Ambuiguity in CFG :-

A terminal string $w \in L(G)$ is ambiguous if there exists two (or) more left most or right most derivations for w.

Ex:- $E \rightarrow E+E$
$/E*E/(E)/id$

$w = id*id+id$

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

$\therefore$ It is ambiguos.

7/3/13

UNIT-4

Minimization of Context free Grammer :-

If 'L' is a non-empty CFL, then it can be generated by CFG 'G' with the following properties:

1. Each variable and each terminal of 'G' appears in the derivation of some word in 'L'.

2. There are no productions of the form $A \rightarrow B$ where A & B are variables.

→ A CFG can be minimized by eliminating

a. null productions

b. unit productions.

c. Non-terminals which cannot derive terminal strings.

d. Non-terminals which cannot appear in sentinential form.

→ **Useless symbols:-**

A symbol 'X' is useful if there is a derivation $S \overset{*}{\Rightarrow} \alpha X \beta \overset{*}{\Rightarrow} w$ for some α, β and w where $w \in T^*$, otherwise X is useless.

→ **Null productions (or) ε-productions:-**

A production of the form $A \rightarrow \epsilon$ is said to be ε-production if ε is in L(G)

then all ϵ-productions from G cannot be eliminated
Otherwise ϵ-productions can be eliminated.

Q: Minimize the following given grammer.

$$S \rightarrow aA|a|B|C$$
$$A \rightarrow aB|\epsilon$$
$$B \rightarrow aA$$
$$C \rightarrow cCD$$
$$D \rightarrow abd$$

Sol: First eliminate the null productions, unit productions and useless symbols from given grammer, then we get reduced grammer.

1. Elimination of ϵ-productions (null) :-

(i) Find the nullable variable set V_n. This set includes all variables which are deriving ϵ symbols.

$$\therefore V_n = \{A\}$$

(ii) Find the productions whose R.H.S. doesn't contain any nullable variable A.

$$\Rightarrow S \rightarrow a|B|C, \; A \rightarrow aB|\epsilon$$
$$C \rightarrow cCD$$
$$D \rightarrow abd$$

(iii) In the remaining productions, replace nullable variable A with ϵ in all possible combinations

$$\Rightarrow S \rightarrow aA$$
$$B \rightarrow aA$$

replace A with ϵ $\Rightarrow$ $S \rightarrow a$
$B \rightarrow a$

~~Put~~

$S \to a \Rightarrow S \to aA|a$

$B \to a \Rightarrow B \to aA|a$

$\therefore$ The resultant productions are

$S \to aA|a|B|C$

$A \to aB$

$B \to aA|a$

$C \to cCD$

$D \to abd.$

2. <u>Elimination of unit productions</u>:-

$$\left.\begin{array}{l} S \to B \\ S \to C \end{array}\right\} \text{unit productions.}$$

So substitute $B \to aA|a$ & $C \to cCD$ in $S \to B$ & $S \to C$ respectively.

$\Rightarrow S \to aA|a|cCD$

$A \to aB$

$B \to aA|a$

$C \to cCD$

$D \to abd$

3. <u>Elimination of useless symbols</u>:-

(i) Find the variables which are not deriving terminal strings.

$S \to aA$	$S \to cCD$
$\to aaB$	$\to ccCDD$
$\to aaa$ (✓)	$\to ccCabdD$
	$\to ccCabdabd$ (X)

$\therefore$ ~~it is eliminated~~.

$\therefore$ eliminate C & its productions

$B \rightarrow aA$

$\rightarrow aaB$

$\rightarrow aaa (v)$

$\therefore$ The new grammer is

$S \rightarrow aA \mid a$

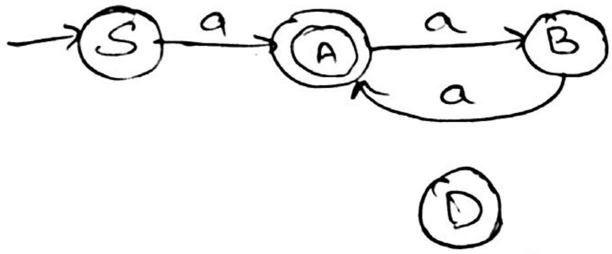
$A \rightarrow aB$

$B \rightarrow aA \mid a$

$D \rightarrow abd.$

(ii) Find the variables which cannot be reached from start symbols and delete those variables & its productions.

