UNIT – III

Context-Free Grammar.

→ The content free grammar can be formally defined as a set denoted by $G = \{V, T, P, S\}$ where V and T are set of non terminals and terminals respectively.

P → production rules.

$A \rightarrow \alpha$. $A \in V_N$ and $\alpha \in (V \cup T)^*$

S → start symbol.

Eg:- 1) $A \rightarrow \alpha$

$S \rightarrow Aa$ $B \rightarrow abc$

$A \rightarrow a$ $A \rightarrow \epsilon$.

2) $S \rightarrow S + S$

$S \rightarrow S * S$

$S \rightarrow (S)$

$S \rightarrow 4$

S → non terminal.

$*, +, (,), 4$ → terminals.

Derivation Trees:–

→ It is useful to display derivations as trees, i.e the graphical representation of derivation of string from a grammar is called as Derivation Tree (or) Parse tree.

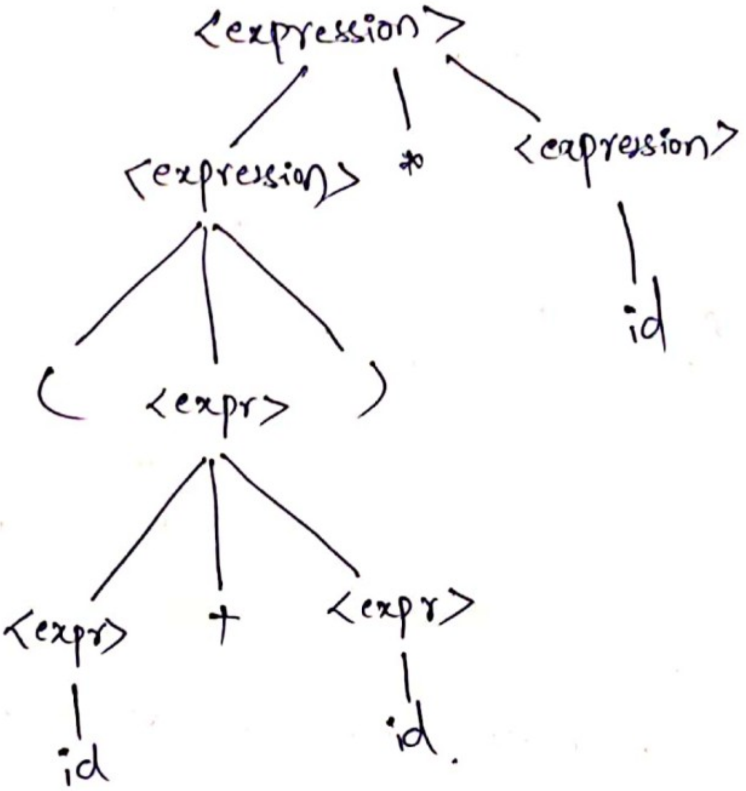
→ derivation or parse tree super impose structure on words of a lang that is useful in applications such as compilation of programming langs.

→ The vertices of a derivation tree are labled with terminals (or) variable symbols of the grammar (or) with ϵ.

Scanned by CamScanner

→ if an interior vertex n is labeled A and it's sons are labeled $x_1, x_2 \ldots x_k$ from the left, then the production must be $A \rightarrow x_1 x_2 \ldots x_k$. 3)

Eg. we have been the derivation (id+id)*id it's parse tree is given as.



Defination of Derivation tree (formally) :→

Let $G = (V, T, P, S)$ be a CFG. A tree is a parse-tree for G if

1) every vertex has a label, which is a symbol of $V \cup T \cup \{\epsilon\}$

2) The label of the root is 'S'.

Scanned by CamScanner

3) if a vertex is interior and has label 'A' then A must be in 'V.'

