CES41 – Compiladores

Análise Léxica da linguagem COMP-ITA 2019

Laboratório 2
26 de Março de 2019

Aluno: Felipe Vieira Coimbra
Professor: Fábio Carneiro Mokarzel

Resultados Obtidos

O analisador léxico foi concluído com sucesso, a entrada usada como teste e a saída correspondente podem ser vistas no arquivo **teste.in** e **teste.out** respectivamente.

A entrada utilizada para teste foi:

---

call
char
do
else
false
float
for
functions
global
if
int
local
logic
main
program
read
return
statements
true
void
while
write

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
0 1 2 3 4 5 6 7 9
'a'
1.2 1.23 1.2e3 1.2e+3 1.2e-3 1.2E3 1.2E+3 1.2E-3

```
||
&&
!
<  <=  >  >=  =  !=
+  -
*  /  %
~

<-
(
)
[
]
{
}
;
,
:

program AnaliseDeTexto {

global:
        char nomes[50,10], palavra[10];
        int ntab, nocorr[50];
    char c; logic fim;

functions:

    int Procura () {

    local:
        int i, inf, sup, med, posic, compara;
        logic achou, fimteste;
    statements:
            achou <- false; inf <- 1; sup <- ntab;
            while (!achou && sup >= inf) {
                    med <- (inf + sup) / 2;
                    compara <- 0; fimteste <- false;
                    for (i <- 0; !fimteste && compara = 0; i <- i+1) {
              if (palavra[i] < nomes[med,i])
                    compara <- ~1;
                else if (palavra[i] > nomes[med,i])
                    compara <- 1;
                if (palavra[i] = '\0' || nomes[med,i] = '\0')
                    fimteste <- true;
                    }
                    if (compara = 0)
                            achou <- true;
```

```
                else if (compara < 0)
                        sup <- med - 1;
                else inf <- med + 1;
        }
        if (achou) posic <- med;
        else posic <- ~inf;
        return posic;
}

void Inserir (int posic) {

local:
   int i, j; logic fim;
statements:
   ntab <- ntab + 1;
   for (i <- ntab; i >= posic+1; i <- i-1) {
      fim <- false;
      for (j <- 0; !fim; j <- j+1) {
         nomes[i,j] <- nomes[i-1,j];
         if (nomes[i,j] = '\0') fim <- true;
      }
   nocorr[i] <- nocorr[i-1];
   }
      fim <- false;
      for (j <- 0; !fim; j <- j+1) {
         nomes[posic,j] <- palavra[j];
         if (palavra[j] = '\0') fim <- true;
      }
   nocorr[posic] <- 1;
}

void ExibirTabela () {

local:
   int i; logic fim;
statements:
   write ("          ",
          "Palavra             ",
          "   Num. de ocorr.");
   for (i <- 1; i <= 50; i <- i+1) write ("-");
   for (i <- 1; i <= ntab; i <- i+1) {
      write ("\n          "); fim <- false;
      for (j <- 0; !fim; j <- j+1) {
         if (nomes[i,j] = '\0') fim <- true;
         else write (nomes[i,j]);
      }
      write (" | ", nocorr[i]);
   }
```

```
    }

    main {

    local:
        int i, posic;
            char c; logic fim;
    statements:
        ntab <- 0;
        write ("Nova palavra? (s/n): ");
        read (c);
        while (c = 's' || c = 'S') {
                write ("\nDigite a palavra: ");
                fim <- false;
            for (i <- 0; !fim; i <- i+1) {
                    read (palavra[i]);
                    if (palavra[i] = '\n') {
                        fim <- true;
                        palavra[i] <- '\0';
                    }
                }
            posic <- Procura ();
            if (posic > 0)
                nocorr[posic] <- nocorr[posic] + 1;
            else
                call Inserir (~posic, i);
                write ("\n\nNova palavra? (s/n): ");
                read (c);
        }
        call ExibirTabela ();
    }

}
```