The dependency graph for above grammer is



$\therefore$ There is no path to 'D' from start symbol 'S'

$\therefore$ The variable 'D' is useless. Hence eliminate 'D' and its productions.

$\therefore$ The final reduced grammer is

$S \rightarrow aA \mid a$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

Q: Reduce the following grammer.

$S \rightarrow AB \mid CA$

$B \rightarrow BC \mid AB$

$A \rightarrow a$

$C \rightarrow aB \mid b$

Sol^n: 3. (i) $S \rightarrow AB$

$\rightarrow aBC$

$\rightarrow aBb$ (x)

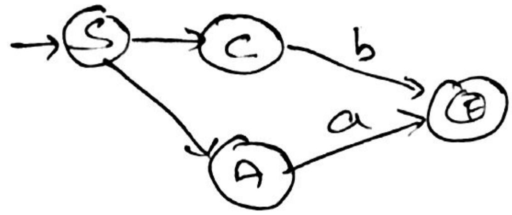
$\therefore$ B is not deriving any terminal string.

$\Rightarrow$ $S \rightarrow CA$ $\quad$ $\therefore$ delete B & its production

$A \rightarrow a$

$C \rightarrow b$

(ii)



From the above dependency graph, we can say there is a path to A & C from S.

$\therefore$ S, A, C are useful variables.

$\therefore$ The final reduced grammer is

$S \rightarrow CA$

$A \rightarrow a$

$C \rightarrow b$

Q: Eliminate all unit & ϵ-productions.

$S \rightarrow AaB \mid aaB$

$A \rightarrow D$

$B \rightarrow bbA \mid \epsilon$

$D \rightarrow E$

$E \rightarrow F$

$F \rightarrow aS$

Sol: 1. (i) $V_n = \{B\}$

(ii) $A \rightarrow D$

$B \rightarrow bbA \mid \epsilon$

$D \rightarrow E$

$E \rightarrow F$

$F \rightarrow aS$

(iii) $S \rightarrow Aa \mid aa$.

$\Rightarrow S \rightarrow AaB \mid Aa \mid aa \mid aaB$.

final $\Rightarrow$ $S \rightarrow AaB \mid Aa \mid aa \mid aaB$

$A \rightarrow D$

$B \rightarrow bbA \mid \epsilon$

$D \rightarrow E$

$E \rightarrow F$

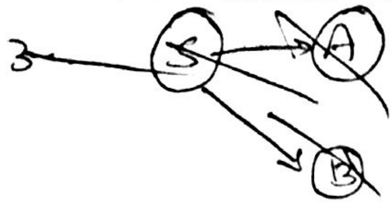
$F \rightarrow aS$

2.

$S \rightarrow AaB \mid Aa \mid aa \mid aaB$, $A \rightarrow aS$

$B \rightarrow bbA$, $D \rightarrow aS$, $E \rightarrow aS$

$F \rightarrow aS$



3. $S \rightarrow AaB \mid aaB \mid Aa \mid aa$

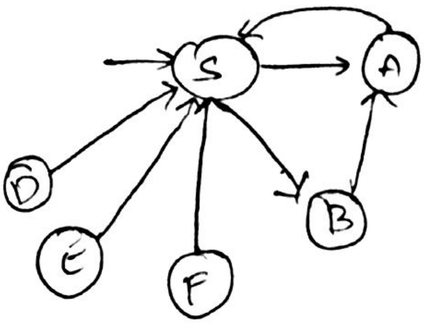
$A \rightarrow aS$

$B \rightarrow bbA$

$D \rightarrow aS$

$E \rightarrow aS$

$F \rightarrow aS$



∴ delete D, E, F

∴ $S \rightarrow AaB \mid aaB \mid Aa \mid aa$

$A \rightarrow aS$

$B \rightarrow bbA$.

11/3/13

Normal forms of context free Grammer:-

→ By placing restrictions on right hand side of CFG, we have 2 normal forms.

1. Chomsky normal form. (CNF)
2. Greibach normal form. (GNF)

1. Chomsky normal form:-

Any CFL without ϵ is generated by a grammer in which all productions are of the form $A \to BC$ (or) $A \to a$.

where A, B, C are variables (or) non-terminals and a is a terminal.

Q: Reduce the following grammer to CNF.

$S \to aAD$

$A \to aB \mid bAB$

$B \to b$

$D \to d$.

first minimize the grammer by eliminating null, unit productions and useless symbols. (it is already reduced).

