Chapter 1 Section 3 Exercises

1. While passwords generally have few restrictions, they are normally not totally free. Suppose that in a certain system, passwords can be of arbitrary length but must contain at least one letter, $a-z$, and one number $0-9$. Construct a grammar that generates the set of such legal passwords.

Solution. The grammar G is $G = (\{\langle password\rangle, \langle letter\rangle, \langle number\rangle, \langle rest\rangle\}, \{a, b, \cdots, z, 0, 1, \cdots, 9\}, \{\langle password\rangle\}, P)$ where P is given by

$$\begin{aligned}
\langle password\rangle &\to \langle rest\rangle \langle letter\rangle \langle rest\rangle \langle number\rangle \langle rest\rangle, \\
\langle password\rangle &\to \langle rest\rangle \langle number\rangle \langle rest\rangle \langle letter\rangle \langle rest\rangle, \\
\langle rest\rangle &\to \langle letter\rangle \langle rest\rangle \mid \langle number\rangle \langle rest\rangle \mid \lambda, \\
\langle letter\rangle &\to a|b|\cdots|z, \\
\langle number\rangle &\to 0|1|\cdots|9.
\end{aligned}$$

2. Suppose that in some programming language, numbers are restricted as follows:

(a) A number may be signed or unsigned.

(b) The value field consists of two nonempty parts, separated by a decimal point.

(c) There is an optional exponent field. If present, this field must contain the letter e, followed by a signed two-digit integer.

Solution. The grammar G is $G = (\{\langle number\rangle, \langle base\rangle, \langle rest\rangle, \langle exponent\rangle, \langle sign\rangle, \langle digit\rangle\}, \{., e, +, -, 0, 1, \cdots, 9\}, \{\langle number\rangle\}, P)$ where P is given by

$$\begin{aligned}
\langle number\rangle &\to \langle base\rangle \mid \langle base\rangle \langle exponent\rangle, \\
\langle base\rangle &\to \langle sign\rangle \langle digit\rangle \langle rest\rangle . \langle digit\rangle \langle rest\rangle, \\
\langle rest\rangle &\to \langle digit\rangle \langle rest\rangle \mid \lambda, \\
\langle exponent\rangle &\to e \langle sign\rangle \langle digit\rangle \langle digit\rangle, \\
\langle sign\rangle &\to + \mid - \mid \lambda, \\
\langle digit\rangle &\to 0|1|\cdots|9.
\end{aligned}$$

3. Give a grammar for the set of integer numbers in C.

Solution. The grammar G is $G = (\{\langle integer\rangle, \langle rest\rangle, \langle sign\rangle, \langle digit\rangle\}, \{+, -, 0, 1, \cdots, 9\}, \{\langle integer\rangle\}, P)$ where P is given by

$$
\begin{aligned}
\langle integer\rangle &\to \langle sign\rangle \langle digit\rangle \langle rest\rangle \mid \langle digit\rangle \langle rest\rangle, \\
\langle rest\rangle &\to \langle digit\rangle \langle rest\rangle \mid \lambda, \\
\langle sign\rangle &\to + \mid - \mid \lambda, \\
\langle digit\rangle &\to 0 \mid 1 \mid \cdots \mid 9.
\end{aligned}
$$

4. Design an accepter for integers in C.

Solution.

5. Give a grammar that generates all real constants in C.

Solution. The grammar G is $G = (\{\langle real\rangle, \langle decimal\rangle, \langle exponent\rangle, \langle rest\rangle, \langle sign\rangle, \langle digit\rangle\}, \{+, -, ., e, 0, 1, \cdots, 9\}, \{\langle real\rangle\}, P)$ where P is given by

$$
\begin{aligned}
\langle real\rangle &\to \langle sign\rangle \langle digit\rangle \langle rest\rangle \langle decimal\rangle \langle exponent\rangle \\
\langle decimal\rangle &\to .\, \langle digit\rangle \langle rest\rangle \mid \lambda \\
\langle exponent\rangle &\to e \, \langle sign\rangle \langle digit\rangle \langle rest\rangle \mid \lambda \\
\langle rest\rangle &\to \langle digit\rangle \langle rest\rangle \mid \lambda, \\
\langle sign\rangle &\to + \mid - \mid \lambda, \\
\langle digit\rangle &\to 0 \mid 1 \mid \cdots \mid 9.
\end{aligned}
$$