E a saída gerada foi:

```
-------------------------------------------------------------------------
                  Texto |       Tipo |                  Atributo
-------------------------------------------------------------------------
                   call|       CALL |                 XXXXXXXX
                   char|       CHAR |                  XXXXXXXX
                     do|         DO |                 XXXXXXXX
                   else|       ELSE |                  XXXXXXXX
                  false|      FALSE |                  XXXXXXXX
                  float|      FLOAT |                  XXXXXXXX
                    for|        FOR |                 XXXXXXXX
              functions|  FUNCTIONS |                        XXXXXXXX
```

global	GLOBAL	XXXXXXXX
if	IF	XXXXXXXX
int	INT	XXXXXXXX
local	LOCAL	XXXXXXXX
logic	LOGIC	XXXXXXXX
main	MAIN	XXXXXXXX
program	PROGRAM	XXXXXXXX
read	READ	XXXXXXXX
return	RETURN	XXXXXXXX
statements	STATEMENTS	XXXXXXXX
true	TRUE	XXXXXXXX
void	VOID	XXXXXXXX
while	WHILE	XXXXXXXX
write	WRITE	XXXXXXXX
a	ID	a
b	ID	b
c	ID	c
d	ID	d
e	ID	e
f	ID	f
g	ID	g
h	ID	h
i	ID	i
j	ID	j
k	ID	k
l	ID	l
m	ID	m
n	ID	n
o	ID	o
p	ID	p
q	ID	q
r	ID	r
s	ID	s
t	ID	t
u	ID	u
v	ID	v
w	ID	w
x	ID	x
y	ID	y
z	ID	z
A	ID	A
B	ID	B
C	ID	C
D	ID	D
E	ID	E
F	ID	F
G	ID	G
H	ID	H

```
     I|         ID |                    I
     J|         ID |                    J
     K|         ID |                    K
     L|         ID |                    L
     M|         ID |                    M
     N|         ID |                    N
     O|         ID |                    O
     P|         ID |                    P
     Q|         ID |                    Q
     R|         ID |                    R
     S|         ID |                    S
     T|         ID |                    T
     U|         ID |                    U
     V|         ID |                    V
     W|         ID |                    W
     X|         ID |                    X
     Y|         ID |                    Y
     Z|         ID |                    Z
     0|      INTCT |                    0
     1|      INTCT |                    1
     2|      INTCT |                    2
     3|      INTCT |                    3
     4|      INTCT |                    4
     5|      INTCT |                    5
     6|      INTCT |                    6
     7|      INTCT |                    7
     9|      INTCT |                    9
   'a'|     CHARCT |                    a
   1.2|    FLOATCT |                 1.200000
  1.23|    FLOATCT |                 1.230000
 1.2e3|    FLOATCT |              1200.000000
1.2e+3|    FLOATCT |              1200.000000
1.2e-3|    FLOATCT |                 0.001200
 1.2E3|    FLOATCT |              1200.000000
1.2E+3|    FLOATCT |              1200.000000
1.2E-3|    FLOATCT |                 0.001200
    |||        OR |                XXXXXXXX
    &&|       AND |                XXXXXXXX
     !|       NOT |                XXXXXXXX
     <|     RELOP |                      LT
    <=|     RELOP |                     LEQ
     >|     RELOP |                      GT
    >=|     RELOP |                     GEQ
     =|     RELOP |                      EQ
    !=|     RELOP |                     DIF
     +|      ADOP |                    PLUS
     -|      ADOP |                   MINUS
     *|     MULTOP |                    TIME
```

