \to aAbB$	$S \to AB$
$A \to aA \mid a$	$A \to BS \mid b$
$B \to bB \mid b$	$B \to SA \mid a$.

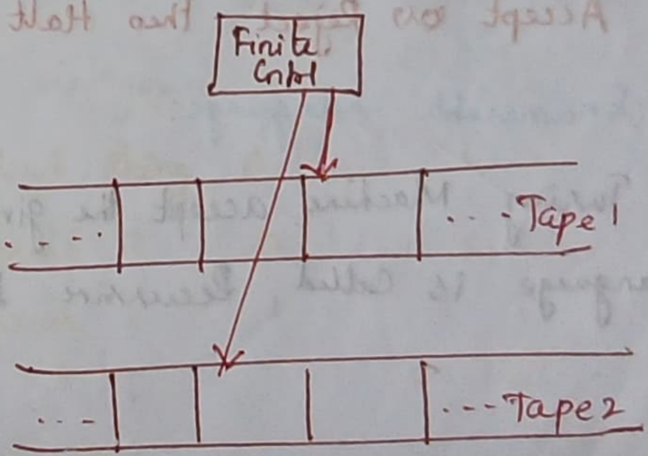
Show that if L is accepted by a Multitape TM, it is accepted by single tape TM also.

Soln:

Let M_1 be a Multitape TM with k tapes. and M_2 be the one-tape (or) single tape TM with $2k$ tracks, where,

(i) One track records the Contents of the tape M_1

(ii) other track holds the symbol Scanned by the Corresponding head of M_1 acts as a marker.



The Finite Control of M_2 Stores

i) The State of M_1

ii) The Count Specifying the number of head markers to the right of M_2's tape head.

6. ~~8→10~~. Convert the grammar into CNF

$S \to aSaA \mid A$

$A \to abA \mid b.$

7. Find a CNF with no useless symbols equivalent to

$S \to AB \mid CA$

$B \to BC \mid AB$

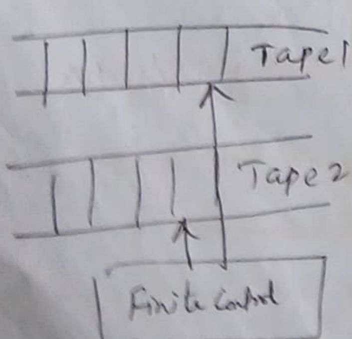
$A \to a$

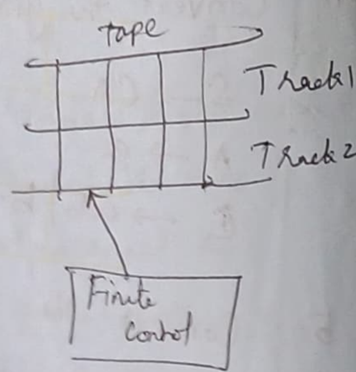
$C \to aB \mid b.$

8. Construct a TM for the Language $L = \{wcw^R \mid w \text{ is } (0+1)^*$

$L = 01c10$

Multitape Turing M/c	Multitrack TM.
1) Many tapes are there with its own initial state & accepting states.	One tape is divided into multiple tracks.
ii) Many finite control heads are there.	Only one finite control is there.





9. Convert the grammar with productions into CNF

$A \rightarrow bAB \mid \lambda, \quad B \rightarrow BAa \mid \lambda \quad [\because \lambda \equiv \epsilon]$.

10. Find a CNF for

$S \rightarrow aAbB$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b$

GNF

$S \rightarrow AB$

$A \rightarrow BS \mid b$

$B \rightarrow SA \mid a$.

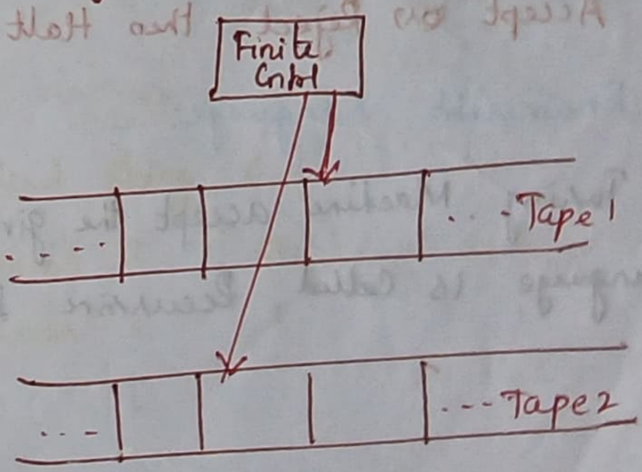
Show that if L is accepted by a Multitape TM, it is accepted by single tape TM also.

Soln:

Let M_1 be a Multitape TM with k tapes. and M_2 be the one-tape (or) single tape TM with $2k$ tracks, where,

(i) One track records the Contents of the tape M_1

(ii) other track holds the symbol Scanned by the Corresponding head of M_1 acts as a marker.



The Finite Control of M_2 Stores

i) The State of M_1

ii) The Count Specifying the number of head markers to the right of M_2's tape head.