Lecture ten

Topics that must be covered in this lecture:

- ***Derivation.***
- ***Context Free Grammar.***
- ***Context Free Language.***
- ***Derivation trees.***
- ***The ambiguous context free grammars.***

Derivation

The set of all strings that can be derived from a grammar is said to be the LANGUAGE generated from that grammar.

Examples1: consider the grammar G1=({S,A},{a,b},S,{S→AB, A→a, B→b}), find language generated from G1.

Solution:

S → AB
→ aB
→ ab

L(G1)={ab}

Examples 2: consider the grammar G2=({S,A},{a,b},S,{S→AB, A→aA|a, B→bB|b}), find language generated from G2.

Solution:

S →AB	S → AB	S → AB	S → AB
→ aB	→ aAbB	→ aAb	→ abB
→ ab	→ aabB	→ aab	→ abb
	→ aabb		

L(G2)= $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$

CONTEXT FREE GRAMMAR

A *context free grammar*, called **CFG**, is defined by 4 tuples as G=(V, Σ, S, P) where:

1. V= set of variables or Non-Terminal symbols
2. Σ= Set of Terminal symbols
3. S = start symbol
4. P = production rule

CFG has production rule of the form:
A→α
Where: α ={V ∪ Σ}* , A∈ V and |A|=1

Context Free Language

Definition: The language generated by the CFG is the set of all strings of terminals that can be produced from the start symbol S using the production as substitutions.
A language generated by the CFG is called a **context free language** (**CFL**).
The set of all CFL is identical to the set of languages accepted by Pushdown automata.
Example for generating a language: that generates equal number of a's and b's in the form a^nb^n, the context free grammar will be defined as:
G=({S,A},{a,b}, {S→aAb, A→aAb|ε})
Sol:
S→aAb
→aaAbb
→aaaAbbb
→aaabbb
→a^3b^3 ⇒ a^nb^n

Example about CFG

Example1

Let the only terminal be a.
Let the only nonterminal be S.
Let the production be:
S → aS
S → λ
The language generated by this CFG is exactly a*. In this language we can have the following derivation: S → aS → aaS → aaaS → aaaaS → aaaaaS → aaaaa λ = aaaaa

Example2

Let the only terminal be a.

Let the only nonterminal be S.

Let the production be:

S → SS

S → a

S → λ

The language generated by this CFG is also just the language a*.

In this language we can have the following derivation:

S → SS → SSS → SaS → SaSS → λ aSS → λ aaS → λ aa λ = aa

Example3

Let the terminals be a, b. And the only nonterminal be S.

Let the production be:

S → aS

S → bS

S → a

S → b

The language generated by this CFG is $(a+b)^+$.

In this language we can have the following derivation:

S → bS → baS → baaS → baab

Example4

Let the terminals be a, b. And the only nonterminal be S.

Let the production be:

S → aS

S → bS

S → λ

The language generated by this CFG is (a+b)*.

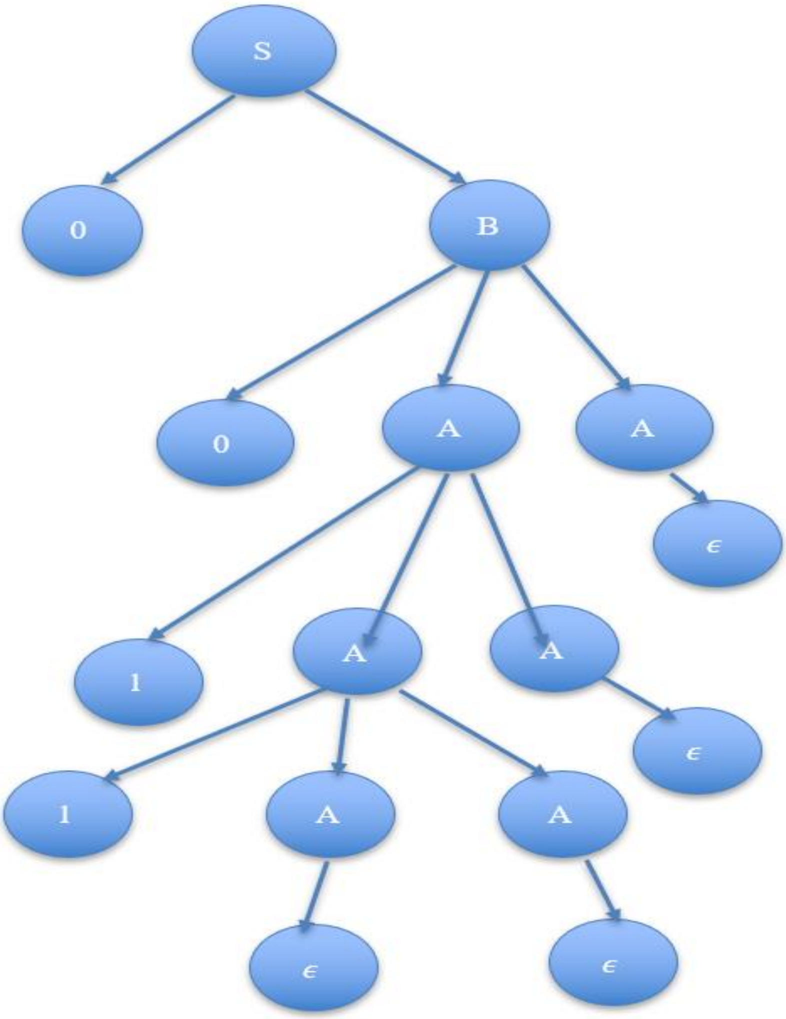
In this language we can have the following derivation:

