Date.
Page.

Unit 4

Context Free Grammar

$\alpha \rightarrow \beta$

$|\alpha| = 1 \quad ; \quad \alpha \in V$

$\beta \in (V \cup T)^*$

$S \rightarrow AB$
$A \rightarrow aA \mid \varepsilon$
$B \rightarrow bB \mid \varepsilon$

It is CFG

String = Yield

Types of derivation

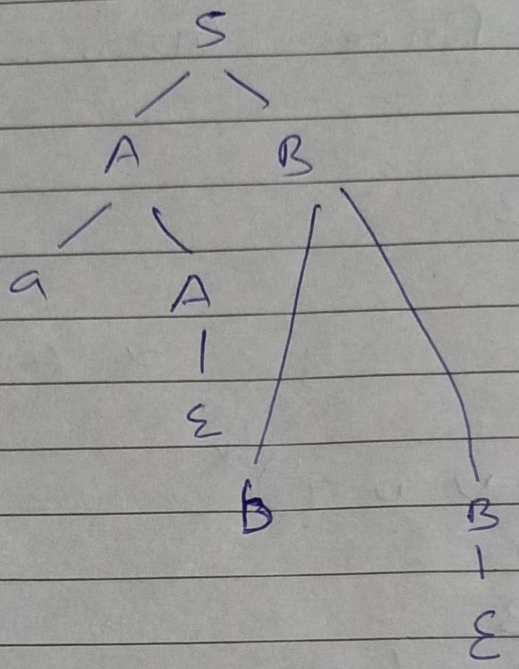
LMD
Leftmost Derivation

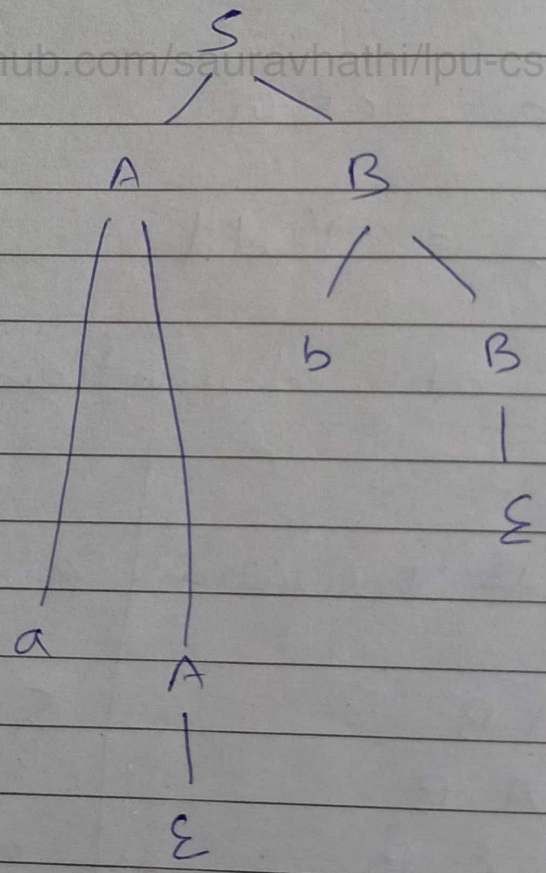
$S \rightarrow AB$
$\rightarrow aAB$
$\rightarrow a \cdot \varepsilon \cdot B$
$\rightarrow aB$
$\rightarrow abB$
$\rightarrow ab$

RMD
Rightmost Derivation

$S \rightarrow AB$
$\rightarrow AbB$
$\rightarrow Ab$
$\rightarrow aAb$
$\rightarrow ab$

Tree Derivation / Parse Tree





Ambiguity in CFG

Any CFG is called to be as ambiguous if there exist any string that could be driven by more than 1 LMD or RMD

