program ::= start [decl-list] stmt-list exit

decl-list ::= decl {decl}

decl ::= type ident-list ";"

ident-list ::= identifier {"," identifier}

type ::= int | float | string

stmt-list ::= stmt {stmt}

stmt ::= assign-stmt ";" | if-stmt | while-stmt| read-stmt ";" | write-stmt ";"

assign-stmt ::= identifier "=" simple\_expr

if-stmt ::= if condition then stmt-list if-stmt-2

if-stmt-2 ::= end | else stmt-list end

condition ::= expression

while-stmt ::= do stmt-list stmt-sufix

stmt-sufix ::= while condition end

read-stmt ::= scan "(" identifier ")"

write-stmt ::= print "(" writable ")"

writable ::= simple-expr | literal

expression ::= simple-expr expression-2

expression-2 ::= relop simple-expr | L

simple-expr ::= term simple-expr-2

simple-expr-2 ::= addop term simple-expr-2 | L

term ::= factor-a term-2

term-2 ::= mulop factor-a term-2 | L

fator-a ::= factor | not factor | "-" factor

factor ::= identifier | constant | "(" expression ")"

relop ::= "==" | ">" | ">=" | "<" | "<=" | "<>"

addop ::= "+" | "-" | or

mulop ::= "\*" | "/" | and

constant ::= integer\_const | float\_const | literal

integer\_const ::= digit {digit}

float\_const ::= digit{digit} “.”digit{digit}

literal ::= " “" {caractere} "”"

identifier ::= letter {letter | digit }

letter ::= [A-za-z]

digit ::= [0-9]

caractere ::= um dos caracteres ASCII, exceto “” e quebra de linha