S → bS → baS → baaS → baa λ =baa

Derivation Tree

Derivation Tree or Parse tree is an ordered rooted tree that graphically represents the semantic information of strings derived from a context free grammar.

Example: For the grammar G=(V,T,S,P) where $S \rightarrow 0B$, $A \rightarrow 1AA|\epsilon$, $B \rightarrow 0AA$, find derivation tree for string 0011 .



Note:

- **Root** vertex must be labeled by the start symbol.
- **Vertex** labeled by non-terminal symbols.
- **Leaves** labeled by terminal symbol or ϵ.
- **Types of derivation tree there are two types of derivation tree:**

Left derivation tree
Is obtained by applying production to the leftmost variable in each step.

Right derivation tree
is obtained by applying production to the rightmost variable in each step.

Example:

Derivation Order

1. $S \to AB$ 2. $A \to aaA$ 4. $B \to Bb$

3. $A \to \lambda$ 5. $B \to \lambda$

Leftmost derivation:

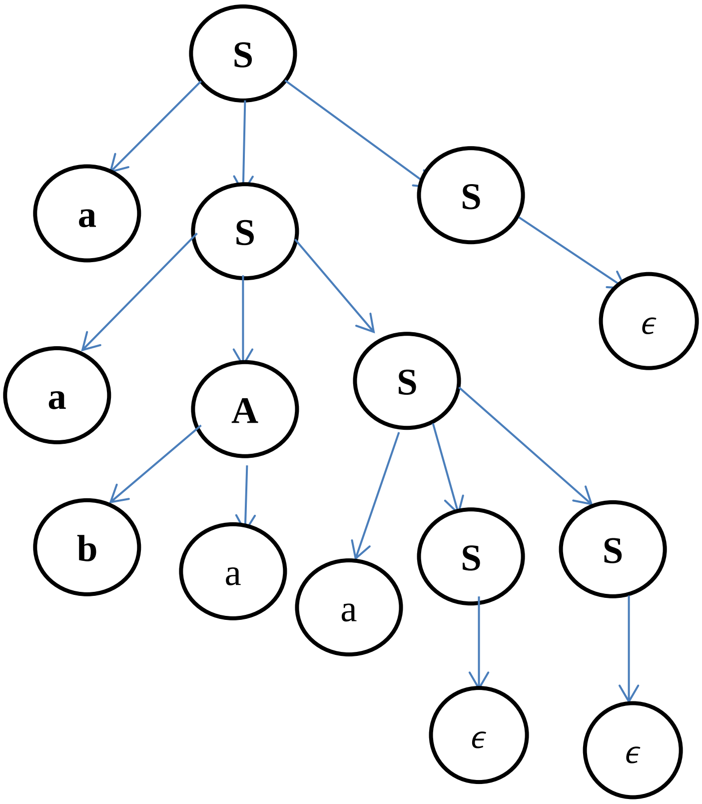
$$S \overset{1}{\Rightarrow} AB \overset{2}{\Rightarrow} aaAB \overset{3}{\Rightarrow} aaB \overset{4}{\Rightarrow} aaBb \overset{5}{\Rightarrow} aab$$

Rightmost derivation:

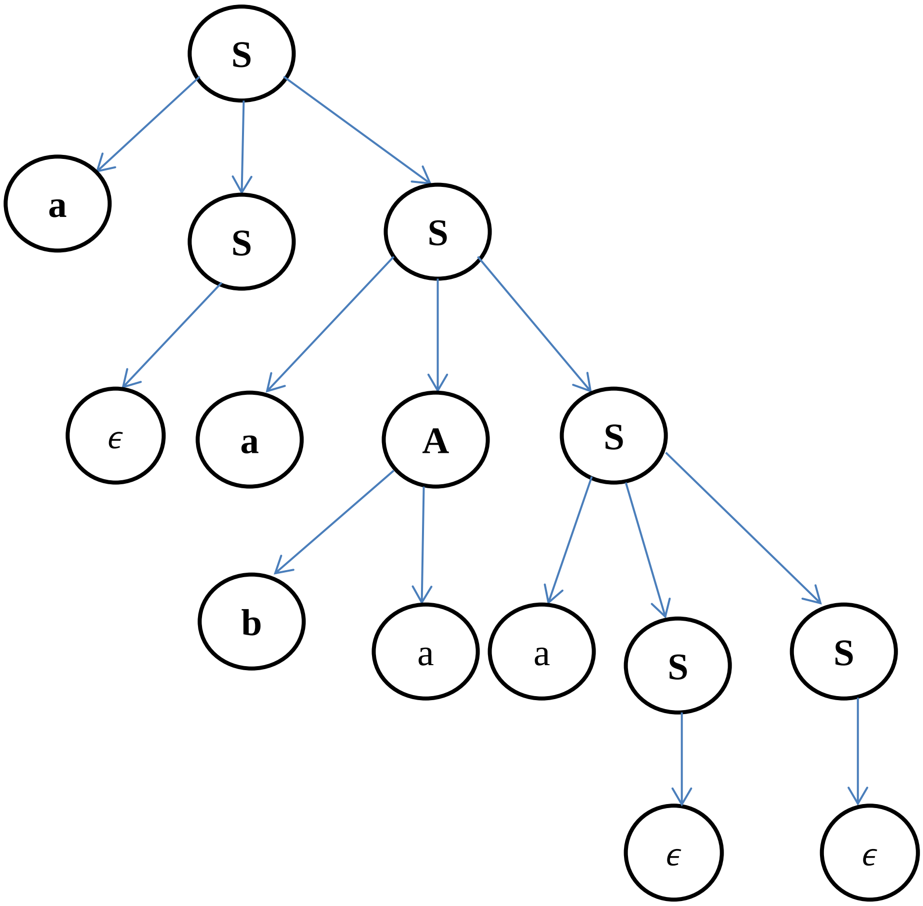
$$S \overset{1}{\Rightarrow} AB \overset{4}{\Rightarrow} ABb \overset{5}{\Rightarrow} Ab \overset{2}{\Rightarrow} aaAb \overset{3}{\Rightarrow} aab$$

Example for generationg the string aabaa from the grammar S→aAS|aSS| ϵ, A→SbA|ba, find left derivation tree and right derivation tree for derive aabaa.

<u>Left derivation tree :</u>



<u>Right derivation tree:</u>



Ambiguity:

A context-free grammar G is **ambiguous**

if some string $w \in L(G)$ has:

two or more leftmost derivations
(or rightmost)

Example1:

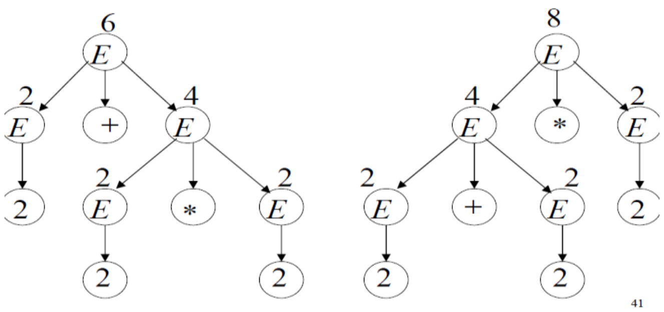
The grammar $E \to E + E \mid E * E \mid (E) \mid a$
is ambiguous:

string $a + a * a$ has two leftmost derivations

$$E \Rightarrow E + E \Rightarrow a + E \Rightarrow a + E * E \Rightarrow a + a * E \Rightarrow a + a * a$$

$$E \Rightarrow E * E \Rightarrow E + E * E \Rightarrow a + E * E \Rightarrow a + a * E \Rightarrow a + a * a$$

$2 + 2 * 2 = 6$ $\qquad$ $2 + 2 * 2 = 8$



Correct result: $2 + 2 * 2 = 6$