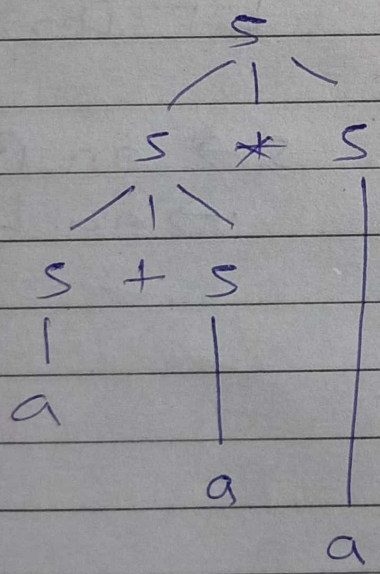
$S \rightarrow S+S / S*S / a$

LMD 1

$S \rightarrow S+S$
$\rightarrow a+S$
$\rightarrow a+S*S$
$\rightarrow a+a*S$
$\rightarrow a+a*a$

LMD 2

$S \rightarrow S*S$
$\rightarrow S+S*S$
$\rightarrow a+S*S$
$\rightarrow a+a*S$
$\rightarrow a+a*a$



$a+a*a$

$S \rightarrow SaS / \varepsilon$

LMD1

$S \rightarrow SaS$
$\rightarrow SaSaS$
$\rightarrow aa$

LMD2

$S \rightarrow SaS$
$\rightarrow \varepsilon aS$
$\rightarrow aSaS$
$\rightarrow aa$

Note ⇒ If in any production there exist both left and right recursions the grammar is always ambiguous

* $S \rightarrow AB / aaB$
$A \rightarrow a / Aa$
$B \rightarrow b$

LMD1

$S \rightarrow AB$
$\rightarrow Aa B$
$\rightarrow aa B$
$\rightarrow aab$

LMD2

$S \rightarrow aa B$
$\rightarrow aab$

Date
Page

Conversion from ambiguous to ~~un~~ unambiguous grammar.

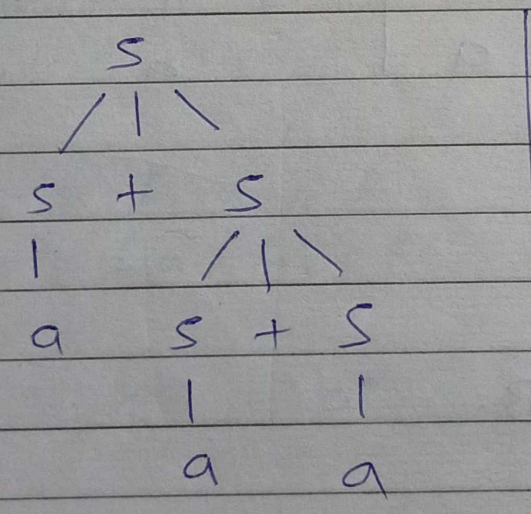
Ambiguity occurs due to 2 problems :-

(i) Associative problem

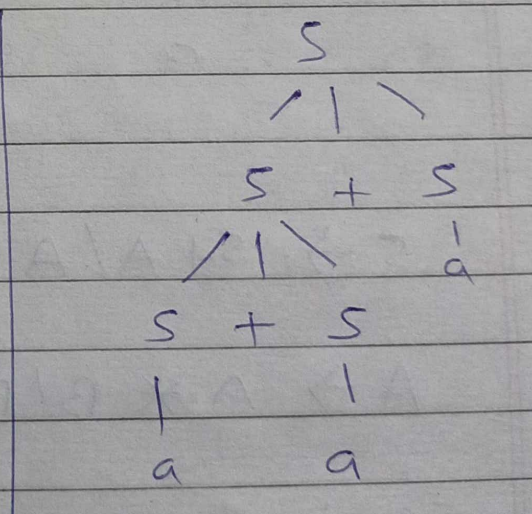
(ii) Precedence problem

(i) If there exist associative problem in ambiguous grammar then only left associativity is allowed, for which right recursion must be removed. For example consider the following grammar

$S \rightarrow S+S/a$



a + (a+a)