/| MULTOP | DIV
%| MULTOP | REM
~| NEG | XXXXXXXX
<-| ASSIGN | XXXXXXXX
(| OPPAR | XXXXXXXX
)| CLPAR | XXXXXXXX
[| OPBRAK | XXXXXXXX
]| CLBRAK | XXXXXXXX
{| OPBRACE | XXXXXXXX
}| CLBRACE | XXXXXXXX
;| SCOLON | XXXXXXXX
,| COMMA | XXXXXXXX
:| COLON | XXXXXXXX
program| PROGRAM | XXXXXXXX
AnaliseDeTexto| ID | AnaliseDeTexto
{| OPBRACE | XXXXXXXX
global| GLOBAL | XXXXXXXX
:| COLON | XXXXXXXX
char| CHAR | XXXXXXXX
nomes| ID | nomes
[| OPBRAK | XXXXXXXX
50| INTCT | 50
,| COMMA | XXXXXXXX
10| INTCT | 10
]| CLBRAK | XXXXXXXX
,| COMMA | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX
10| INTCT | 10
]| CLBRAK | XXXXXXXX
;| SCOLON | XXXXXXXX
int| INT | XXXXXXXX
ntab| ID | ntab
,| COMMA | XXXXXXXX
nocorr| ID | nocorr
[| OPBRAK | XXXXXXXX
50| INTCT | 50
]| CLBRAK | XXXXXXXX
;| SCOLON | XXXXXXXX
char| CHAR | XXXXXXXX
c| ID | c
;| SCOLON | XXXXXXXX
logic| LOGIC | XXXXXXXX
fim| ID | fim
;| SCOLON | XXXXXXXX
functions| FUNCTIONS | XXXXXXXX
:| COLON | XXXXXXXX
int| INT | XXXXXXXX

```
Procura|        ID |                Procura
      (|     OPPAR |               XXXXXXXX
      )|     CLPAR |               XXXXXXXX
      {|   OPBRACE |                 XXXXXXXX
  local|     LOCAL |                 XXXXXXXX
      :|     COLON |                XXXXXXXX
    int|       INT |               XXXXXXXX
      i|        ID |                  i
      ,|     COMMA |                 XXXXXXXX
    inf|        ID |                inf
      ,|     COMMA |                 XXXXXXXX
    sup|        ID |                 sup
      ,|     COMMA |                 XXXXXXXX
    med|        ID |                  med
      ,|     COMMA |                 XXXXXXXX
  posic|        ID |                posic
      ,|     COMMA |                 XXXXXXXX
 compara|        ID |                 compara
      ;|    SCOLON |                 XXXXXXXX
  logic|     LOGIC |                 XXXXXXXX
  achou|        ID |                 achou
      ,|     COMMA |                 XXXXXXXX
fimteste|        ID |              fimteste
      ;|    SCOLON |                 XXXXXXXX
statements| STATEMENTS |                    XXXXXXXX
      :|     COLON |                XXXXXXXX
  achou|        ID |                 achou
     <-|    ASSIGN |                 XXXXXXXX
  false|     FALSE |                XXXXXXXX
      ;|    SCOLON |                 XXXXXXXX
    inf|        ID |                inf
     <-|    ASSIGN |                 XXXXXXXX
      1|     INTCT |                    1
      ;|    SCOLON |                 XXXXXXXX
    sup|        ID |                 sup
     <-|    ASSIGN |                 XXXXXXXX
   ntab|        ID |                 ntab
      ;|    SCOLON |                 XXXXXXXX
  while|     WHILE |                  XXXXXXXX
      (|     OPPAR |               XXXXXXXX
      !|      NOT |               XXXXXXXX
  achou|        ID |                  achou
     &&|       AND |                XXXXXXXX
    sup|        ID |                  sup
     >=|     RELOP |                      GEQ
    inf|        ID |                inf
      )|     CLPAR |               XXXXXXXX
      {|   OPBRACE |                 XXXXXXXX
```

```
    med|        ID |                med
     <-|    ASSIGN |             XXXXXXXX
     (|     OPPAR |            XXXXXXXX
    inf|        ID |               inf
     +|      ADOP |                PLUS
    sup|        ID |               sup
     )|     CLPAR |            XXXXXXXX
     /|    MULTOP |                 DIV
     2|     INTCT |                   2
     ;|    SCOLON |             XXXXXXXX
compara|        ID |            compara
     <-|    ASSIGN |             XXXXXXXX
     0|     INTCT |                   0
     ;|    SCOLON |             XXXXXXXX
fimteste|        ID |           fimteste
     <-|    ASSIGN |             XXXXXXXX
  false|     FALSE |             XXXXXXXX
     ;|    SCOLON |             XXXXXXXX
    for|       FOR |            XXXXXXXX
     (|     OPPAR |             XXXXXXXX
     i|        ID |                i
     <-|    ASSIGN |             XXXXXXXX
     0|     INTCT |                   0
     ;|    SCOLON |             XXXXXXXX
     !|       NOT |           XXXXXXXX
fimteste|        ID |           fimteste
     &&|       AND |             XXXXXXXX
compara|        ID |            compara
     =|     RELOP |                 EQ
     0|     INTCT |                   0
     ;|    SCOLON |             XXXXXXXX
     i|        ID |                i
     <-|    ASSIGN |             XXXXXXXX
     i|        ID |                i
     +|      ADOP |                PLUS
     1|     INTCT |                   1
     )|     CLPAR |            XXXXXXXX
     {|   OPBRACE |              XXXXXXXX
    if|        IF |           XXXXXXXX
     (|     OPPAR |            XXXXXXXX
palavra|        ID |            palavra
     [|    OPBRAK |             XXXXXXXX
     i|        ID |                i
     ]|    CLBRAK |             XXXXXXXX
     <|     RELOP |                 LT
  nomes|        ID |             nomes
     [|    OPBRAK |             XXXXXXXX
    med|        ID |                med
```

