```
<program>                ::= program { <define> <execute> }
<define>                 ::= define { <variable-sel> } | ε
<variable-sel>           ::= <not-variable> <variable> | <variable> | <not-variable> | <variable> <not-variable> | ε
<execute>                ::= execute { <command-list> }
<not-variable>           ::= not variable <not-variable-sel>
<not-variable-sel>       ::= <type> <is> <identifier-list-value> <not-variable-sel> | ε
<variable>               ::= variable <variable-sel>
<variable-sel>           ::= <type> <is> <identifier-list> , <variable-sel> | ε
<command-list>           ::= <set> .
                           | <get> .
                           | <put> .
                           | <verify> .
                           | <loop> .
                           | <while> .
<type>                   ::= natural | real | char | boolean
<identifier-list-value>  ::= <identifier> <value> | <identifier> <value> , <identifier-list-value>
<identifier-list>        ::= <identifier> | <identifier> , <identifier-list> | <identifier> [ <digit> ] | <identifier> | <digit> ] ,
<identifier-list>
<set>                    ::= set <expression> to <identifier-list>
<get>                    ::= get { <identifier-list> }
<put>                    ::= put { <identifier-list> }
<verify>                 ::= verify <expression> <false-true-sel>
<false-true-sel>         ::= <is-true> <is-false> | <is-false> <is-true> | <is-true> | <is-false> //
<loop>                   ::= loop { <command-list> } while <expression> <is> true
<while>                  ::= while <expression> <is> true do { <command-list> }

<identifier>             ::= _ <identifier-sel-u> | <letter> <identifier-sel>
<identifier-sel-u>       ::= <letter> <identifier-sel> | <digit> <letter> <identifier-sel>
<identifier-sel>         ::= <letter> <identifier-sel> | <digit> <letter> <identifier-sel> | _ <identifier-sel-u> | ε

<is-true>                ::= <is> true { <command-list> }
<is-false>               ::= <is> false { <command-list> }
<is>                     ::= is
<value>                  ::= <natural> | <real> | <char> | <boolean>
<expression>             ::= <simple-expression> <expression-sel>
<expression-sel>         ::= <relational-op> <simple-expression> <expression-sel> | ε
<simple-expression>      ::= <term> <simple-expression-sel>
<simple-expression-sel>  ::= <additive-op> <term> <simple-expression-sel> | ε
<term>                   ::= <factor> <term-sel>
<term-sel>               ::= <multiplicative-op> <factor> <term-sel> | ε
<factor>                 ::= <literal>
                           | <identifier>
                           | <sub-expression>
                           | <unary>
<sub-expression>         ::= ( <expression> )
<unary>                  ::= + <unary-sel> | - <unary-sel> | ! <unary-sel>
<unary-sel>              ::= <expression> <unary-sel> | ε
<relational-op>          ::= < | > | == | != | <= | >=
<additive-op>            ::= + | - | |
<multiplicative-op>      ::= * | ** | / | % | %%
<literal>                ::= <natural>
                           | <real>
                           | <char>
                           | <boolean>
<natural>                ::= <digit>
<real>                   ::= <digit> . <digit>
<digit>                  ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<letter>                 ::= A | B | ... | Y | Z | a | b | ... | y | z
<char>                   ::= <char-sel>
<char-sel>               ::= <ascii> <char-sel> | ε
<ascii>                  ::= #0 | #1 | ... | #254 | #255
<boolean>                ::= true | false


Vt = { program define not variable is natural real char boolean execute set get to put loop while do true false { } [ ] , + - * / ** % %% == !
= < > <= >= & | ! ( ) . }
```

CPRNOS MACHADO