Chapter 5 Exercises

2. <S> = $ | <W> | $ <S>
<W> = abb | a <W> bb

Write all the strings in this language that contain 7 or fewer characters:
$, abb, aabbbb, $$, $$$, $$$$, $$$$$, $$$$$$, $$$$$$$, $abb, $aabbbb, $$abb, $$$abb, $$$$abb

3. Write a recursive grammar for the language of strings of one or more letters. The first letter of each string must be uppercase, and all the other letters in the string must be lowercase.

<str> = <uppercase> | <str> <lowercase>
<uppercase> = A | B | … | Z
<lowercase> = a | b | … | z

4. Consider a language of character strings that contain only dots and dashes. All strings in this language contain at least four characters and begin with either two dots or dashes. If the first two characters are dots, the last one must be a dash; if the first two characters are dashes, the last one must be a dot. Write a recursive grammar for this language.

<D> = . . <dd> - | - - <dd> .
<dd> = . | - | <dd> <dd>

5. Consider a language of strings that contains only X's, Y's, and Z's. A string in this language must begin with an X. If a Y is present in a string, it must be the final character of this string.

a. Grammar:
<XYZ> = X | X <xz> | X Y | X <xz> Y
<xz> = X | Z | <xz> <xz>

b. Two-character strings in the language:
X, XX, XZ, XY

6. Consider a language of words, where each word is a string of dots and dashes. The following grammar describes the language:

<word> = <dot> | <dash> <word> | <word> <dot>
<dot> = .
<dash> = -

a. Three-character strings in the language:
- - ., - . ., . . .

b. The string “. . . . - - -” is not in the language because it is not possible for a dash to come after a dot. That is because <dash> comes before <word> in one of the conditions of the grammar defined above, and <word> appears before <dot> in the other instance. Dashes will always precede dots.

c. “- - - - - - .” contains more dots than dashes and is in the language. The answer is correct because one dot is required in every string of the language (i.e. it is the base case; <word> = <dot>). Then, we can keep looking at the <dash> <word> section and keep adding dashes to the string, starting with “.”, which represents <word>, and then looking at <dash> <word> and adding a dash, making “- .”, which represents <word> also, and then looking at <dash> <word> again and adding another dash, making “- - .”, and so on.

9.

a. Grammar of language L of the form A^nB^(2n):
<AB> = ABB | A <AB> BB

10. The expression in this problem is not a prefix expression because a prefix expression is defined by this grammar:

<prefix> = <identifier> | <operator> <identifier> <identifier>
<operator> = + | - | * | /
<identifier> = a | b | … | z

The expression eventually ends up as the form <operator> <identifier>, with the operator being the + sign, and this does not conform to the rules defined by the grammar above.

11. The expression in this problem is a postfix expression because it conforms to this grammar:

<postfix> = <identifier> | <identifier> <identifier> <operator>
<operator> = + | - | * | /
<identifier> = a | b | … | z

The expression eventually ends up as the form <identifier> <identifier> <operator>.

12.

<S> = <L> | <D> <S> <S>
<L> = A | B
<D> = 1 | 2

a. Three-character strings of the language:
1AA, 1AB, 1BB, 1BA, 2AA, 2AB, 2BB, 2BA

B. 11AA

13. Consider a language of the following character strings: The letter A, the letter B, the letter C followed by a string that is in the language, and the letter D followed by a string in the language. For example, these strings are in this language: A, CA, CCA, DCA, B, CB, CCB, DB, and DCCB.

a. Grammar:
<charStr> = <letters> | A | B | <letters> A | <letters> B
<letters> = C | D | <letters> <letters>

b. CAB is not in the language. A and B can’t follow a string that is in the language according to the language’s definition, and since A and B are both strings that are a part of the language, the construction CAB is disqualified from being a valid member of it because B follows A.