Parul® University
Vadodara, Gujarat

NAAC GRADE A++

Information and Communication Technology

Introduction to syntax analysis

Study Guide

Asst. Prof. Vaibhavi Parikh
CSE, PIT
Parul University

2.2 Use of context-free grammars (CFG) in the specification of the syntax of programming languages

1. Context free grammar is a formal grammar which is used to generate all possible strings in a given formal language.
2. Context free grammar G can be defined by four tuples as:
3. G= (V, T, P, S)

   Where,
4. G describes the grammar
5. T describes a finite set of terminal symbols.
6. V describes a finite set of non-terminal symbols
7. P describes a set of production rules
8. S is the start symbol.
9. In CFG, the start symbol is used to derive the string. You can derive the string by repeatedly replacing a non-terminal by the right hand side of the production, until all non-terminal have been replaced by terminal symbols.

Production rules:

1. S → aSa
2. S → bSb
3. S → c

Now check that abbcbba string can be derived from the given CFG.

1. S ⇒ aSa
2. S ⇒ abSba
3. S ⇒ abbSbba
4. S ⇒ abbcbba

By applying the production S → aSa, S → bSb recursively and finally applying the production S → c, we get the string abbcbba.

Derivation

Derivation is a sequence of production rules. It is used to get the input string through these production rules. During parsing we have to take two decisions. These are as follows:

We have to decide the non-terminal which is to be replaced.

We have to decide the production rule by which the non-terminal will be replaced.

We have two options to decide which non-terminal to be replaced with production rule.

Left-most Derivation

In the left most derivation, the input is scanned and replaced with the production rule from left to right. So in left most derivatives we read the input string from left to right.

Example:

Production rules:

1. S = S + S
2. S = S - S
3. S = a | b |c

Input:

```
a - b + c
```

The left-most derivation is:

1. S = S + S
2. S = S - S + S
3. S = a - S + S
4. S = a - b + S
5. S = a - b + c

Right-most Derivation

In the right most derivation, the input is scanned and replaced with the production rule from right to left. So in right most derivatives we read the input string from right to left.

Example:

S = S + S

S = S - S

S = a | b |c

Input:

```
a - b + c
```

The right-most derivation is:

S = S - S

S = S - S + S

S = S - S + c

S = S - b + c

S = a - b + c

3. Summary of Key Concepts

- The role of a parser in a compiler is to check the source code's syntax, ensuring it follows the rules of the programming language, and to build a structured representation of the code, such as a parse Tree.
- It does this by taking a stream of tokens from the lexical analyzer, verifying the sequence against the language's grammar, and producing an output that can be used for further compilation stages.

  ss

Next Steps

- Explore other basic functions of CFG Context Free Grammar and Language.

References:

1. **Book Reference**

   Aho, A. V., Lam, M. S., Sethi, R., & Ullman, J. D. (2006). *Compilers: Principles, Techniques, and Tools* (2nd ed.). Pearson Education.

2. **Journal Article**

   Muchnick, S. S., & Hecht, M. S. (2018). Advances in compiler optimization: Modern approaches language processing. *Journal of Computer Science and Engineering*, 12(4), 233–245

3. **Website Reference**

   GeeksforGeeks. (2024). *Structure of a compiler*. Retrieved November 7, 2025, from https://www.geeksforgeeks.org/structure-of-compiler/

4. **Conference Presentation**

   Sharma, R., & Patel, D. (2022). *Applications of compiler technology in modern software development*. Paper presented at the International Conference on Advanced Computing and Communication Systems (ICACCS), Chennai, India.

5. **Report**

IEEE Computer Society. (2021). *Trends in programming language translation and compiler design*.

6. **Sources**

TutorialsPoint. (2024). *Lexical Analysis in Compiler Design.*

.

Parul® University
Vadodara, Gujarat
NAAC GRADE A++

https://paruluniversity.ac.in/