Unit-3. Context free Grammar

* Grammar

Language generator.

Components

1. Variables
2. Terminals
3. Production
4. Start State

1. Variables
   To store data and can be replaced.
   Represented with capital letters.
2. Terminals
   End point of production.
   Represented with small letters.
3. Production
   give new value / state.
   Represented with '→'.
4. Start state
   First production.
   Represented with 'S'.

Representation of Grammar

G = (V, T, P, S)

Note:

Terminals — keywords (if, else if, for, while, do while)
digits (0 - 9)
symbols (+, -, *, /)
Lowercase letters (a - z)

Chomsky hierarchy of Generative grammar / Types of grammar

Noam Chomsky — founder

1. Type 0 grammar (Phrase structure)
2. Type 1 grammar (Context sensitive grammar)
3. Type 2 grammar (Contex free grammar)
4. Type 3 grammar (Regular grammar

1. Type 0 grammar

- A grammar $G = (V, T, P, S)$ is said to type 0 if all the productions are of the form $\alpha \to \beta$ where,

  $\alpha \in (V \cup T)^+$ and $\beta \in (V \cup T)^*$

- No restrictions on length
- β can have ϵ, ϵ can appear on RHS.
- α cannot have ϵ, LHS cannot be ϵ.

Example –
$S \to aAb \mid \epsilon$
$aA \to bAA$
$bA \to a$

2. Type 1 grammar

- A grammar $G = (V, T, P, S)$ is said to be type 1 if the productions are of the form $\alpha \to \beta$ where,

  $\alpha \in (V \cup T)^+$ and $\beta \in (V \cup T)^+$

- Length of β must be atleast the length of α. $|\beta| \geq |\alpha|$
- ϵ cannot be on LHS as well as RHS $\Rightarrow$ ϵ – free grammar

~~RHS ⇒ ∅~~

Example –
$S \to aAb$
$aA \to bAA$
$bA \to aa$

2. Type 2 grammar

- A grammar $G = (V, T, P, S)$ is said to be type 2 if all productions are of the form $A \to \alpha$ where

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

- LHS should be variables or Non-Terminals.
- RHS can be ϵ.

Example –
$S \to aB \mid bA \mid \epsilon$
$A \to aB$
$B \to bA \mid \epsilon$

Type 3 grammar

- A grammar $G = (V, T, P, S)$ is said to be type3 if grammar is said to be right linear or left linear.

Example –

$A \rightarrow wB$ (right linear)

$A \rightarrow Bw$ (left linear)

- Right linear – If NT (non-terminals) or variables is present towards the RHS

Example –

$S \rightarrow aaB \mid bbA \mid \epsilon$

$A \rightarrow aA \mid b$

$B \rightarrow bB \mid a \mid \epsilon$

- Left linear – If NT or variables appears towars LHS.

Example –

$S \rightarrow Baa \mid Abb \mid \epsilon$

$A \rightarrow Aa \mid b$

$B \rightarrow Bb \mid a \mid \epsilon$

---

* Design Language from CFG

1. $G = (V, T, P, S)$

$V = \{S\}$

$T = \{a, b\}$

$P = \{S \rightarrow aS \mid b\}$

$S = \{S\}$

→ $S \rightarrow aS$ $\quad$ $S \rightarrow b$

$S \rightarrow aaS$ $\quad$ $S \rightarrow aS$

$S \rightarrow aaaS$ $\quad$ $S \rightarrow ab$

$S \rightarrow aaab$

$S \rightarrow a^*b$

$L(G) = a^*b \mid ab$

2. $G = (V, T, P, S)$

$V = \{S\}$

$T = \{a\}$

$S = \{S\}$

$P = \{S \rightarrow aS \mid \epsilon\}$

→ $S \rightarrow aS$
$S \Rightarrow a\epsilon$
$S \Rightarrow a$

$S \rightarrow aS$
$S \rightarrow aaS$
$S \Rightarrow aa\epsilon$
$S \Rightarrow aa$

