Chapter 3

Derivations beginning with \( (AdditiveExpression) \) produce correctly formed expressions with additive operators, multiplicative operators, and parentheses. For example,

\[
(AdditiveExpression) \Rightarrow (AdditiveExpression) \pm (MultiplicativeExpression)
\]

\[
\Rightarrow (MultiplicativeExpression) \pm (MultiplicativeExpression)
\]

\[
\Rightarrow (UnaryExpression) \pm (MultiplicativeExpression)
\]

\[
\Rightarrow (Identifier) \pm (MultiplicativeExpression)
\]

\[
\Rightarrow (Identifier) +
\]

\[
(MultiplicativeExpression) \ast (MultiplicativeExpression)
\]

begins such a derivation. Derivations from \( (UnaryExpression) \) can produce literals, variables, or \( (Expression) \) to obtain nested parentheses.

The rules that define identifiers, literals, and expressions show how the design of a large language is decomposed into creating rules for frequently recurring subsets of the language. The resulting variables \( (Identifier) \), \( (Literal) \), and \( (Expression) \) become the building blocks for higher-level rules.

The start symbol of the grammar is \( (CompilationUnit) \) and the derivation of a Java program begins with the rule

\[
(CompilationUnit) \rightarrow (PackageDeclaration)_{opt} (ImportDeclarations)_{opt}
\]

\[
(TypesDeclarations)_{opt}.
\]

A string of terminal symbols derivable from this rule is a syntactically correct Java program.

Exercises

1. Let \( G \) be the grammar

\[
S \rightarrow abSc \mid A
\]

\[
A \rightarrow cAd \mid cd.
\]

a) Give a derivation of \( ababcdedc. \)

b) Build the derivation tree for the derivation in part (a).

2. Let \( G \) be the grammar

\[
S \rightarrow ASB \mid \lambda
\]

\[
A \rightarrow abAb \mid \lambda
\]

\[
B \rightarrow bBa \mid ba.
\]

a) Give a leftmost derivation of \( aabba. \)

b) Give a rightmost derivation of \( ababbababaa. \).
c) Build the derivation tree for the derivations in parts (a) and (b).

d) Use set notation to define $L(G)$.

3. Let $G$ be the grammar

\[
S \rightarrow SAB \mid \lambda \\
A \rightarrow aA \mid a \\
B \rightarrow bB \mid \lambda.
\]

a) Give a leftmost derivation of $abbaab$.
b) Give two leftmost derivations of $aa$.
c) Build the derivation tree for the derivations in part (b).
d) Give a regular expression for $L(G)$.

4. Let $DT$ be the derivation tree

\[
\begin{array}{c}
S \\
A \\
& \quad A \\
& \quad \quad a \\
& \quad B \\
& \quad \quad B \\
& \quad \quad \quad a \\
& \quad \quad \quad \quad b
\end{array}
\]

a) Give a leftmost derivation that generates the tree $DT$.
b) Give a rightmost derivation that generates the tree $DT$.
c) How many different derivations are there that generate $DT$?

5. Give the leftmost and rightmost derivations corresponding to each of the derivation trees given in Figure 3.3.

6. For each of the following context-free grammars, use set notation to define the language generated by the grammar.

\[
\begin{aligned}
a) S & \rightarrow aSB \mid \lambda \\
B & \rightarrow bB \mid b
\end{aligned}
\]

\[
\begin{aligned}
b) S & \rightarrow aSbb \mid A \\
A & \rightarrow cA \mid c \\
B & \rightarrow bb \mid b
\end{aligned}
\]

\[
\begin{aligned}
c) S & \rightarrow abSde \mid A \\
A & \rightarrow cdAb \mid \lambda
\end{aligned}
\]

7. Construct a grammar over $\{a, b, c\}$ whose language is $\{a^n b^{2n} c^m \mid n, m > 0\}$.

8. Construct a grammar over $\{a, b, c\}$ whose language is $\{a^n b^m c^{2n+m} \mid n, m > 0\}$.

9. Construct a grammar over $\{a, b, c\}$ whose language is $\{a^n b^m c^i \mid 0 \leq n + m \leq i\}$. 
10. Construct a grammar over \( \{a, b\} \) whose language is \( \{a^nb^n | 0 \leq n \leq m \leq 3n\} \).

11. Construct a grammar over \( \{a, b\} \) whose language is \( \{a^nb^i a^n | i = m + n\} \).

12. Construct a grammar over \( \{a, b\} \) whose language contains precisely the strings with the same number of a's and b's.

13. Construct a grammar over \( \{a, b\} \) whose language contains precisely the strings of odd length that have the same symbol in the first and middle positions.

14. For each of the following regular grammars, give a regular expression for the language generated by the grammar.

   a) \( S \rightarrow aA \)
   \( A \rightarrow aA | bA | b \)

   b) \( S \rightarrow aA \)
   \( A \rightarrow aA | bB \)
   \( B \rightarrow bB | \lambda \)

   c) \( S \rightarrow aS | bA \)
   \( A \rightarrow bB \)
   \( B \rightarrow aB | \lambda \)

   d) \( S \rightarrow aS | bA | \lambda \)
   \( A \rightarrow aA | bS \)

For Exercises 15 through 25, give a regular grammar that generates the described language.

15. The set of strings over \( \{a, b, c\} \) in which all the a's precede the b's, which in turn precede the c's. It is possible that there are no a's, b's, or c's.

16. The set of strings over \( \{a, b\} \) that contain the substring \( aa \) and the substring \( bb \).

17. The set of strings over \( \{a, b\} \) in which the substring \( aa \) occurs at least twice. (Hint: Beware of the substring \( aaa \).)

18. The set of strings over \( \{a, b\} \) that contain the substring \( ab \) and the substring \( ba \).

19. The set of strings over \( \{a, b\} \) in which the number of a's is divisible by three.

20. The set of strings over \( \{a, b\} \) in which every a is either immediately preceded or immediately followed by b, for example, \( baab, aba, \) and \( b \).

21. The set of strings over \( \{a, b\} \) that do not contain the substring \( aba \).

22. The set of strings over \( \{a, b\} \) in which the substring \( aa \) occurs exactly once.

23. The set of strings of odd length over \( \{a, b\} \) that contain exactly two b's.

24. The set of strings over \( \{a, b, c\} \) with an odd number of occurrences of the substring \( ab \).

25. The set of strings over \( \{a, b\} \) with an even number of a's or an odd number of b's.

26. The grammar in Figure 3.1 generates \( (b^*ab^*a^*)^+ \), the set of all strings with a positive, even number of a's. Prove this.

27. Prove that the grammar given in Example 3.2.2 generates the prescribed language.

28. Let \( G \) be the grammar

   \[ S \rightarrow aSb \mid B \]
   \[ B \rightarrow bB \mid b. \]

   Prove that \( L(G) = \{a^n b^m | 0 \leq n < m\} \).