Context-Free Grammars

Raju Hazari

Department of Computer Science and Engineering
National Institute of Technology Calicut

November 4, 2023

Context-Free Grammars

- The topic of context-free languages is perhaps the most important aspect of formal language theory as it applies to programming languages.
- Actual programming languages have many features that can be described elegantly by means of context-free languages.
- What formal language theory tells us about context-free languages has important applications in the design of programming languages as well as in the construction of efficient compilers.

Context-Free Grammars

- The productions in regular grammar are restricted in two ways:
  - The left side must be a single non-terminal.
  - The right side has a special form
    - ★ left linear or right linear
    - ★ at most one non-terminal appears on the right side of any production
- To create grammars that are more powerful, we must relax some of these restrictions.
- By retaining the restriction on the left side, by permitting anything on the right side, we get context-free grammars.

Context-Free Grammars

- **Definition : Context-Free Grammar:**

  A grammar $G = (N, T, S, P)$ is said to be context-free if all productions in P have the form

  $$A \to x$$

  where $A \in N$ and $x \in (N \cup T)^*$

Context-Free Grammars

- **Definition : Context-Free Grammar:**

  A grammar $G = (N, T, S, P)$ is said to be context-free if all productions in P have the form

  $$A \to x$$

  where $A \in N$ and $x \in (N \cup T)^*$

- **Definition : Context-Free Language:**

  A Language L is said to be context-free if and only if there is a context-free grammar G such that $L = L(G)$.

Context-Free Grammars

- **Definition : Context-Free Grammar:**

  A grammar $G = (N, T, S, P)$ is said to be context-free if all productions in P have the form

  $$A \to x$$

  where $A \in N$ and $x \in (N \cup T)^*$

- **Definition : Context-Free Language:**

  A Language L is said to be context-free if and only if there is a context-free grammar G such that $L = L(G)$.

- Every regular grammar is context-free, so a regular language is also a context-free one.

Context-Free Grammars

- **Example 1:** The grammar G = ($\{S\}$, {a, b}, S, P), with P given by

$$1.\ S \to aSb,$$
$$2.\ S \to ab$$

is context-free.

Context-Free Grammars

- **Example 1:** The grammar G = ({S}, {a, b}, S, P), with P given by

$$1.\ S \rightarrow aSb,$$
$$2.\ S \rightarrow ab$$

is context-free.

  - Some of the derivations of this grammar is:
  - $S \overset{2}{\Rightarrow} ab$ $\quad ab \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{2}{\Rightarrow} aabb$ or $S \overset{*}{\Rightarrow} a^2b^2$ $\quad a^2b^2 \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{1}{\Rightarrow} aaSbb \overset{2}{\Rightarrow} aaabbb$ or $S \overset{*}{\Rightarrow} a^3b^3$ $\quad a^3b^3 \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{1}{\Rightarrow} aaSbb \overset{1}{\Rightarrow} aaaSbbb \overset{2}{\Rightarrow} aaaabbbb$ or $S \overset{*}{\Rightarrow} a^4b^4$ $\quad a^4b^4 \in L(G)$

Context-Free Grammars

- **Example 1:** The grammar G = ({S}, {a, b}, S, P), with P given by

$$1.\ S \to aSb,$$
$$2.\ S \to ab$$

is context-free.

  - Some of the derivations of this grammar is:
  - $S \overset{2}{\Rightarrow} ab$ $\quad ab \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{2}{\Rightarrow} aabb$ or $S \overset{*}{\Rightarrow} a^2b^2$ $\quad a^2b^2 \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{1}{\Rightarrow} aaSbb \overset{2}{\Rightarrow} aaabbb$ or $S \overset{*}{\Rightarrow} a^3b^3$ $\quad a^3b^3 \in L(G)$
  - $S \overset{1}{\Rightarrow} aSb \overset{1}{\Rightarrow} aaSbb \overset{1}{\Rightarrow} aaaSbbb \overset{2}{\Rightarrow} aaaabbbb$ or $S \overset{*}{\Rightarrow} a^4b^4$ $\quad a^4b^4 \in L(G)$

Thus, G can derive only strings of the form a^nb^n.

So, $L(G) = \{a^nb^n : n \geq 1\}$ and the language is context free.

Context-Free Grammars

- **Example 2:** The grammar G = ($\{S\}$, {a, b}, S, P), with P given by

$$1.\ S \to aSa,$$
$$2.\ S \to bSb,$$
$$3.\ S \to \epsilon$$

is context-free.

Context-Free Grammars

- **Example 2:** The grammar G = ($\{S\}$, {a, b}, S, P), with P given by

$$\begin{aligned} &1.\ S \to aSa,\\ &2.\ S \to bSb,\\ &3.\ S \to \epsilon \end{aligned}$$

is context-free.