$S \rightarrow aS$
$S \rightarrow aaS$
$S \rightarrow aaaS$
$S \rightarrow aa--S$
$S \rightarrow aaa---aa$

i.e., $S \rightarrow a^*\epsilon$
$\Rightarrow a^*$

$\therefore L(G) = a^*$

3. $V = \{S\}$

$T = \{a\}$

$S = \{S\}$

$P \{S \rightarrow aaS \mid aa\}$

→ $S \Rightarrow aaS$
$\rightarrow aaaa$

$S \rightarrow aaS$
$\rightarrow aaaaS$
$\rightarrow aaaaaaS$
$\rightarrow aaaaaaaa$

i.e, $S \rightarrow (aa)^+$
$L(G) = (aa)^+$

4. $G = (\{S, T\}, \{a, b, c\}, P, S)$

$P = \{S \rightarrow aT,\ T \rightarrow bbT \mid c\}$

→ $S \rightarrow aT$
$\rightarrow abbT$
$\rightarrow abbc$

$S \rightarrow aT$
$\rightarrow abbT$
$\rightarrow abbbbT$
$\rightarrow abbbbbbT$
$\rightarrow abbbbbbc$

ie, $S \rightarrow a(bb)^+c$
$\therefore L(G) = a(bb)^+c$

5. $S \to aCa$

$C \to aCa \mid b$

$\to$ $S \to aCa$

$\to aaCaa$

$\to aaaCaaa$

$\to aaabaaa$

$S \to aCa$

$\to aba$

i.e., $S \to a^+ b a^+$

$\therefore L(G) = a^+ b a^+$

* Design CFG from FA

1. 

$\to$ Transitions:

$\delta(S, a) = S$

grammar:

$S \to aS$

$\because$ S is final state

$S \to \epsilon$

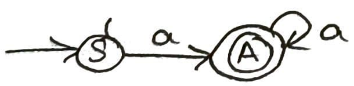
$G = (V, T, P, S)$

$V = \{S\}$

$T = \{a\}$

$P = \{S \to aS \mid \epsilon\}$

$S = \{S\}$

2. 

$\to$ Transitions:

$\delta(S, a) = A$

$\delta(A, a) = A$

grammar:

$S \to aA$

$A \to aA$

$\because$ A is final state

$A \to \epsilon$

The grammar is defined as,

$G = (V, T, P, S)$

$V = \{S, A\}$

$T = \{a\}$

$P = \{S \to aA, A \to aA \mid \epsilon\}$

$S = \{S\}$

3.

→ Transitions:

$\delta(S,a)=A$
$\delta(S,b)=S$
$\delta(A,a)=A$
$\delta(A,b)=A$

∵ A is final state

Grammar:

$S \to aA$
$S \to bS$
$A \to aA$
$A \to bA$
$A \to \epsilon$

The grammar is defined as:

$G=(V,T,P,S)$

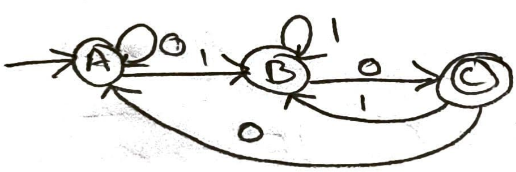
$V=\{S,A\}$

$T=\{a,b\}$

$P=\{S \to aA \mid bS,\ A \to aA \mid bA \mid \epsilon\}$

$S=\{S\}$

4.



→ Transitions:

$\delta(A,0)=A$
$\delta(A,1)=B$
$\delta(B,0)=C$
$\delta(B,1)=B$
$\delta(C,0)=A$
$\delta(C,1)=B$

∵ C is the final state

Grammar:

$A \to 0A$
$A \to 1B$
$B \to 0C$
$B \to 1B$
$C \to 0A$
$C \to 1B$
$C \to \epsilon$

The grammar is defined as:

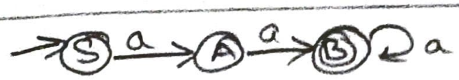
$G=(V,T,P,S)$

$V=\{A,B,C\}$

$T=\{0,1\}$

$P=\{A \to 0A \mid 1B,\ B \to 1B \mid 0C,\ C \to 0A \mid 1B \mid \epsilon\}$

$S=\{A\}$

5. 