Sol: The productions should be in the form

$A \to BC$ (or) $A \to a$.

$B \to b$ and $D \to d$ are already in the form of CNF.

The remaining productions are

$S \to aAD$

$A \to aB \mid bAB$.

To convert the above productions into CNF, introduce two new non-terminals symbols C_a & C_b

i.e., $C_a \to a$, $C_b \to b$

replace these productions in the above prodn

$\Rightarrow$ $S \to C_aAD$

$A \to C_aB \mid C_bAB$.

Now introduce two new non-terminal symbols.

$E \to AD$, $F \to AB$

$\Rightarrow$ $S \to C_aE$

$A \to C_aB \mid C_bF$.

$\therefore$ The final productions are

$S \to C_aE$

$A \to C_aB \mid C_bF$

$E \to AD$

$F \to AB$

$C_a \to a$

$C_b \to b$

$B \to b$

$D \to d$

Q: Reduce the following grammer to CNF.

$S \to bA \mid aB$

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

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

Sol: This is already in minimized form.

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

remaining are

$S \to bA \mid aB$

$A \to bAA \mid aS$

$B \to aBB \mid bS$

introduce $C_a \rightarrow a$, $C_b \rightarrow b$

$$\Rightarrow \quad S \rightarrow C_bA \mid C_aB$$
$$A \rightarrow C_bAA \mid C_aS$$
$$B \rightarrow C_aBB \mid C_bS$$

introduce $E \rightarrow AA$, $F \rightarrow BB$

$$\Rightarrow \quad S \rightarrow C_bA \mid C_aB$$
$$A \rightarrow C_bE \mid C_aS$$
$$B \rightarrow C_aF \mid C_bS$$

$\therefore$ final productions are

$$S \rightarrow C_bA \mid C_aB$$
$$A \rightarrow C_bE \mid C_aS \mid a$$
$$B \rightarrow C_aF \mid C_bS \mid b.$$
$$C_a \rightarrow a$$
$$C_b \rightarrow b$$
$$E \rightarrow AA$$
$$F \rightarrow BB$$

Q: Reduce the following to CNF.

$$S \rightarrow aAbB$$
$$A \rightarrow aA \mid a$$
$$B \rightarrow bB \mid b$$

Sol:

$$C_a \rightarrow a, \quad C_b \rightarrow b$$

$$\Rightarrow \quad S \rightarrow C_aAC_bB$$
$$A \rightarrow C_aA \mid a$$
$$B \rightarrow C_bB \mid b.$$

$E \to C_a A$, $F \to C_b B$

$\Rightarrow S \to EF$

$A \to C_a A \mid a$

$B \to C_b B \mid b$

$C_a \to a$

$C_b \to b$

$E \to C_a A$

$F \to C_b B$

Q: Given the CNF for

$S \to AB \mid CA$

$B \to BC \mid AB$

$A \to a$

$C \to aB \mid b$

Sol: final reduced grammer is

$S \to CA$

$A \to a$ (from back).

$C \to b$

This is final CNF.

Q: Convert the following to CNF

$S \to aA \mid a \mid B \mid C$

$A \to aB \mid \epsilon$

$B \to aA$

$C \to cCD$

Sol: (i) reduced form is $S \to aA \mid a$

$A \to aB$ (from back)

$B \to aA \mid a$

(ii) converting to CNF.

$C_a \to a.$

$\Rightarrow S \to C_a A | a$

$A \to C_a B$

$B \to C_a A | a.$

∴ final CNF productions are

$S \to C_a A | a$

$A \to C_a B$

$B \to C_a A | a$

$C_a \to a.$

2. Greibach Normal form:-

Every CFL 'L' without 'ε' canbe generated by a grammer for which every production is of the form $A \to a\alpha$. where $A \to$ variable, a is a terminal and 'α' is a string of variables (empty).

→ To convert a given grammer into GNF, the following two lemmas are considered.

lemma-1: The production $A_i \to A_j \gamma$. if $j \geq i$ then lemma-1 is satisfied.

lemma-2: let $A \to A\alpha_1 | A\alpha_2 | .. | A\alpha_r$ be the set of A productions for which A is the left most symbol of the right hand side.

let $A \to \beta_1 | \beta_2 | ... | \beta_s$ be the remaining A productions.