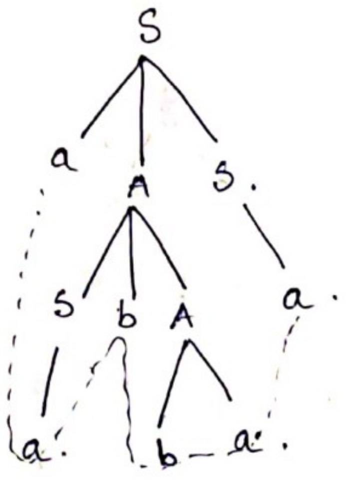
4) if n has label A & vertex $n_1, n_2, \ldots n_k$ are the sons of vertex n, in order from the left, with labels $X_1, X_2 \ldots X_k$ respectively then $A \rightarrow X_1 X_2 \ldots X_k$ must be a productions in p.

5) If vertex n has label 'ϵ', then n is a leaf and is the only son of its father.

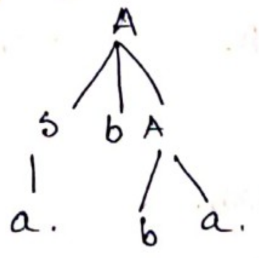
Eg: $S \rightarrow aAS \mid a$
$A \rightarrow SbA \mid SS \mid ba$.

derivation $S \Rightarrow aAS$
$\Rightarrow aSbAS$
$\Rightarrow aSbAa$.
$\Rightarrow aSbbaa$
$\Rightarrow aabbaa$.

This is S-tree.



$\Rightarrow$ subtree is A-tree.



$S \overset{*}{\Rightarrow} aabbaa$.

$A \overset{*}{\Rightarrow} abba$.

Scanned by CamScanner

Scanned by CamScanner

→ if we read the labels of the laves from left to right, we have a sentential form, such String is called as field of the derivation tree.

Let the α be the field of the derivation tree for the $G = (V, T, P, S)$ then $S \overset{*}{\underset{G}{\Rightarrow}} \alpha$ is the sentential form for it.

→ A subtree of a derivation tree is a particular vertex of the tree together with all it's decendants, and edges connecting them and their labels.

<u>Note</u>:- It looks just like derivation tree, but the root may not be the start symbol of the grammar.

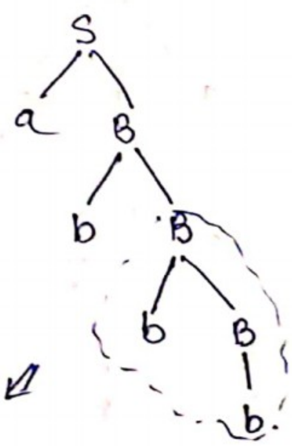
→ if variable 'A' labels the root, then we call the subtree as A-tree.

Eg. $S \rightarrow aA \mid aB$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b.$

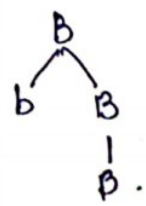
$S \Rightarrow aB$
$\Rightarrow abB$
$\Rightarrow abbB$
$\Rightarrow abbb.$



→ field is abbb.

→ subtree is B-tree

field of B-tree is bb.



Scanned by CamScanner

Left most and Rightmost derivations:-

→ if at each step in a derivation a production is applied to the leftmost variable then the derivation is (called) said to be a leftmost.

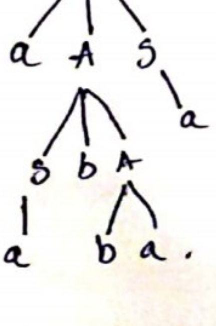
→ A derivation in which the right-most variable is replaced at each step is said to be right-most

→ if w is in L(G) for CFG, then w has atleast one parse tree and corresponding to a particular parse tree, w has a unique left-most and a unique right-most derivation which means.

w may have several right-most (or) left-most derivations since there may be more than one parse tree for w.

But from each parse tree, only one left-most & right most derivation may be obtained.

Eg:- consider the derivation tree



The leftmost derivation

$S \Rightarrow aAS$
$\Rightarrow aSbAS$
$\Rightarrow aabAS$
$\Rightarrow aabbaS$
$\Rightarrow aabbaa.$

The right-most derivation.