→ Transitions:

$\delta(S,a) = A$
$\delta(A,a) = B$
$\delta(B,a) = B$
$\because$ B is final state

grammar:
$S \to aA$
$A \to aB$
$B \to aB$
$B \to \epsilon$

The grammar is defined as:

$G = (V, T, P, S)$
$V = \{S, A, B\}$
$T = \{a\}$
$P = \{S \to aA,\ A \to aB,\ B \to aB \mid \epsilon\}$
$S = \{S\}$

---

* Passing / Derivation

The sequence of substitutions used to obtain a particular string is called passing / derivation.

$S \to X\alpha Y$
$\alpha \to \beta$
$\therefore S \to X\beta Y$

Note:
Substitute for variables only.

Types of Substitution.
1. Left most derivation
2. Right most derivation

1. LMD
Productions are applied from the left most production.

Example –

$S \to XYX$
$X \to a$
$y \to a$

To derive aba

S → XYX
→ aYX
→ abX
→ aba

2. RMD

S → XYX ; X → a , Y → b
S → XYa
→ Xba
→ aba

1. S → AB
A → aaA | ϵ
B → Bb | ϵ
To derive: aab

LMD:

S → AB
→ aa B (A → aaA)(A → ϵ)
→ aa Bb (B → ϵ)
→ aab

RMD:

S → AB (B → Bb)
S → ABb (B → ϵ)
S → Ab (A → aaA)
S → aaAb (A → ϵ)
S → aab

2. S → aB | bA
A → a | aS | bAA
B → b | bS | aBB
To derive : aaabbabbba

→ LMD:

$S \rightarrow aB$ $(B \rightarrow aBB)$
$\rightarrow aaBB$ $(B \rightarrow aBB)$
$\rightarrow aaaBB$ $(B \rightarrow bS)$
$\rightarrow aaabS$ $(S \rightarrow bA$
$\rightarrow aaabBA$ $(A \rightarrow aS)$
$\rightarrow aaabbaS$ $(S \rightarrow bA)$
$\rightarrow aaabbabA$ $(A \rightarrow bAA)$
$\rightarrow aaabbabbAA$ $(A \rightarrow bAA)$
$\rightarrow aaabbabbbA$ $(A \rightarrow a)$
$\rightarrow aaabbabbba$

RMD:

$S \rightarrow aB$ $(B \rightarrow aBB)$
$\rightarrow aaBB$ $(B \rightarrow aBB)$
$\rightarrow aaBaBB$ $(B \rightarrow bS)$
$\rightarrow aaBaBbS$ $(S \rightarrow bA)$
$\rightarrow aaBaBbbA$ $(A \rightarrow a)$
$\rightarrow aaBaBbba$ $(B \rightarrow b)$
$\rightarrow aaBabbba$ $(B \rightarrow aBB)$
$\rightarrow aaaBBbbba$ $(B \rightarrow b)$
$\rightarrow aaaBbabbba$ $(B \rightarrow b)$
$\rightarrow aaabbabbba$

3. $S \rightarrow aAB$
$A \rightarrow bBb$
$B \rightarrow A \mid \epsilon$
To derive: abbbb

→ LMD:

$S \rightarrow aAB$ $(A \rightarrow bBb)$
$\rightarrow abBbB$ $(B \rightarrow A)$
$\rightarrow abAbB$ $(A \rightarrow aBb)$
$\rightarrow abbBbbB$ $(B \rightarrow \epsilon)$
$\rightarrow abbbbB$ $(B \rightarrow \epsilon)$
$\rightarrow abbbb$

RMD:

$S \rightarrow aAB \; (B \rightarrow A)$

$\rightarrow aAA \; (A \rightarrow bBb)$

$\rightarrow aAbBb \; (B \rightarrow \epsilon)$

$\rightarrow aAbb \; (A \rightarrow bBb)$

$\rightarrow abBbbb \; (B \rightarrow \epsilon)$

$\rightarrow abbbb$

4. $G = (\{S, B\}, \{c, d\}, P, S)$

$P = \{S \rightarrow cBS \mid c$

$B \rightarrow SdB \mid SS \mid dc\}$

To derive: ccdcddcc

LMD:

$S \rightarrow cBS \; (B \rightarrow SdB)$

$\rightarrow cSdBS \; (S \rightarrow c)$

$\rightarrow ccdBS \; (B \rightarrow SdB)$

$\rightarrow ccdSdBS \; (S \rightarrow c)$

$\rightarrow ccdcdBS \; (B \rightarrow dc)$

$\rightarrow ccdcddcS \; (S \rightarrow c)$

$\rightarrow ccdcddcc$

RMD:

~~S → cBS (S → cBS)~~

~~S → cBcBS (S → c)~~

~~→ cBcBc (B → dc)~~

~~→ cBcdcc (B → cBS)~~

~~→ ccBScdcc (S →~~

* Parse Tree / Derivation Tree

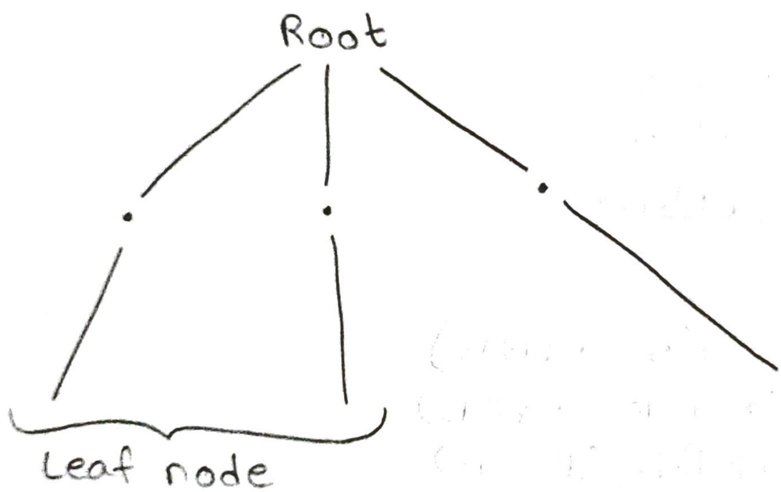
The derivation or parsing process in CFG if represented in the form of a tree.

Root which labelled as 'S'.

Label of internal vertex is 'variable'.

Each vertex can be either variable / Terminal / ε

The leaf node should always be a terminal.



1. $E \to E+E$

$E \to E*E$

$E \to id$

$V = \{E\}$, $T = \{id\}$, $S = \{E\}$

$W = id + id * id$

→ RMD:

$E \to E+E$

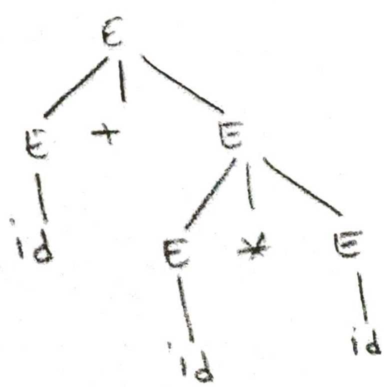
$\to E+E*E$

$\to E+E*id$

$\to E+id*id$

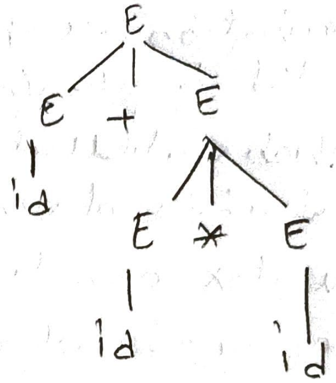
$\to id+id*id$

Parse tree:



RMD:

Parse tree:



$E \rightarrow E+E$

$\rightarrow id+E$

$\rightarrow id+E*E$

$\rightarrow id+id*E$

$\rightarrow id+id*id$

2. $S \rightarrow aB \mid bA$

$A \rightarrow a \mid aS \mid bAA$

$B \rightarrow b \mid bS \mid aBB$

$W = aaabbabbba$

→ LMD:

$S \rightarrow aB$ $(B \rightarrow aBB)$

$\rightarrow aaBB$ $(B \rightarrow aBB)$

$\rightarrow aaaBBB$ $(B \rightarrow b)$

$\rightarrow aaabBB$ $(B \rightarrow b)$

$\rightarrow aaabbB$ $(B \rightarrow aBB)$

$\rightarrow aaabbaBB$ $(B \rightarrow b)$

$\rightarrow aaabbabB$ $(B \rightarrow bS)$

$\rightarrow aaabbabbS$ $(S \rightarrow bA)$

$\rightarrow aaabbabbbA$ $(A \rightarrow a)$

$\rightarrow aaabbabbba$

Parse Tree: