



Curso: Bacharelado em Ciência da Computação - BCC

Componente Curricular: Compiladores

Professora: Maria Aparecida Amorim Sibaldo de Carvalho

Estudante: Laysa Alves Protásio, Mariane de Melo Silva

## ***TinyPascal***

<programa> ::= <declarações-globais> <bloco> '.'

### DECLARAÇÕES

<declarações-globais> ::= { <decl-var> } { <decl-proc-func> }

<decl-var> ::= **var** <lista-id> ':' <tipo> ';'

<lista-id> ::= <ident> { ',' <ident> }

<tipo> ::= **integer** | **boolean**

<decl-proc-func> ::= <decl-procedimento> | <decl-funcao>

<decl-procedimento> ::= **procedure** <ident> [ '(' <param-list> ')' ] ';' <bloco> ';'

<decl-funcao> ::= **function** <ident> [ '(' <param-list> ')' ] ':' <tipo> ';' <bloco> ';'

<bloco> ::= **begin** <stmt-list> **end**

<param-list> ::= <param> { ',' <param> }

<param> ::= <lista-id> ':' <tipo>

### COMANDOS

<stmt-list> ::= <stmt> { ';' <stmt> }

<stmt> ::=

    <if-stmt>

    | <while-stmt>

    | <write-stmt>

    | <break-stmt>

    | <continue-stmt>

    | <return-stmt>

    | <assign-or-call>

<if-stmt> ::= **if** <expr> **then** <stmt> [ **else** <stmt> ]

<while-stmt> ::= **while** <expr> **do** <stmt>



## Curso: Bacharelado em Ciência da Computação - BCC

Componente Curricular: Compiladores

Professora: Maria Aparecida Amorim Sibaldo de Carvalho

Estudante: Laysa Alves Protásio, Mariane de Melo Silva

`<write-stmt> ::= writeln '(' <out-list> ')'`  
`<out-list> ::= <out-item> { ',' <out-item> }`  
`<out-item> ::= <ident> | <number> | <string>`

`<break-stmt> ::= break`  
`<continue-stmt> ::= continue`  
`<return-stmt> ::= return <expr>`

`<assign-or-call> ::= <ident> <rest-id>`  
`<rest-id> ::= '=' <expr>`  
`| '(' [ <expr-list> ] ')'`

### EXPRESSÕES

`<expr-list> ::= <expr> { ',' <expr> }`  
  
`<expr> ::= <simple-expr> [ <rel-op> <simple-expr> ]`  
`<rel-op> ::= = | <> | < | <= | > | >=`  
  
`<simple-expr> ::= <term> { ( + | - | or ) <term> }`  
`<term> ::= <factor> { ( * | / | and ) <factor> }`

### NÚMEROS E IDENTIFICADORES

`<ident> ::= letra { letra | dígito | '_' }`

`<number> ::= dígito { dígito }`

`<factor> ::= <ident>`  
`| <number>`  
`| '(' <expr> ')'`  
`| true | false`  
`| not <factor>`

`dígito ::= ( 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 )`  
`letra ::= ( a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z|`  
`A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z )`