program ::= | function program

function ::= FUNCTION identifier SEMICOLON BEGIN\_PARAMS decline END\_PARAMS BEGIN\_LOCALS decline END\_LOCALS BEGIN\_BODY statline END\_BODY

declaration ::= identifier COMMA declaration | identifier COLON INTEGER | identifier COLON ARRAY L\_SQUARE\_BRACKET number R\_SQUARE\_BRACKET OF INTEGER

number ::= NUMBER

decline ::= | declaration SEMICOLON decline

identifier ::= IDENT

statement ::= var ASSIGN expression | IF boolexp THEN statline stathelp ENDIF| WHILE boolexp BEGINLOOP statline ENDLOOP | DO BEGINLOOP statline ENDLOOP WHILE boolexp | FOR var ASSIGN number SEMICOLON boolexp SEMICOLON var ASSIGN expression BEGINLOOP statline ENDLOOP | READ varline| WRITE varline| CONTINUE | RETURN expression

stathelp ::= | ELSE statline

varline ::= var | var COMMA varline

statline ::= | statement SEMICOLON statline

boolexp ::= relationandexpr| relationandexpr OR boolexp

relationandexpr ::= relationexpr | relationexpr AND relationandexpr

relationexpr ::= NOT relationhelper | relationhelper

relationhelper ::= expression comp expression | TRUE | FALSE | L\_PAREN boolexp R\_PAREN

comp ::= EQ | NEQ | LT | GT | LTE | GTE

expression ::= multiplicativeexp | multiplicativeexp ADD expression | multiplicativeexp SUB expression

multiplicativeexp ::= term | term MULT multiplicativeexp | term DIV multiplicativeexp | term MOD multiplicativeexp

term ::= term1 | SUB term1 | identifier L\_PAREN expresscomm R\_PAREN

term1 ::= var | number | L\_PAREN expression R\_PAREN

var ::= identifier | identifier L\_SQUARE\_BRACKET expression R\_SQUARE\_BRACKET

expresscomm ::= expression | expression COMMA expresscomm