Let $G^1 = (V \cup \{B\}, T, P^1, S)$ be the CFG formed by adding the variable 'B' to 'V' and replacing all the 'A' productions by the following productions:

1) $\left.\begin{array}{l} A \rightarrow \beta_i \\ A \rightarrow \beta_i B \end{array}\right\} 1 \leq i \leq \beta$

2) $\left.\begin{array}{l} B \rightarrow \alpha_i \\ B \rightarrow \alpha_i B \end{array}\right\} 1 \leq i \leq r$

Then $L(G^1) = L(G)$.

Q: Reduce the following into GNF.

Sol?:

$A_1 \rightarrow A_2 A_3$

$A_2 \rightarrow A_3 A_1 | b$ (already minimized)

$A_3 \rightarrow A_1 A_2 | a$.

∴ The right hand side of the productions for A_1 and A_2 start with terminals (or) higher number variables.

we Begin with the production

$A_3 \rightarrow A_1 A_2$

and substitute the string $A_2 A_3$ for A_1.

∴ $A_3 \rightarrow A_2 A_3 A_2$ (Not satisfying $j > i$)

again substitute $A_2 \rightarrow A_3 A_1 | b$ for first occurrence of A_2

∴ $A_3 \rightarrow A_3 A_1 A_3 A_2 \mid b A_3 A_2$

∗ ~~Right~~

Now new set is

$A_1 \rightarrow A_2 A_3$

$A_2 \rightarrow A_3 A_1 | b$

$A_3 \rightarrow A_3 A_1 A_3 A_2 | b A_3 A_2 | a$

Now we apply lemma-2 to the productions.

Consider $A_3 \rightarrow A_3 \overbrace{A_1 A_3 A_2}^{\alpha_1} \mid \overbrace{b A_3 A_2}^{\beta_1} \mid \overbrace{a}^{\beta_2}$

Here $r = 1$ & $s = 2$

Now replace all the A_3 productions by

1. $\left.\begin{array}{l} A \rightarrow \beta_i \\ A \rightarrow \beta_i B_3 \end{array}\right\} 1 \le i \le r$

2. $\left.\begin{array}{l} B_3 \rightarrow \alpha_i \\ B_3 \rightarrow \alpha_i B_3 \end{array}\right\} 1 \le i \le s.$

and symbol B_3 is introduced.

~~consider A_3~~

By substituting β_1 & β_2, we get

$$A_3 \rightarrow bA_3A_2 \mid a \mid bA_3A_2B_3 \mid aB_3$$

By substituting α_1, we get

$$B_3 \rightarrow A_1A_3A_2 \mid A_1A_3A_2B_3$$

$\therefore$ The resulting set is

$A_1 \rightarrow A_2A_3$

$A_2 \rightarrow A_3A_1 \mid b$

$A_3 \rightarrow bA_3A_2 \mid a \mid bA_3A_2B_3 \mid aB_3$

$B_3 \rightarrow A_1A_3A_2 \mid A_1A_3A_2B_3.$

Substitute A_3 in A_2

$\Rightarrow A_2 \rightarrow bA_3A_2A_1 \mid aA_1 \mid bA_3A_2B_3A_1 \mid aB_3A_1 \mid b$

Substitute A_2 in A_1

$\Rightarrow A_1 \rightarrow bA_3A_2A_1A_3 \mid aA_1A_3 \mid bA_3A_2B_3A_1A_3 \mid$

$aB_3A_1A_3 \mid bA_3.$

in B_3, substitute A_1.

$\therefore$ (A) $B_3 \rightarrow bA_3A_2A_1A_3A_3A_2 \mid aA_1A_3A_3A_2 \mid$
$bA_3A_2B_3A_1A_3A_3A_2 \mid aB_3A_1A_3A_3A_2 \mid$
$bA_3A_3A_2 \mid bA_3A_2A_1A_3A_3A_2B_3 \mid$
$aA_1A_3A_3A_2B_3 \mid bA_3A_2B_3A_1A_3A_3A_2B_3 \mid$
$aB_3A_1A_3A_3A_2B_3 \mid bA_3A_3A_2B_3$.

$\therefore$ final GNF productions are

(A) $A_1 \rightarrow bA_3A_2A_1A_3 \mid aA_1A_3 \mid bA_3A_2B_3A_1A_3 \mid$
$aB_3A_1A_3 \mid bA_3$.