,| COMMA | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
)| CLPAR | XXXXXXXX
compara| ID | compara
<-| ASSIGN | XXXXXXXX
~| NEG | XXXXXXXX
1| INTCT | 1
;| SCOLON | XXXXXXXX
else| ELSE | XXXXXXXX
if| IF | XXXXXXXX
(| OPPAR | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
>| RELOP | GT
nomes| ID | nomes
[| OPBRAK | XXXXXXXX
med| ID | med
,| COMMA | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
)| CLPAR | XXXXXXXX
compara| ID | compara
<-| ASSIGN | XXXXXXXX
1| INTCT | 1
;| SCOLON | XXXXXXXX
if| IF | XXXXXXXX
(| OPPAR | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
=| RELOP | EQ
'\0'| CHARCT | \0
|||| OR | XXXXXXXX
nomes| ID | nomes
[| OPBRAK | XXXXXXXX
med| ID | med
,| COMMA | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
=| RELOP | EQ
'\0'| CHARCT | \0
)| CLPAR | XXXXXXXX
fimteste| ID | fimteste
<-| ASSIGN | XXXXXXXX

```
   true|       TRUE |              XXXXXXXX
     ;|     SCOLON |              XXXXXXXX
     }|    CLBRACE |              XXXXXXXX
     if|         IF |           XXXXXXXX
     (|      OPPAR |              XXXXXXXX
compara|         ID |              compara
     =|      RELOP |                 EQ
     0|      INTCT |                 0
     )|      CLPAR |              XXXXXXXX
  achou|         ID |              achou
    <-|     ASSIGN |              XXXXXXXX
   true|       TRUE |              XXXXXXXX
     ;|     SCOLON |              XXXXXXXX
   else|       ELSE |              XXXXXXXX
     if|         IF |           XXXXXXXX
     (|      OPPAR |              XXXXXXXX
compara|         ID |              compara
     <|      RELOP |                 LT
     0|      INTCT |                 0
     )|      CLPAR |              XXXXXXXX
    sup|         ID |              sup
    <-|     ASSIGN |              XXXXXXXX
    med|         ID |              med
     -|      ADOP |              MINUS
     1|      INTCT |                 1
     ;|     SCOLON |              XXXXXXXX
   else|       ELSE |              XXXXXXXX
    inf|         ID |              inf
    <-|     ASSIGN |              XXXXXXXX
    med|         ID |              med
     +|      ADOP |               PLUS
     1|      INTCT |                 1
     ;|     SCOLON |              XXXXXXXX
     }|    CLBRACE |              XXXXXXXX
     if|         IF |           XXXXXXXX
     (|      OPPAR |              XXXXXXXX
  achou|         ID |              achou
     )|      CLPAR |              XXXXXXXX
  posic|         ID |              posic
    <-|     ASSIGN |              XXXXXXXX
   med|         ID |              med
     ;|     SCOLON |              XXXXXXXX
   else|       ELSE |              XXXXXXXX
  posic|         ID |              posic
    <-|     ASSIGN |              XXXXXXXX
     ~|        NEG |              XXXXXXXX
    inf|         ID |              inf
     ;|     SCOLON |              XXXXXXXX
```

```
  return|     RETURN |                XXXXXXXX
