Exam 1 Practice Problems

Problem 1 Consider the context-free grammar that generates arithmetic expressions (over digits and plus and times) in infix notation.

$$\begin{array}{lcl} S & \to & E \\ E & \to & E \texttt{ + } T \\ E & \to & T \\ T & \to & T \texttt{ * digit} \\ T & \to & \texttt{digit} \end{array}$$

(1) Construct an attribute grammar that translates the expressions into prefix notation. For example, `9 + 5 * 2` translates into `+ 9 * 5 2`.

(2) Construct an attribute grammar that associates an attribute `count` to the root of the tree S such that `count` contains the number of digits in the expression. For example, S.`count` for `9 + 5 * 2 * 8` is 4.

Problem 2 Let synthesized attribute *val* give the value of the binary number generated by S in the following grammar. For example, on input `101.101`, $S.val = 5.625$.

$$\begin{array}{lcl} S & \to & L \texttt{ . } L \mid L \\ L & \to & LB \mid B \\ B & \to & \texttt{0} \mid \texttt{1} \end{array}$$

Use synthesized attributes to determine $S.val$.