(A) $A_2 \rightarrow bA_3A_2A_1 \mid aA_1 \mid bA_3A_2B_3A_1 \mid aB_3A_1 \mid b$.

(A) $A_3 \rightarrow bA_3A_2 \mid a \mid bA_3A_2B_3 \mid aB_3$.

$B_3 \rightarrow$ (above)

11/3/13

Q: Convert into GNF.

$E \rightarrow E+T \mid T$

$T \rightarrow T*F \mid F$

$F \rightarrow (E) \mid a.$

Minimization:-

$E \rightarrow E+T \mid T*F \mid (E) \mid a$

$T \rightarrow T*F \mid (E) \mid a$

$F \rightarrow (E) \mid a$

let $\left.\begin{array}{l} X \rightarrow + \\ Y \rightarrow * \\ Z \rightarrow) \end{array}\right\}$ New 3 variables.

$\Rightarrow E \rightarrow EXT \mid TYF \mid (EZ \mid a$

$T \rightarrow TYF \mid (EZ \mid a$

$F \rightarrow (EZ \mid a$

$X \rightarrow +$

$Y \rightarrow *$

$Z \rightarrow)$

E, T, F, X, Y, Z are renamed with

$A_1, A_2, A_3, A_4, A_5, A_6.$

$\Rightarrow A_1 \rightarrow A_1A_4A_2 \mid A_2A_5A_3 \mid (A_1A_6 \mid a$

$A_2 \rightarrow A_2A_5A_3 \mid (A_1A_6 \mid a$

$A_3 \rightarrow (A_1A_6 \mid a$

$A_4 \rightarrow +$

$A_5 \rightarrow *$

$A_6 \rightarrow)$

lemma-1:-

$A_i \rightarrow A_j \;;\; j \geq i$

satisfied by all the productions.

lemma-2:-

$A \rightarrow A\alpha_1 \mid A\alpha_2 \mid \dots \mid A\alpha_r \mid \beta_1 \mid \beta_2 \dots \mid \beta_s$

$\Rightarrow A \rightarrow \beta_i \mid \beta_i B$

$B \rightarrow \alpha_i \mid \alpha_i B.$

$A_2 \rightarrow A_2 \underbrace{A_5 A_3}_{\alpha_1} \mid \underbrace{(A_1 A_6}_{\beta_1} \mid \underbrace{a}_{\beta_2}$

$\Rightarrow A_2 \rightarrow (A_1 A_6 \mid (A_1 A_6 B_2 \mid a \mid a B_2$

$B_2 \rightarrow A_5 A_3 \mid A_5 A_3 B_2$

Now substitute A_2 in A_1

$\therefore A_1 \rightarrow A_1 \underbrace{A_4 A_2}_{\alpha_1} \mid \underbrace{(A_1 A_6 A_5 A_3}_{\beta_1} \mid \underbrace{(A_1 A_6 B_2 A_5 A_3}_{\beta_2} \mid$

$\underbrace{a A_5 A_3}_{\beta_3} \mid \underbrace{a B_2 A_5 A_3}_{\beta_4} \mid \underbrace{(A_1 A_6}_{\beta_5} \mid \underbrace{a}_{\beta_6}$

$\Rightarrow A_1 \rightarrow (A_1 A_6 A_5 A_3 \mid (A_1 A_6 A_5 A_3 B_1 \mid (A_1 A_6 B_2 A_5 A_3 \mid$

$(A_1 A_6 B_2 A_5 A_3 B_1 \mid a A_5 A_3 \mid a A_5 A_3 B_1 \mid$

$a B_2 A_5 A_3 \mid a B_2 A_5 A_3 B_1 \mid (A_1 A_6 \mid (A_1 A_6 B_1 \mid$

$a \mid a B_1.$

$B_1 \rightarrow A_4 A_2 \mid A_4 A_2 B_1$

$\downarrow$

$\Rightarrow B_1 \rightarrow + A_2 \mid + A_2 B_1$

$B_2 \rightarrow * A_3 \mid * A_3 B_2$

25/3/13 M

PUSH DOWN AUTOMATA (PDA)

→ A PDA M is a system $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$
where Q - finite set of states.