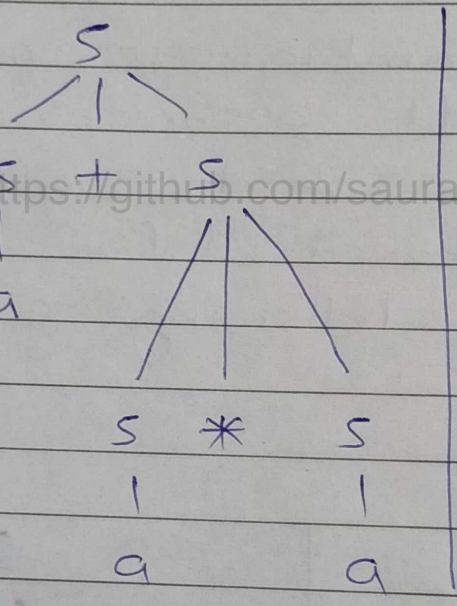
(a+a) + a

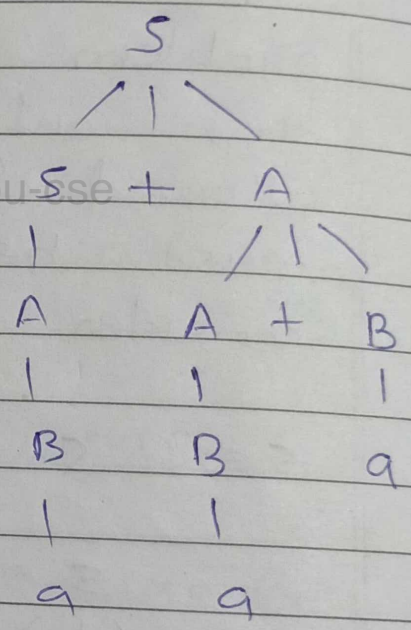
$S \rightarrow S+a/a$

(ii) Precedence problem

If there exist precedence problem then the parse tree with higher precedence at the bottom level of parse tree must be considered and right recursion must be removed.

$S \rightarrow S+S / S*S / a$





$S \rightarrow S+A / A$

$A \rightarrow A*B / B$

$B \rightarrow a$

Date.
Page.

Construction of CFG

(i) Construct CFG for the lang containing any no of as.

$S \rightarrow Sa / \varepsilon$

(ii) Any no of as, bs $(a+b)^*$

$S \rightarrow Sa / Sb / \varepsilon$

(iii) At least 00

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

$S \rightarrow A\,0\,A\,0\,A$
$A \rightarrow A0 / A1 / \varepsilon$

(iv) Create CFG for language with at least 1 occurrence of 'aaa'

$S \rightarrow A\,aaa\,A$
$A \rightarrow Aa / Ab / \varepsilon$

(v) $L = a^n . b^n \; ; \; n \geq 1$

$S \rightarrow asb / ab$

(vi) $L = W c W^R$, $W \in \{a, b\}^*$

$S \to a S a$

$S \to b S b$

$S \to c$

(vii) $a^n b^m c^n$; $n, m \geq 1$

$S \to a S c \mid a A c$

$A \to b A \mid b$

(viii) $G = (\{S\}, \{0, 1\}, \{S \to 0S1, S \to \wedge\}, S)$

$L(G)$ $0^n 1^n$, $n \geq 1$

(ix) $S \to aCa$ $C \to aCa \mid b$

$L(G) = a^n b a^n$

(x) $S \to aS \mid bS \mid a \mid b$

$(a + b)^+$

* Palindrome over (a, b)

$S \to aSa \mid a \mid \varepsilon$

$S \to bSb \mid b \mid \varepsilon$

Date
Page

* $L = \{wcw^T\}$

$S \rightarrow aSa / bSb / c$

* $L = \{a^n b^n c^i ; n \geq 1, i \geq 0\}$

$S \rightarrow aSbA / ab$

$A \rightarrow cA / \varepsilon$

* $L = \{a^j b^n c^n \quad j \geq 0 \quad n \geq 1\}$

$S \rightarrow AbSc / bc$

$A \rightarrow aA / \varepsilon$

Simplification of CFG :-

CFG should be simplified by removing null and unit production.

(i) Removing null production

Steps

(a) Identify the nullable variables from P.

(b) Substitute them one by one to produce P'.

