

**Universidade Federal do Agreste de Pernambuco**

**Bacharelado em Ciências da Computação**

**Disciplina: Compiladores**

**Professor: Maria Aparecida Amorim Sibaldo de Carvalho**

**Alunos: Arnaldo Vitor B. da Silva**

**Vítor A. B. da Silva**

**BNF/Gramática**

**# ----- Variável inicial -----**

**<S> ::= (<variable> | <operation> | <call-function> | <call-procedure> | <function> | <procedure>)\***

**# ----- Tipos de dados -----**

**<max-letter> ::= "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"**

**<min-letter> ::= "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"**

**<number> ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"**

**<boolean> ::= "True" | "False"**

**<integer> ::= <number>(<number>)\***

**<arithmetic-operation> ::= "+" | "-" | "\*" | "/"**

**<boolean-operation> ::= "==" | "!=" | "<" | "<=" | ">" | ">="**

**# ----- Variáveis -----**

**<variable-type> ::= "integer" | "boolean"**

**<variable-label> ::= (<min-letter>)(<min-letter>)\***

**<variable-value> ::= <integer> | <boolean> | <variable-label> | <operation> | <call-function>**

<variable> ::= <variable-type><variable-label> "=" <variable-value> ";;"

#### #----- Operações -----

<operation> ::= <op1> | <op2>

<op1> ::= (<integer> | <variable-label>) <arithmetic-operation> (<integer> | <variable-label>)

<op2> ::= (<boolean> | <variable-label> | <integer>) <boolean-operation> (<boolean> | <variable-label> | <integer>)

#### #---- Procedimentos e Funções-----

<parameters> ::= (  $\epsilon$  | <variable-type><variable-label> | "boolean"<boolean> | "integer"<integer> )  
<next-parameters>

<next-parameters> ::=  $\epsilon$  | " " (<variable-type><variable-label> | "boolean"<boolean> | "integer"<integer>) <next-parameters>

<call-label> ::= <max-letter> (<max-letter>)\*

<call-function> ::= <call-label> "(" <parameters> ")" ";;"

<call-procedure> ::= <call-label> "(" <parameters> ")" ";;"

<function> ::= "function" <call-label> "(" <parameters> ")" "{" <escape> <return> "}"

<procedure> ::= "procedure" <call-label> "(" <parameters> ")" "{" <escape> "}"

#### #-----Escopo-----

<scope-auxiliar> ::= <call-function> | <call-procedure> | <variable> | <system-call>

<scope> ::= <scope-auxiliar> ";;" (  $\epsilon$  | <scope> )

<system-call> ::= <if> | <while> | <print>

<if> ::= "IF" "(" <op2> ")" "{" <scope> "}" (  $\epsilon$  | "ELSE" "{" <scope> "}" )

<while> ::= "WHILE" "(" <op2> ")" "{" <scope> <control> "}"

<control> ::= <break> | <continue> |  $\epsilon$

<break> ::= "BREAK" ";;"

<continue> ::= "CONTINUE" ";;"

<return> ::= "RETURN" (<integer> | <boolean> | <variable-label>) ";;"

<print> ::= "PRINT" "(" <parameters> ")" ";;"