

5th Coursework

1/3/2004

Deadline: 5/3/2004 - 15:30 (A39)

The syntax of programming languages is usually given as a context-free grammar. To keep things simple we consider a very simple language which we call P. The syntax of P is given by the following CFG $G = (V, \Sigma, S, P)$:

- $V = \{Prog, Stmt, Stmts, Expr, OpName, Num\}$
- $\Sigma = \{\{, \}, (,), +, *, -, \text{div}, \text{if}, \text{while}, \text{print}, =, ;, 0, 1, x, y, z\}$
- $S = Prog$
- P is given by:

$$\begin{aligned} Prog &\rightarrow \{Stmts\} \\ Stmts &\rightarrow \epsilon \mid Stmt\ Stmts \\ Stmt &\rightarrow Name = Expr; \\ &\quad \mid \text{if } (Expr)\ Stmt \\ &\quad \mid \text{while } (Expr)\ Stmt \\ &\quad \mid \text{print } Expr ; \\ &\quad \mid ; \\ &\quad \mid Prog \\ Expr &\rightarrow Name \mid Num \mid (Expr\ Op\ Expr) \\ Name &\rightarrow x \mid y \mid z \\ Op &\rightarrow + \mid * \mid - \mid \text{div} \\ Num &\rightarrow 0 \mid 1 \end{aligned}$$

Note that we have only a finite set of names and numeric constants. In a real language the syntax of those would have been specified using a regular language (i.e. a 2-level approach would have been used).

1. Decide which of the following are correct P programs and give a reason for programs you reject.
 - (a) { print x ; { print y ; } }
 - (b) { if(x) ; ; }
 - (c) { x = x + 1 ; print x ; }
 - (d) { print y }
 - (e) { while (z) { x = (x + 1) ; } print y ; }
 - (f) { { { } } }

(g) { print (0) ; }

(h) { if (x = 0) print x ; }

(i) print x ;

(j) { while (x) { } while (y) { } }

2. Give derivations and parse trees for the following P programs:

(a) { while (x) print y ; print z ; }

(b) { while (x) { print y ; print z ; } }