(c) If null is part of language that is $\wedge \in L(G)$ then $\wedge$ cant be removed.

ex ⇒ P: $S \to aS/AB$
$A \to \wedge$
$B \to \wedge$
$C \to a$

Removal of $A \to \wedge$	Removal of $B \to \wedge$
$S \to aS/AB/B$	$S \to aS/AB/B/A/\varepsilon$
$B \to \wedge$	
$C \to a$	$C \to a$

$S \to aS/AB/B/A/\varepsilon$

ex ⇒ P: $S \to ABC$
$A \to a/bbD$
$B \to a$
$B \to \wedge$
$C \to b$ $\quad C \to \wedge$
$D \to c/d$

Date
Page

Removal of $B \rightarrow \wedge$

$S \rightarrow ABC \mid AC$
$A \rightarrow a \mid bbD$
$B \rightarrow a$
$C \rightarrow b \qquad C \rightarrow \wedge$
$D \rightarrow c \mid d$

Removal of $C \rightarrow \wedge$

$S \rightarrow ABC \mid AC \mid AB \mid A$
$A \rightarrow a \mid bbD$
$B \rightarrow a$
$C \rightarrow b$
$D \rightarrow c \mid d$

(iii) Removal of unit production

Steps

(a) Identify all unit productions

(b) Replace B with its terminal where $\alpha = B$

(c) Removal of all unreachable productions.

ex =>

P: $S \to AB$
$A \to a$
$B \to c / b$
$C \to D$
$D \to E$
$E \to a$

$B \to C \to D \to E \to a$

$S \to AB$
$A \to a$
$B \to b$
$E \to a$

$B \to a$
$C \to a$
$D \to a$

final answer

$S \to AB$
$A \to a$
$B \to b / a$

$S \to ab / aa$

ex =>

P: $S \to aA / B$
$A \to ba / bb$
$B \to A / bba$

$S \to B \to A$

$S \to aA / B$
$B \to ba / bb / bba$
$A \to ba / bb$

Date. /
Page /

S → aA / B ba / bb / bba
A → ba / bb

* S → aS / A
A → ε

S → aS / ε

* S → ABC
A → aA / ε
B → bB / ε
C → aa

S → ABC / BC	S → ABC / BC / AC / C
A → a	A → a
B → bB / ε	B → b
C → aa	C → aa

* Remove unit production.

S → ABC / aA
A → aa / B
B → C / bba / ab
C → a

A → B → C → a

Date. |
Page. |

S → ABC / aA
A → aa / a / bba / ab
B → a / bba / ab
C → a

* Check ambiguity

S → a / abSb / aAb
A → bS / aAAb

S → abSb
→ ababS

S → aAb
S → abSb
→ ababS

Ambiguous

* S → aB / ab
B → ABb / b
A → aAB / a

S → aB
→ ab

S → ab

Ambiguous

Q) What is sum of indegree and outdegree over a state in DFA
$\Sigma = \{a, b, c\}$

=> Can't be determined

Q) What is the sum of minimum and maximum no of final state in DFA with n+1 total states.

=> n+2

* Normal forms of CFG

There are 2 types of normal forms

(i) CNF (Chomsky Normal Form)

(ii) GNF (Griebach Normal Form)

CNF :-

(a) Any production which is in the form of $A \to a$ or $A \to BC$ is in CNF.

(b) There should not be any unit or null productions however starting symbol may produce null.

$S \rightarrow \varepsilon$

(c) No of steps required to form a string of length n is $2n-1$

(d) The parse tree is similar to binary tree (always)

(e) The productions are restricted (always)

GNF :-

(a) Any production which is in the form of

$A \rightarrow aV^*$

where $a \in T$

$V^* \in V_n^*$

(b) There should not be any unit or null production however starting symbol may produce null

$S \rightarrow \varepsilon$

(c) No of steps required to form a string of length n is n.

(d) The parse tree is similar to binary tree (always)

(e) The productions are non-restricted.

ex-> $S \to aAB$ → GNF
$A \to a$ → CNF / GNF
$B \to b/AB$ → CNF
↳ CNF / GNF

(a) CFG (b) CNF (c) GNF

Conversion from CFG to GNF

(a) There should not be null or unit productions in the Grammar.

(b) Eliminate terminals on RHS by replacing them with another variables also include all forms of

$A \to a$ or $A \to BC$

in P'

(c) Restrict the no of variable by considering

$A \rightarrow A_1 A_2 A_3 \text{ ---- } A_m \quad ; \quad m \geq 3$

then the new productions will be

$A \rightarrow A_1 C_1$
$C_1 \rightarrow A_2 C_2$
$C_2 \rightarrow A_3 C_3$ ------

Also include all the CNF productions in P'.