$S \Rightarrow aAS$
$\Rightarrow aAa$
$\Rightarrow aSbAa$
$\Rightarrow aSbbaa$
$\Rightarrow aabbaa.$

Scanned by CamScanner

(1) construct a CFG for the language having any number of a's over the set $\Sigma = \{a\}$.

sol. $r.e = a^*$

production rules. $S \to aS$
$S \to \epsilon$.

(2) Try to recognize the language L for given CFG.

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

where $P = \quad S \to aSb$
$S \to ab.$

sol:- $S \to aSb$
$\to aaSbb$
$\to aaabbb$ $\quad L = \{a^n b^n$ where $n \geq 1\}$.

(3) Construct the CFG for R.E $(0+1)^*$

$P = \quad S \to 0S \mid 1S$
$S \to \epsilon$.

(4) Construct a grammar for lang containing strings of at least two a's.

sol. $S \to AaAaA$
$A \to aA \mid bA \mid \epsilon$.

(5) construct a grammar generating $L = wcw^T$ where $w \in \{a, b\}^*$

sol:- $L = \{aca, aacaa, bcb, abcba, bacab, ---\}$

Scanned by CamScanner

$S \rightarrow aSa$
$S \rightarrow bSb$
$S \rightarrow c$

$L = wcw^T$ where $w \in (a+b)^*$

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

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

P can be $\Rightarrow$ $S \rightarrow aSa$
$S \rightarrow bSb$
$S \rightarrow a$
$S \rightarrow b$.
$S \rightarrow \epsilon$.

Eg. The string abaaba derived

$\Rightarrow S$
aSa
$abSba$
$abaSaba$
$aba\epsilon aba$
$abaaba$

(7). Construct CFG which consists of all the strings having atleast one occurence of 000.

$r.e = (0+1)^* 000 (0+1)^*$.

$S \rightarrow ATA$
$A \rightarrow 0A \mid 1A \mid \epsilon$.
$T \rightarrow 000$.

(8) Construct CFG for the language in which there are no consecutive b's, the strings may or may not have consecutive a's.

Sol: $S \rightarrow aS \mid bA \mid a \mid b \mid \epsilon$.
$A \rightarrow aS \mid a \mid \epsilon$.

Scanned by CamScanner

(9) Construct CFG for the language containing atleast one occurrence of double a.

Sol.

$S \rightarrow BAB$

$A \rightarrow aa$

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

$A \rightarrow aa \rightarrow$ double a's

Eg: $S \rightarrow BAB \rightarrow aBAB \rightarrow abBAB \rightarrow ab\overset{\epsilon}{B}AB$

$abAB \Rightarrow abaaB \rightarrow abaabB \Rightarrow abaabb$

(10) Construct CFG for the language containing all the strings of diff first and last symbols over $\Sigma = \{0,1\}$.

Sol. $r.e = [0(0+1)^*1 + 1(0+1)^*0]$

$S \rightarrow 0A1 \mid 1A0$

$A \rightarrow 0A \mid 1A \mid \epsilon$.

(11) Obtain a CFG to generate the set of all strings over alphabet $\{a,b\}$ with exactly twice as many a's as b's.

Sol. aba (or) aab (or) baa.

$S \Rightarrow abaS \mid aabS \mid baaS \mid \epsilon$.

(12) Find DFA and CFG for the following lang

$L = \{$odd-length strings in $\{a,b\}^*$ with middle symbol a$\}$.

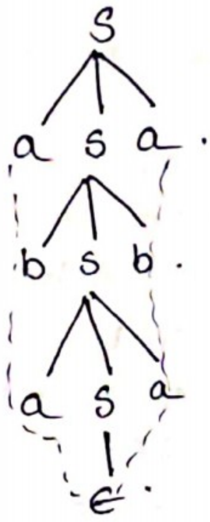
Sol. $\boxed{a \text{ (or) } b}^n \; a \; \boxed{a \text{ (or) } b}^n \quad n \geq 0$.

