

PROGRAM: functions

functions:

- | function functions {printf("functions -> function functions\n");}

function: FUNCTION identifier SEMICOLON BEGIN_PARAMS declarations END_PARAMS
BEGIN_LOCALS declarations END_LOCALS BEGIN_BODY statements END_BODY

declarations:

- | declaration SEMICOLON declarations

declaration: identifiers COLON ARRAY L_SQUARE_BRACKET NUMBER
R_SQUARE_BRACKET OF INTEGER

- | identifiers COLON INTEGER

- | identifiers COLON ENUM L_PAREN identifiers R_PAREN

statements:

- | statement SEMICOLON statements statement:

- | var ASSIGN expressions

- | IF boolexpressions THEN statements ENDIF

- | IF boolexpressions THEN statements ELSE statements ENDIF {

- | WHILE boolexpressions BEGINLOOP statements ENDLOOP

- | DO BEGINLOOP statements ENDLOOP WHILE boolexpressions

- | READ vars

- | WRITE vars

- | CONTINUE

- | RETURN expressions

boolexpressions: relationandexpressions

- | relationandexpressions OR relationandexpressions

relationandexpressions: relationexpressions

- | relationexpressions AND relationandexpressions

relationexpressions: NOT relationexpress

- | relationexpress

relationexpress: expressions comps expressions

- | TRUE

- | FALSE

- | L_PAREN boolexpressions R_PAREN

comps: EQ

- | NEQ
- | LT
- | GT
- | LTE
- | GTE

expressions: expression
| expression COMMA expressions

expression: multiplicative_expression
| multiplicative_expression ADD expression
| multiplicative_expression SUB expression

multiplicative_expression: term
| term MULT multiplicative_expression
| term DIV multiplicative_expression
| term MOD multiplicative_expression

term: var
| SUB var
| NUMBER
| SUB NUMBER
| L_PAREN expression R_PAREN
| SUB L_PAREN expression R_PAREN
| IDENT L_PAREN expressions R_PAREN

vars: var
| var COMMA vars

var: identifier
| identifier L_SQUARE_BRACKET expressions R_SQUARE_BRACKET

identifiers: identifier
| identifier COMMA identifiers

identifier: IDENT