  - Some of the derivations of this grammar is:
  - $S \overset{1}{\Rightarrow} aSa \overset{3}{\Rightarrow} aa$ $\quad aa \in L(G)$
  - $S \overset{2}{\Rightarrow} bSb \overset{3}{\Rightarrow} bb$ $\quad bb \in L(G)$
  - $S \overset{1}{\Rightarrow} aSa \overset{2}{\Rightarrow} abSba \overset{3}{\Rightarrow} abba$ or $S \overset{*}{\Rightarrow} abba$ $\quad abba \in L(G)$
  - $S \overset{2}{\Rightarrow} bSb \overset{1}{\Rightarrow} baSab \overset{3}{\Rightarrow} baab$ or $S \overset{*}{\Rightarrow} baab$ $\quad baab \in L(G)$
  - $S \overset{1}{\Rightarrow} aSa \overset{1}{\Rightarrow} aaSaa \overset{2}{\Rightarrow} aabSbaa \overset{3}{\Rightarrow} aabbaa$ or $S \overset{*}{\Rightarrow} aabbaa$ $\quad aabbaa \in L(G)$

Context-Free Grammars

- **Example 2:** The grammar G = $(\{S\}, \{\mathrm{a}, \mathrm{b}\}, S, \mathrm{P})$, with P given by

$$1.\ S \rightarrow aSa,$$
$$2.\ S \rightarrow bSb,$$
$$3.\ S \rightarrow \epsilon$$

  is context-free.
  - Some of the derivations of this grammar is:
  - $S \stackrel{1}{\Rightarrow} aSa \stackrel{3}{\Rightarrow} aa$ $aa \in L(G)$
  - $S \stackrel{2}{\Rightarrow} bSb \stackrel{3}{\Rightarrow} bb$ $bb \in L(G)$
  - $S \stackrel{1}{\Rightarrow} aSa \stackrel{2}{\Rightarrow} abSba \stackrel{3}{\Rightarrow} abba$ or $S \stackrel{*}{\Rightarrow} abba$ $abba \in L(G)$
  - $S \stackrel{2}{\Rightarrow} bSb \stackrel{1}{\Rightarrow} baSab \stackrel{3}{\Rightarrow} baab$ or $S \stackrel{*}{\Rightarrow} baab$ $baab \in L(G)$
  - $S \stackrel{1}{\Rightarrow} aSa \stackrel{1}{\Rightarrow} aaSaa \stackrel{2}{\Rightarrow} aabSbaa \stackrel{3}{\Rightarrow} aabbaa$ or $S \stackrel{*}{\Rightarrow} aabbaa$ $aabbaa \in L(G)$

  From these derivation it is clear that $L(G) = \{ww^R : w \in a, b^*\}$ and the language is context free.

Context-Free Grammars

- **Example 2:** The grammar G = ($\{S\}$, {a, b}, S, P), with P given by

$$1.\ S \rightarrow aSa,$$
$$2.\ S \rightarrow bSb,$$
$$3.\ S \rightarrow \epsilon$$

is context-free.

  - Some of the derivations of this grammar is:
  - $S \overset{1}{\Rightarrow} aSa \overset{3}{\Rightarrow} aa$ $\quad aa \in L(G)$
  - $S \overset{2}{\Rightarrow} bSb \overset{3}{\Rightarrow} bb$ $\quad bb \in L(G)$
  - $S \overset{1}{\Rightarrow} aSa \overset{2}{\Rightarrow} abSba \overset{3}{\Rightarrow} abba$ or $S \overset{*}{\Rightarrow} abba$ $\quad abba \in L(G)$
  - $S \overset{2}{\Rightarrow} bSb \overset{1}{\Rightarrow} baSab \overset{3}{\Rightarrow} baab$ or $S \overset{*}{\Rightarrow} baab$ $\quad baab \in L(G)$
  - $S \overset{1}{\Rightarrow} aSa \overset{1}{\Rightarrow} aaSaa \overset{2}{\Rightarrow} aabSbaa \overset{3}{\Rightarrow} aabbaa$ or $S \overset{*}{\Rightarrow} aabbaa$ $\quad aabbaa \in L(G)$

  From these derivation it is clear that $L(G) = \{ww^R : w \in a, b^*\}$ and the language is context free.

- From example 1 and example 2, we can say that the family of regular language is a **proper subset** of the family of the context-free language.

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

is a context-free grammar. The corresponding language is:

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

is a context-free grammar. The corresponding language is:
$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

Context-Free Grammars

- **Example 3:** The grammar G with productions
$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$
is a context-free grammar. The corresponding language is:
$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions
$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$
is a context-free grammar. The corresponding language is:

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

is a context-free grammar. The corresponding language is:
$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions

$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$

is a context-free grammar. The corresponding language is:
$$L(G) = \{a^n b^m : n \neq m\}$$

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions

$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{a^n b^m : n \neq m\}$$

- Both the grammar generate context-free language, but what is the difference between these two grammar ?