$r.e = (a+b)^n a (a+b)^n \quad n \geq 0$.

Productions $\Rightarrow$ $S \rightarrow aSa \mid aSb \mid bSb \mid bSa \mid a$.

Scanned by CamScanner

(1) Draw a derivation tree for the string abaaba for the CFG given by G where $P = S \to aSa$
$S \to bSb$
$S \to b \mid a \mid \epsilon$.

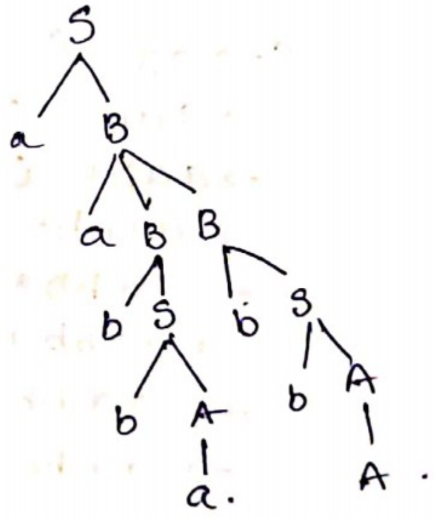


$\Rightarrow aba\epsilon aba \Rightarrow \underline{abaaba}$

(2) Construct the derivation tree for the string aabbabba from the CFG given by.

$S \to aB \mid bA$
$A \to a \mid aS \mid bAA$
$B \to b \mid bS \mid aBB$.

tree :->



derivation :- $S \to aB$
$a\boxed{aBB}$
$aa\boxed{bS}B$
$aab\boxed{bA}B$
$aabb\boxed{a}B$
$aabba\boxed{bS}$
$aabbab\boxed{bA}$
$aabbabb\boxed{a}$.

Scanned by CamScanner

(3) Consider the CFG $S \to 0B \mid 1A$
$A \to 0 \mid 0S \mid 1AA$
$B \to 1 \mid 1S \mid 0BB$.

The derivation trees for the string 001101

L.M.D ⇒ $S \to 0B$
$\to 00BB$
$\to 001SB$
$\to 0011AB$
$\to 00110B$
$\to 001101$

R.M.D ⇒ $S \to 0B$
$\to 00BB$
$\to 00B1$
$\to 001S1$
$\to 0011A1$
$\to 001101$

(4) Derive the string "aabbabba" for L.M.D + R.M.D using a CFG given by. $S \to aB \mid bA$
$A \to a \mid aS \mid bAA$
$B \to b \mid bS \mid aBB$.

Sol LMD:→

$S \to aB$ $(S \to aB)$
$\to aaBB$ $(B \to aBB)$
$\to aabB$ $(B \to b)$
$\to aabbS$ $(B \to bS)$
$\to aabbaB$ $(S \to aB)$
$\to aabbabS$ $(B \to bS)$
$\to aabbabbA$ $(S \to bA)$
$\to aabbabba$. $(A \to a)$.

R.M.D:→
$S \to aB$ $(S \to aB)$
$S \to aaBB$. $(B \to aBB)$
$S \to aaBbS$ $(B \to bS)$
$S \to aaBbbA$ $(S \to bA)$
$S \to aaBbba$ $(A \to a)$
$S \to aabSbba$ $(B \to bS)$
$S \to aabbAbba$ $(S \to bA)$
$S \to aabbabba$. $(A \to a)$

(5) Derive the string 1000111 for LM and RM.D using CFG. $G = \{V, T, P, S\}$ where.
$V = \{S, T\}$
$T = \{0, 1\}$
$P =$ $S \to T00T$
$T \to 0T \mid 1T \mid \epsilon$.

Scanned by CamScanner

(6) Find the L.M and R.M derivations for the word abba in the grammar $S \to AA$

$A \to aB$

$B \to bB \mid \epsilon$.

(7) Consider the grammar $E \to +EE \mid *EE \mid -EE \mid x \mid y$.
Find the leftmost & rightmost derivation for the string $+*-xyxy$ and write parse tree.