Σ - alphabet

Γ - alphabet called stack alphabet

$q_0 - \in Q$ is an initial state

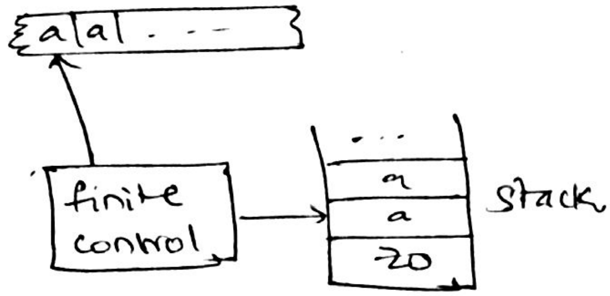
$Z_0 - \in \Gamma$ is a particular stack symbol called initial stack symbol.

F - is a subset of Q, set of final states.

δ - Transition function.

$\delta : (Q \times (\Sigma \cup \epsilon) \times \Gamma) \rightarrow (Q \times \Gamma^*)$

→ The modal of PDA is as follows:



↳ The PDA on some state and on reading an i/p symbol from tape and top most symbol from push down stack (PDS) moves to a new state and writes a string of symbols in PDS.

↳ The move of PDA (interpretation) is

$\delta(q, a, z) = ((P_1, \gamma_1), (P_2, \gamma_2) \ldots (P_n, \gamma_n))$

where $q, P_i, 1 \le i \le n$ are states,

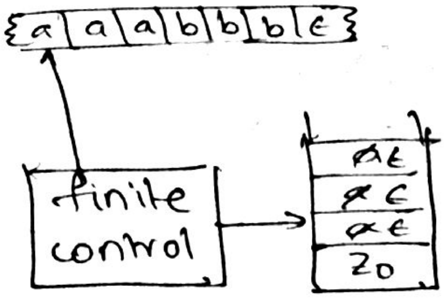
a is in Σ.

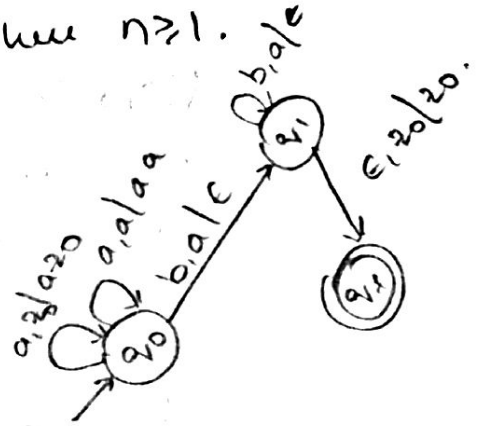
↳ Σ is a stack symbol and γ_i is in Γ^*, $1 \le i \le n$, n is the stack size.

→ The interpretation of $\delta(q, \epsilon, Z) = ((P_1, r_1), (P_2, r_2) \ldots (P_m, r_m))$ is that the PDA in state 'q' independent of the input symbol being scanned and with 'Z', the top most symbol on the stack, can enter a state P_i and replace it by r_i for any $i, 1 \leq i \leq m$.

Q: Design a PDA for $a^n b^n$ where $n \geq 1$.

Sol:

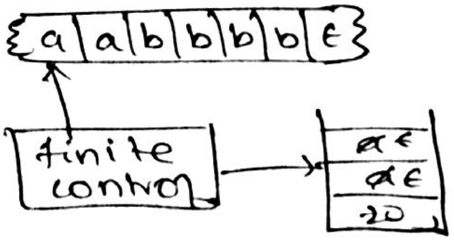


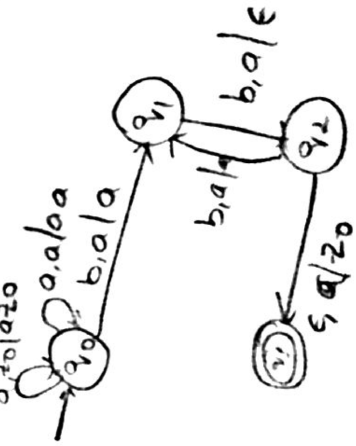


$$\delta(q_0, a, z_0) \rightarrow (q_0, az_0)$$
$$\delta(q_0, a, a) \rightarrow (q_0, aa)$$
$$\delta(q_0, b, a) \rightarrow (q_1, \epsilon)$$
$$\delta(q_1, b, a) \rightarrow (q_1, \epsilon)$$
$$\delta(q_1, \epsilon, z_0) \rightarrow (q_f, z_0).$$

Q: Construct a PDA for $a^n b^{2n} \mid n \geq 1$.