* $S \rightarrow aAD$
$A \rightarrow aB / bAB$
$B \rightarrow b$
$D \rightarrow d$

Step 1

Step 2

$S \rightarrow aAD \Rightarrow S \rightarrow CaAD \quad ; \quad Ca \rightarrow a$

$A \rightarrow aB \Rightarrow A \rightarrow CaB$

$A \rightarrow bAB \Rightarrow A \rightarrow CbAB \quad ; \quad Cb \rightarrow b$

Date
Page

Step 3

$S \rightarrow C_a AD$

$S \rightarrow C_a X \quad ; \quad X \rightarrow AD$

$A \rightarrow C_b AB$

$A \rightarrow C_b Y \quad ; \quad Y \rightarrow AB$

P'

$B \rightarrow b$
$D \rightarrow d$
$C_a \rightarrow a$
$C_b \rightarrow b$
$A \rightarrow C_a B$

$S \rightarrow C_a X$
$X \rightarrow AD$
$A \rightarrow C_b Y$
$Y \rightarrow AB$

* Elimination of left recursion

$A \Rightarrow A\alpha / \beta$

$A \rightarrow BA'$ and $A' \rightarrow \alpha A' / \varepsilon$

ex => $S \rightarrow S+T \mid T$

A α β

$S \rightarrow TZ$

$Z \rightarrow +TZ \mid \varepsilon$

ex => $S \rightarrow a \mid b \mid S * T$

$S \rightarrow S*T \mid a \mid b$

$S \rightarrow aZ \mid bZ$

$Z \rightarrow *TZ \mid \varepsilon$

* Conversion from CFG to GNF

(a) Remove all the unit or null productions if there are any.

(b) CFG must be in CNF if not then convert it into CNF

(c) Change the names of all the non-terminals into Ai in the increasing order of i

(d) If the production is of the form $A_i \rightarrow A_j X$ then after the productions where $j < i$

(e) Remove all the left recursions.

ex =>

$S \rightarrow XA \mid BB$
$B \rightarrow b \mid SB$
$X \rightarrow a$
$A \rightarrow a$

Step 1 ✓

Step 2 ✓

Step 3

$S \rightarrow A_1$ $\quad X \rightarrow A_2$
$A \rightarrow A_3$ $\quad B \rightarrow A_4$

$A_1 \rightarrow A_2 A_3 \mid A_4 A_4$
$A_4 \rightarrow b \mid A_1 A_4$
$A_2 \rightarrow a$
$A_3 \rightarrow a$

Step 4

$A_4 \rightarrow A_1 A_4$

$A_4 \rightarrow A_2 A_3 A_4 \;/\; A_4 A_4 A_4$

$A_4 \rightarrow a A_3 A_4 \;/\; A_4 A_4 A_4$

$A_1 \rightarrow A_2 A_3 \;/\; A_4 A_4$

$A_4 \rightarrow b \;/\; a A_3 A_4 \;/\; A_4 A_4 A_4$

$A_2 \rightarrow a$

$A_3 \rightarrow a$

Step 5

$A \rightarrow A_4 \qquad \alpha \rightarrow A_4 A_4 \qquad \beta \rightarrow b / a A_3 A_4$

$A_4 \rightarrow bZ \;/\; a A_3 A_4 Z$

$Z \rightarrow A_4 A_4 Z \;/\; \varepsilon$

$A_4 \rightarrow bZ \;/\; a A_3 A_4 Z \;/\; b \;/\; a A_3 A_4$

$Z \rightarrow A_4 A_4 Z \;/\; A_4 A_4$

Conversion from CFG to PDA

→ Every CFG could be converted into PDA and vice-versa by following rules.

(i) For every non-terminal

$(q, \varepsilon, A) = (q, B)$

where $A \to B$

(ii) For every terminal 'a'

$(q, a, a) = (q, \varepsilon)$

ex ⇒ $S \to 0S1 / 00 / 11$

$S01^n \Rightarrow$

$(q, \varepsilon, S) \to (q, 0S1)$

$(q, \varepsilon, S) \to (q, 00)$

$(q, \varepsilon, S) \to (q, 11)$

$(q, 0, 0) \to (q, \varepsilon)$

$(q, 1, 1) \to (q, \varepsilon)$