

Government College of Engineering, Jalgaon

(An Autonomous Institute of Govt. of Maharashtra)

Name of Student:.....

PRN:.....

Course Teacher: Sharayu Bonde

Date of Performance:

Date of completion:

Experiment No. B: 04

Title: Write a C program to check whether a string belongs to grammar or not.

Theory:

A context-free grammar (CFG) is a set of recursive rewriting rules (or productions) used to generate patterns of strings.

A CFG consists of the following components:

- a set of *terminal symbols*, which are the characters of the alphabet that appear in the strings generated by the grammar.
- a set of *nonterminal symbols*, which are placeholders for patterns of terminal symbols that can be generated by the nonterminal symbols.
- a set of *productions*, which are rules for replacing (or rewriting) nonterminal symbols (on the left side of the production) in a string with other nonterminal or terminal symbols (on the right side of the production).
- a *start symbol*, which is a special nonterminal symbol that appears in the initial string generated by the grammar.

Definition – A context-free grammar (CFG) consisting of a finite set of grammar rules is a quadruple (**N**, **T**, **P**, **S**) where

- **N** is a set of non-terminal symbols.
- **T** is a set of terminals where $N \cap T = \text{NULL}$.
- **P** is a set of rules, $P: N \rightarrow (N \cup T)^*$, i.e., the left-hand side of the production rule **P** does have any right context or left context.
- **S** is the start symbol.

Example

- The grammar (**{A}**, **{a, b, c}**, **P**, **A**), $P: A \rightarrow aA, A \rightarrow abc$.
- The grammar (**{S, a, b}**, **{a, b}**, **P**, **S**), $P: S \rightarrow aSa, S \rightarrow bSb, S \rightarrow \epsilon$
- The grammar (**{S, F}**, **{0, 1}**, **P**, **S**), $P: S \rightarrow 00S \mid 11F, F \rightarrow 00F \mid \epsilon$

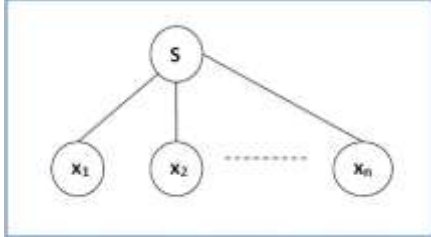
Generation of Derivation Tree

A derivation tree or parse tree is an ordered rooted tree that graphically represents the semantic information a string derived from a context-free grammar.

Representation Technique

- **Root vertex** – Must be labeled by the start symbol.
- **Vertex** – Labeled by a non-terminal symbol.
- **Leaves** – Labeled by a terminal symbol or ϵ .

If $S \rightarrow x_1x_2 \dots x_n$ is a production rule in a CFG, then the parse tree / derivation tree will be as follows –



There are two different approaches to draw a derivation tree –

Top-down Approach –

- Starts with the starting symbol **S**
- Goes down to tree leaves using productions

Bottom-up Approach –

- Starts from tree leaves
- Proceeds upward to the root which is the starting symbol **S**

Derivation or Yield of a Tree

The derivation or the yield of a parse tree is the final string obtained by concatenating the labels of the leaves of the tree from left to right, ignoring the Nulls. However, if all the leaves are Null, derivation is Null.

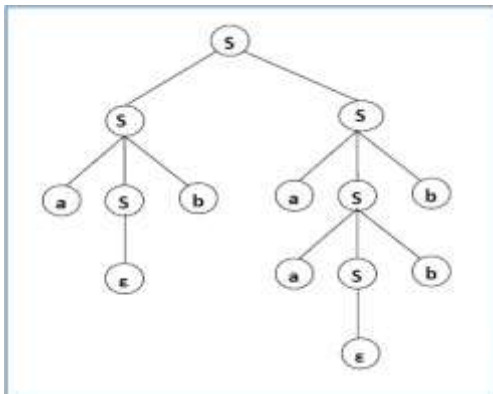
Example

Let a CFG $\{N, T, P, S\}$ be

$N = \{S\}$, $T = \{a, b\}$, Starting symbol = S , $P = S \rightarrow SS \mid aSb \mid \epsilon$

One derivation from the above CFG is “abaabb”

$S \rightarrow SS \rightarrow aSbS \rightarrow abS \rightarrow abaSb \rightarrow abaaSbb \rightarrow abaabb$



Sharayu Nivrutti Bonde
Course Teacher