Sol:





$$\delta(q_0, a, z_0) \rightarrow (q_0, az_0)$$
$$\delta(q_0, a, a) \rightarrow (q_0, aa)$$
$\delta(q_0, b, a) \rightarrow (q_1, a)$ as it is

$\delta(q_1, b, a) \rightarrow (q_2, \epsilon)$ pop

$\delta(q_2, b, a) \rightarrow (q_1, a)$ as it is

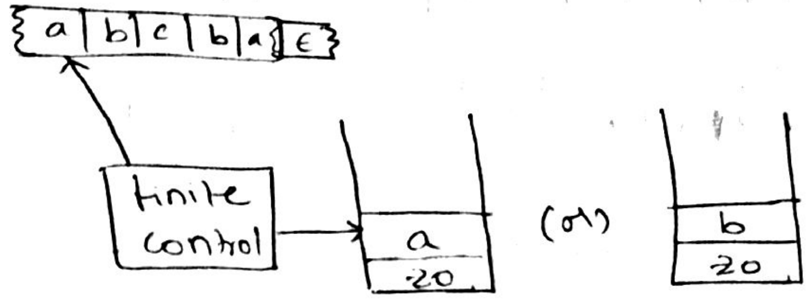
$\delta(q_2, \epsilon, a) \rightarrow (q_f, z_0)$ pop.

Generated by CamScanner from intsig.com

1/4/13

Q: Design a PDA for $L = \{wcw^R \mid w \text{ is } (a+b)^*\}$

Sol: $L = \{wcw^R \mid w \text{ is } (a+b)^*\}$



$\delta(q_0, a, z_0) \rightarrow (q_0, az_0)$ (aca)

$\delta(q_0, b, z_0) \rightarrow (q_0, bz_0)$ (bcb)

$\delta(q_0, a, a) \rightarrow (q_0, aa)$ (aacaa)

$\delta(q_0, b, a) \rightarrow (q_0, ba)$ (abcab)

$\delta(q_0, b, b) \rightarrow (q_0, bb)$ (bbcbb)

$\delta(q_0, a, b) \rightarrow (q_0, ab)$ (bacab)

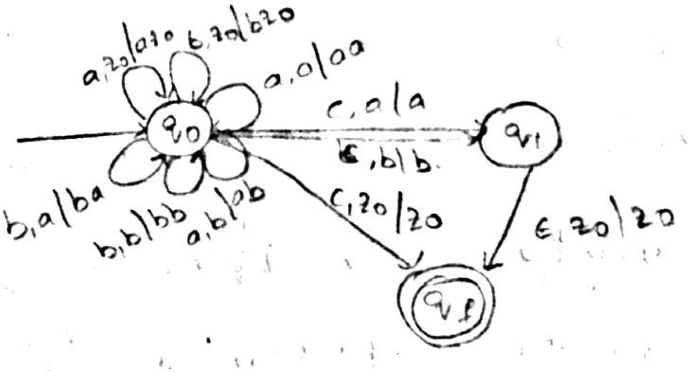
$\delta(q_0, c, z_0) \rightarrow (q_f, z_0)$ (c)

$\left.\begin{array}{l} \delta(q_0, c, a) \rightarrow (q_1, a) \\ \delta(q_0, c, b) \rightarrow (q_1, b) \end{array}\right\} \rightarrow$ (as it is)

$\delta(q_1, b, b) \rightarrow (q_1, \epsilon)$

$\delta(q_1, a, a) \rightarrow (q_1, \epsilon)$

$\delta(q_1, \epsilon, z_0) \rightarrow (q_f, z_0)$



Designing a PDA when context free grammar is given :-

Method-1:- By using empty stack.

→ If L is a context free language, then we construct a PDA 'M' accepting 'L' by empty stack (or) empty store.

→ We construct M by making use of productions in 'G'

*Construction of PDA:-

Let $G = (V, T, P, S)$ is a context free grammar. We construct a PDA $M = (\{q\}, \Sigma, V \cup \Sigma, \delta, q, S, \phi)$ where δ is defined by the following rules,

Rule-1: $\delta(q, \epsilon, A) = \{(q, \alpha) \mid A \to \alpha \text{ is in } P\}$

Rule-2: $\delta(q, a, a) \Rightarrow \{(q, \epsilon) \mid \text{for every } a \text{ in } \Sigma\}$

↳ We can xplain the contruction in following way

- Here the pushdown symbols in M are variables and terminals.

