```
<atribuição> ::=
       <variável> := <expressão>
<bool-lit>::=
       true
       | false
<comando> ::=
       <atribuição>
        | <condicional>
        | <iterativo>
        | <comando-composto>
<comando-composto> ::=
       begin < lista-de-comandos > end
<condicional> ::=
       if <expressão> then <comando> ( else <comando> | <vazio> )
<corpo> ::=
        <declarações> <comando-composto>
<declaração> ::=
       <declaração-de-variável>
<declaração-de-variável> ::=
       var <lista-de-ids> : <tipo>
<declarações> ::=
       <declaração>;
        | <declarações> <declaração> ;
        | <vazio>
<digito> ::=
       0
        | 1
        | 2
        | 9
<expressão> ::=
       <expressão-simples>
        | <expressão-simples> <op-rel> <expressão-simples>
<expressão-simples> ::=
       <expressão-simples> <op-ad> <termo>
       | <termo>
<fator> ::=
       <variável>
        | <literal>
        | "(" <expressão> ")"
<float-lit>::=
       <int-lit> . <int-lit>
        | <int-lit> .
        |. <int-lit>
<id>::=
       <letra>
        | <id> <letra>
        | <id> <digito>
<int-lit> ::=
       <digito>
        | <int-lit> <digito>
<iterativo> ::=
```

```
while <expressão> do <comando>
<letra> ::=
       a
        | b
       | c
       | ...
       | z
<lista-de-comandos> ::=
       <comando>;
       | lista-de-comandos> <comando> ;
       | <vazio>
lista-de-ids> ::=
       <id>
        | lista-de-ids> , <id>
<literal> ::=
       <bool-lit>
        | <int-lit>
       | <float-lit>
<op-ad> ::=
        |-
       or
<op-mul> ::=
       1/
       | and
<op-rel> ::=
       <
        | >
        | <=
       | >=
        | =
       | <>
<outros>::=
       !
        | @
       | #
       | ...
ograma> ::=
       program <id> ; <corpo> .
<termo> ::=
       <termo> <op-mul> <fator>
       | <fator>
<tipo> ::=
        | <tipo-simples>
<tipo-simples> ::=
       integer
       | real
       | boolean
<variável> ::=
       <id>
<vazio> ::=
```