Practice Set 1: CFG

3.1.2. Consider the grammar (V, Σ, R, S), where V, Σ, and R are defined as follows:

$$\begin{aligned} V &= \{a, b, S, A\}, \\ \Sigma &= \{a, b\}, \\ R &= \{S \to aAa, \\ &\quad\; S \to bAb, \\ &\quad\; S \to e, \\ &\quad\; A \to SS\}. \end{aligned}$$

Give a derivation of the string *baabbb* in G. (Notice that, unlike all other context-free languages we have seen so far, this one is very difficult to describe in English.)

3.1.1. Consider the grammar $G = (V, \Sigma, R, S)$, where

$$\begin{aligned} V &= \{a, b, S, A\}, \\ \Sigma &= \{a, b\}, \\ R &= \{S \to AA, \\ &\qquad A \to AAA,. \\ &\qquad A \to a, \\ &\qquad A \to bA, \\ &\qquad A \to Ab\}. \end{aligned}$$

(a) Which strings of $L(G)$ can be produced by derivations of four or fewer steps?
(b) Give at least four distinct derivations for the string $babbab$.
(c) For any $m, n, p > 0$, describe a derivation in G of the string $b^m ab^n ab^p$.

3.1.3. Construct context-free grammars that generate each of these languages.

(a) $\{wcw^R : w \in \{a, b\}^*\}$

(b) $\{ww^R : w \in \{a, b\}^*\}$

(c) $\{w \in \{a, b\}^* : w = w^R\}$

3.1.7. Let $G = (V, \Sigma, R, S)$, where $V = \{a, b, S\}$, $\Sigma = \{a, b\}$, and $R = \{S \rightarrow aSb, S \rightarrow aSa, S \rightarrow bSa, S \rightarrow bSb, S \rightarrow e\}$. Show that $L(G)$ is regular.

3.1.8. A program in a real programming language, such as C or Pascal, consists of statements, where each statement is one of several types:

(1) assignment statement, of the form id $:= E$, where E is any arithmetic expression (generated by the grammar of Example 3.1.3).

(2) conditional statement, of the form, say, if $E < E$ then statement, or a while statement of the form while $E < E$ do statement.

(3) goto statement; furthermore, each statement could be preceded by a label.

(4) compound statement, that is, many statements preceded by a begin, followed by an end, and separated by a ";".

Give a context-free grammar that generates all possible statements in the simplified programming language described above.

3.2.2. Show that the context-free grammar given in Example 3.1.4, which generates all strings of balanced parentheses is ambiguous. Give an equivalent unambiguous grammar.

Example 3.1.4: The following grammar generates all strings of properly balanced left and right parentheses: every left parenthesis can be paired with a unique subsequent right parenthesis, and every right parenthesis can be paired with a unique preceding left parenthesis. Moreover, the string between any such pair has the same property. We let $G = (V, \Sigma, R, S)$, where

$$\begin{aligned}
V &= \{S, (,)\},\\
\Sigma &= \{(,)\},\\
R &= \{S \to e,\\
&\qquad S \to SS,\\
&\qquad S \to (S)\}.
\end{aligned}$$

3.2.3. Consider the grammar of Example 3.1.3. Give two derivations of the string $\mathsf{id} * \mathsf{id} + \mathsf{id}$, one which is leftmost and one which is not leftmost.

3.2.4. Draw parse trees for each of the following.

(a) The grammar of Example 3.1.2 and the string "big Jim ate green cheese."

(b) The grammar of Example 3.1.3 and the strings $\mathsf{id} + (\mathsf{id} + \mathsf{id}) * \mathsf{id}$ and $(\mathsf{id} * \mathsf{id} + \mathsf{id} * \mathsf{id})$.