Rule-1: If the PDA reads a variable A on the top of PDS, it makes an ϵ move by placing the R.H.S of any A prodn.

Rule-2: If the PDA reads a terminal a on the stack and if it matches with current input symbol, then the PDA erases a.

Q: S.T the set of all strings {a,b} consisting of equal no. of a's and b's accepted by a PDA.

Sol:

$$S \rightarrow aSbS \mid bSaS \mid \epsilon.$$

The PDA for above CFG is as follows:

$$M = (\{q\}, \Sigma, \underbrace{V \cup \Sigma}_{T}, \delta, q, S, \phi)$$

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

$V = \{S\}$

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

and δ is defined as follows:

Here we construct the PDA to accept the language generated by the grammer by empty stack.

$$\delta(q_0, \epsilon, S) \rightarrow (q_0, aSbS)$$

$$\delta(q_0, \epsilon, S) \rightarrow (q_0, bSbS)$$

$$\delta(q_0, \epsilon, S) \rightarrow (q_0, \epsilon)$$

$$\delta(q_0, a, a) \rightarrow (q_0, \epsilon)$$

$$\delta(q_0, b, b) \rightarrow (q_0, \epsilon)$$

$\delta(q_0, aabbab, S) \vdash \delta(q_0, aabbab, aSbS) \vdash \delta(q_0, abbab, SbS)$

Construct of CFG for a given PDA:—

If $M = \{Q, \Sigma, T, \delta, q_0, Z_0, F\}$ be a PDA then there exist a CFG G such that $L(G) = L(M)$ where $L(G)$ is a content free lang.

Construction of G:—

we define $G = \{V, T, P, S\}$ where $V =$

$$\{S\} \cup \{ \{q, z, q'\} / q, q' \in Q, z \in T\}$$

The prodns are induced by moves of PDA as follows:

R1: S productions are given by

$$S \; \cancel{\forall} \; q \in Q, \; S \to \{q_0, Z_0, q\}$$

R2: each move erasing a pushdown symbol given by $\delta(q, a, z) = (q', \epsilon)$

$$[q, z, q'] \to a$$

R3: each move not erasing a pushdown symbol is given by R_3: $\delta(q, a, z) = (q_1, Z_1 Z_2 \cdots Z_m)$

$$[q, z, q'] \to a\,[q_1, z, q_2]\,[q_2, z_2, q_3] \cdots [q_m, z_m, q']$$

state.

Q :- Construct a CFG which accepts M where M is $\{\{q_0, q_1, q_2\}, \{a, b, c\}, \{a, b, Z_0\}, \delta, q_0, Z_0, q_2\}$

where δ is given by $\delta(q_0, a, Z_0) = (q_0, aZ_0)$

$\delta(q_0, b, Z_0) = (q_0, bZ_0)$; $\delta(q_0, b, a) = (q_0, ba)$

$\delta(q_0, a, a) = (q_0, aa)$; $\delta(q_0, a, b) = (q_0, ab)$

$\delta(q_0, b, b) = (q_0, bb)$; $\delta(q_0, c, Z_0) = (q_1, Z_0)$

$\delta(q_0, c, a) = (q_1, a)$; $\delta(q_0, c, b) = (q_1, b)$

$\delta(q_1, a, a) = (q_1, \epsilon)$; $\delta(q_1, b, b) = (q_1, \epsilon)$

$\delta(q_1, \epsilon, Z_0) = (q_2, Z_0)$

$V = \{S, [q_0, a, q_0], [q_0, a, q_1], [q_0, a, q_2], [q_0, b, q_0], [q_0, b, q_1], [q_0, b, q_2], [q_0, Z_0, q_0], [q_0, Z_0, q_1], [q_0, Z_0, q_2] [q_1, a, q_0], [q_1, a, q_1], [q_1, a, q_2], [q_1, b, q_0], [q_1, b, q_1], [q_1, b, q_2], [q_1, Z_0, q_0], [q_1, Z_0, q_1], [q_1, Z_0, q_2], [q_2, a, q_0], [q_2, b, q_0]), [q_2, Z_0, q_0], [q_2, a, q_1], [q_2, b, q_1], [q_2, Z_0, q_1], [q_2, a, q_2], [q_2, b, q_2], [q_2, Z_0, q_2]\}$