Scanned by CamScanner

Ambiguity in Context Free Grammar:—

→ A context-free grammar G such that some word has two parse-trees (which means more than one left-most derivation or more than one right-most derivation) is said to be "ambiguous"

→ A CFL for which every CFG is ambiguous is said to be an inherently ambiguous CFL.

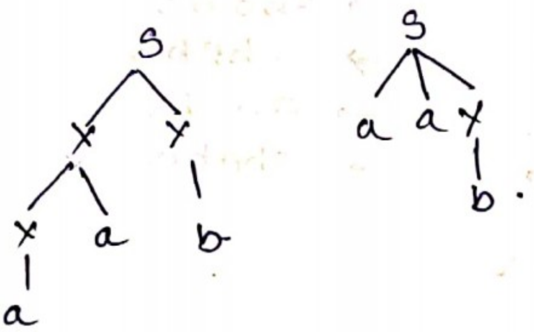
Examples:—

(1) $S \rightarrow XY \mid aaY$
$X \rightarrow Xa \mid a$
$Y \rightarrow b.$

Consider the string $w = aab \in L(G)$

$S \rightarrow XY$
$\rightarrow XaY$
$\rightarrow aaY$
$\rightarrow aab.$

$S \rightarrow aaY$
$\rightarrow aab.$



Hence the given grammar is ambiguous.

(2) $S \rightarrow aSbS \mid bSaS \mid \epsilon.$ consider a string abab.

Sol: $S \rightarrow aSbS.$
$\rightarrow a\epsilon bS$
$\rightarrow abS$
$\rightarrow abaSbS$
$\rightarrow aba\epsilon b\epsilon$
$\rightarrow abab.$

$S \rightarrow aSbS$
$\rightarrow aSb\epsilon.$
$\rightarrow aSb$
$\rightarrow abSaSb$
$\rightarrow ab\epsilon a\epsilon b$
$\rightarrow abab.$

Hence grammar is ambiguous.//

Scanned by CamScanner

③ $S \to bAc \mid bac$ $\quad w = bac.$

$A \to a$

$C \to c.$

$S \Rightarrow bAC.$ $\quad S \Rightarrow baC$

$\Rightarrow bac.$ $\quad S \Rightarrow bac.$

-Hence it is ambiguous.

④. $S \to S+S \mid S*S \mid a \mid b$ $\quad w = a+a*b.$

$S \Rightarrow S+S$

$a+S$

$a+S*S$

$a+a*S$

$a+a*b.$

$S \to S*S.$

$\to S+S*S$

$\to a+S*S$

$\to a+a*S$

$\to a+a*b$

-Hence ambiguous.

⑤. $S \to SbS \mid a.$ $\quad w = ababa.$

$S \Rightarrow SbS.$

$\Rightarrow abS$

$\Rightarrow abSbS$

$\Rightarrow ababS$

$\Rightarrow ababa.$

$S \Rightarrow SbS$

$\Rightarrow SbSbS$

$\Rightarrow abSbS$

$\Rightarrow ababS$

$\Rightarrow ababa$

-Hence ambiguous.

⑥ $S \to a \mid abSb \mid aAb$

$A \to bS \mid aAAb.$

$w = abab.$

$S \Rightarrow abSb.$

$\Rightarrow abab.$

$S \Rightarrow aAb.$

$\Rightarrow abSb$

$\Rightarrow abab$

-Hence ambiguous

Scanned by CamScanner

⑦ Consider the Grammar $G = (V, \Sigma, R, S)$

where $V = \{S, A\}$

$S \to AA$
$A \to AAA$
$A \to a$
$A \to bA$
$A \to Ab.$

Show that this is an ambiguous grammar.
(Consider the String babbab)

⑧ Prove that the following grammar is ambiguous on the String 'aab'

$S \to aS \mid aSbS \mid \epsilon.$

⑨. Show that the grammar is ambiguous.

$S \to a \mid SA \mid bSS \mid SSb \mid SbS.$

Scanned by CamScanner