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions

$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{a^n b^m : n \neq m\}$$

- Both the grammar generate context-free language, but what is the difference between these two grammar ?
  - First one is linear, but second one is not linear.

Context-Free Grammars

- **Example 3:** The grammar G with productions

$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions

$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$

  is a context-free grammar. The corresponding language is:

$$L(G) = \{a^n b^m : n \neq m\}$$

- Both the grammar generate context-free language, but what is the difference between these two grammar ?
  - First one is linear, but second one is not linear. Why?

Context-Free Grammars

- **Example 3:** The grammar G with productions
$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$
is a context-free grammar. The corresponding language is:
$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$
- **Example 4:** The grammar G with productions
$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$
is a context-free grammar. The corresponding language is:
$$L(G) = \{a^n b^m : n \neq m\}$$
- Both the grammar generate context-free language, but what is the difference between these two grammar ?
  - First one is linear, but second one is not linear. Why?
    - ★ A **linear grammar** is a grammar in which at most one non-terminal can occur on the right side of any production.

Context-Free Grammars

- **Example 3:** The grammar G with productions
$$S \to abB, \qquad B \to bbAa,$$
$$A \to aaBb, \qquad A \to \epsilon$$
is a context-free grammar. The corresponding language is:
$$L(G) = \{ab(bbaa)^n bba(ba)^n : n \geq 0\}$$

- **Example 4:** The grammar G with productions
$$S \to AS_1|S_1B, \qquad A \to aA|a,$$
$$S_1 \to aS_1b|\epsilon, \qquad B \to bB|b$$
is a context-free grammar. The corresponding language is:
$$L(G) = \{a^n b^m : n \neq m\}$$

- Both the grammar generate context-free language, but what is the difference between these two grammar ?
  - First one is linear, but second one is not linear. Why?
    - ⋆ A **linear grammar** is a grammar in which at most one non-terminal can occur on the right side of any production.
- Regular and linear grammars are context-free, but a context-free grammar is not necessarily linear.

Context-Free Grammars

- **Example 5:** Consider the grammar G with production

$$S \to aSb|SS|\epsilon$$

  - This is another grammar that is context-free, but not linear. Some strings in $L(G)$ are $abaabb$, $aababb$, and $ababab$. So, $L(G) = \{w \in \{a, b\}^* : n_a(w) = n_b(w)$ and $n_a(v) \geq n_b(v)$,where v is any prefix of w$\}$.

- We can see the connection with programming languages clearly if we replace a and b with left and right parenthesis, respectively.

- The language $L(G)$ includes such strings as $(())$ and $()()()$ and is in fact the set of all properly nested parenthesis structures for the common programming languages.

- So, the language generated is $L(G)$ consists of well formed strings of parenthesis.
  - The language of well formed strings of parenthesis is called the **Dyck set**.

Leftmost and Rightmost Derivations

- In a grammar that is not linear, a derivation may involve sentential forms with more than one variable (or non-terminal).
- In such cases, we have a choice in the order in which variables are replaced.

Leftmost and Rightmost Derivations

- In a grammar that is not linear, a derivation may involve sentential forms with more than one variable (or non-terminal).
- In such cases, we have a choice in the order in which variables are replaced.
- **Definition :** Leftmost and Rightmost derivations :
  - A derivation is said to be **leftmost** if in each step the leftmost variable in the sentential form is replaced.
  - A derivation is said to be **rightmost** if in each step rightmost variable in the sentential form is replaced.

Leftmost and Rightmost Derivations

- **Example 6:** Consider the grammar G with productions

$$1.\ S \rightarrow aAB,$$
$$2.\ A \rightarrow bBb,$$
$$3.\ B \rightarrow A|\epsilon$$

  - Then,

$$S \overset{1}{\Rightarrow} aAB \overset{2}{\Rightarrow} abBbB \overset{3}{\Rightarrow} abAbB \overset{2}{\Rightarrow} abbBbbB \overset{3}{\Rightarrow} abbbbB \overset{3}{\Rightarrow} abbbb$$

is a **leftmost derivation** of the string $abbbb$.

Leftmost and Rightmost Derivations

- **Example 6:** Consider the grammar G with productions

$$1.\ S \to aAB,$$

$$2.\ A \to bBb,$$

$$3.\ B \to A|\epsilon$$

  - Then,

    $S \overset{1}{\Rightarrow} aAB \overset{2}{\Rightarrow} abBbB \overset{3}{\Rightarrow} abAbB \overset{2}{\Rightarrow} abbBbbB \overset{3}{\Rightarrow} abbbbB \overset{3}{\Rightarrow} abbbb$

    is a **leftmost derivation** of the string $abbbb$.

