prog_start → functions

function → FUNCTION ident SEMICOLON BEGIN_PARAMS declarations END_PARAMS BEGIN_LOCALS declarations END_LOCALS BEGIN_BODY statements END_BODY

functions → function functions | (epsilon)

declarations → declaration SEMICOLON declarations | (epsilon)

declaration → identifiers COLON ARRAY L_SQUARE_BRACKET NUMBER R_SQUARE_BRACKET OF INTEGER

identifiers → ident COMMA identifiers

ident → IDENT

statements → statement SEMICOLON | statement SEMICOLON statements

statement → var ASSIGN expression | IF bool_exp THEN statements ENDIF | IF bool_exp THEN statements ELSE statements ENDIF | WHILE bool_exp BEGINLOOP statements ENDLOOP | DO BEGINLOOP statements ENDLOOP WHILE bool_exp | READ vars | WRITE vars | CONTINUE | RETURN expression

bool_exp → relation_and_exp | relation_and_exp OR bool_exp

relation_and_exp → relation_ exp | relation_exp AND relation_and_exp

relation_exp → expression comp expression | NOT expression comp expression | TRUE | NOT TRUE | FALSE | NOT FALSE | L_PAREN bool_exp R_PAREN | NOT L_PAREN bool_exp R_PAREN

comp → EQ | NEQ | LT | GT | LTE GTE

expression → multiplicative_exp | multiplicative_exp ADD expression |

multiplicative_exp SUB expression

expressions → multiple_exp | (epsilon)

multiplicative_exp → term | term MULT multiplicative_exp | term DIV

multiplicative_exp | term MOD multiplicative_exp

multiple_exp → expression | expression COMMA multiple_exp

term → var | SUB | NUMBER | SUB NUMBER | L_PAREN expression

R_PAREN | SUB L_PAREN expression R_PAREN | ident L_PAREN expressions

R_PAREN

var → ident | ident L_SQUARE_BRACKET expression R_SQUARE_BRACKET

vars → var | var COMMA vars