  - A **rightmost derivation** of the same string is

    $S \overset{1}{\Rightarrow} aAB \overset{3}{\Rightarrow} aA \overset{2}{\Rightarrow} abBb \overset{3}{\Rightarrow} abAb \overset{2}{\Rightarrow} abbBbb \overset{3}{\Rightarrow} abbbb$

Derivation Tree or Parse Tree

- A second way of showing derivation, independent of the order in which productions are used, is by a **derivation** or **parse tree**.
- A derivation tree is an ordered tree in which nodes are labeled with the left side of productions and in which the children of a node represent its corresponding right sides.

Derivation Tree or Parse Tree

- **Definition :** Derivation Tree

  Let $G = (N, T, P, S)$ be a context-free grammar. An ordered tree is a derivation tree for G if and only if it has the following properties.

  1. The root is labeled S.
  2. Every leaf has a label from $T \cup \{\epsilon\}$
  3. Every interior vertex (a vertex that is not a leaf) has a label from N.
  4. If a vertex has label $A \in N$, and its children are labeled (from left to right) $a_1, a_2, \cdots, a_n$, then P must contain a production of the form
  $$A \to a_1 a_2 \cdots a_n.$$
  5. A leaf labeled ϵ has no sibling, that is, a vertex with a child labeled ϵ can have no other children.

Derivation Tree or Parse Tree

- **Definition :** Derivation Tree

  Let $G = (N, T, P, S)$ be a context-free grammar. An ordered tree is a derivation tree for G if and only if it has the following properties.

  1. The root is labeled S.
  2. Every leaf has a label from $T \cup \{\epsilon\}$
  3. Every interior vertex (a vertex that is not a leaf) has a label from N.
  4. If a vertex has label $A \in N$, and its children are labeled (from left to right) $a_1, a_2, \cdots, a_n$, then P must contain a production of the form
  $$A \rightarrow a_1 a_2 \cdots a_n.$$
  5. A leaf labeled ϵ has no sibling, that is, a vertex with a child labeled ϵ can have no other children.

- A tree that has properties 3, 4, and 5, but in which 1 does not necessarily hold and in which property 2 is replaced by

  2a. Every leaf has a label from $N \cup T \cup \{\epsilon\}$,

  is said to be a **partial derivation tree**.

Derivation Tree or Parse Tree

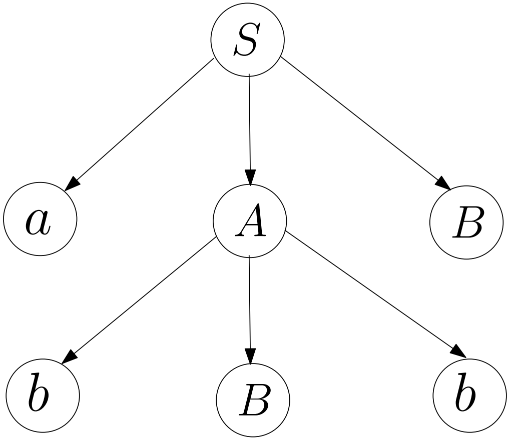
- **Example 7:** Consider the grammar G with productions

$$S \to aAB,$$

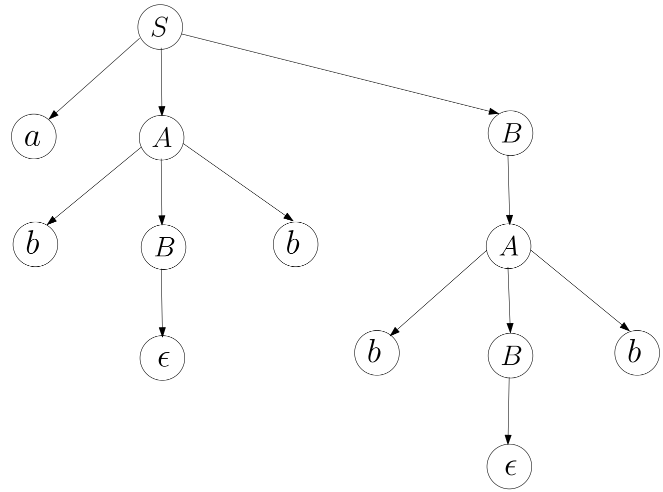
$$A \to bBb,$$

$$B \to A|\epsilon$$

The partial derivation tree for G



The derivation tree for G



- The string $abBbB$, which is yield of the first tree, is a sentential form of G. The yield of the second tree, $abbbb$, is a sentence of $L(G)$.
  - The string of symbols obtained by reading the leaves of the tree from left to right, omitting any ϵ's encountered, is said to be the **yield** of the tree.