   posic|         ID |                  posic
      ;|     SCOLON |                 XXXXXXXX
      }|    CLBRACE |                 XXXXXXXX
    void|       VOID |                XXXXXXXX
  Inserir|         ID |                Inserir
      (|      OPPAR |                XXXXXXXX
     int|        INT |               XXXXXXXX
   posic|         ID |                  posic
      )|      CLPAR |                XXXXXXXX
      {|    OPBRACE |                  XXXXXXXX
   local|      LOCAL |                 XXXXXXXX
      :|      COLON |                XXXXXXXX
     int|        INT |               XXXXXXXX
       i|         ID |                    i
       ,|      COMMA |                 XXXXXXXX
       j|         ID |                    j
       ;|     SCOLON |                 XXXXXXXX
   logic|      LOGIC |                 XXXXXXXX
     fim|         ID |                  fim
       ;|     SCOLON |                 XXXXXXXX
statements| STATEMENTS |                    XXXXXXXX
       :|      COLON |                XXXXXXXX
    ntab|         ID |                  ntab
      <-|     ASSIGN |                 XXXXXXXX
    ntab|         ID |                  ntab
       +|       ADOP |                   PLUS
       1|      INTCT |                      1
       ;|     SCOLON |                 XXXXXXXX
     for|        FOR |                XXXXXXXX
       (|      OPPAR |                XXXXXXXX
       i|         ID |                    i
      <-|     ASSIGN |                 XXXXXXXX
    ntab|         ID |                  ntab
       ;|     SCOLON |                 XXXXXXXX
       i|         ID |                    i
      >=|      RELOP |                    GEQ
   posic|         ID |                  posic
       +|       ADOP |                   PLUS
       1|      INTCT |                      1
       ;|     SCOLON |                 XXXXXXXX
       i|         ID |                    i
      <-|     ASSIGN |                 XXXXXXXX
       i|         ID |                    i
       -|       ADOP |                   MINUS
       1|      INTCT |                      1
       )|      CLPAR |                XXXXXXXX
       {|    OPBRACE |                  XXXXXXXX
```

```
  fim|        ID |                fim
   <-|    ASSIGN |           XXXXXXXX
false|     FALSE |           XXXXXXXX
    ;|    SCOLON |           XXXXXXXX
  for|       FOR |          XXXXXXXX
    (|     OPPAR |          XXXXXXXX
    j|        ID |                j
   <-|    ASSIGN |           XXXXXXXX
    0|     INTCT |                0
    ;|    SCOLON |           XXXXXXXX
    !|       NOT |          XXXXXXXX
  fim|        ID |                fim
    ;|    SCOLON |           XXXXXXXX
    j|        ID |                j
   <-|    ASSIGN |           XXXXXXXX
    j|        ID |                j
    +|     ADOP |              PLUS
    1|     INTCT |                1
    )|     CLPAR |          XXXXXXXX
    {|   OPBRACE |             XXXXXXXX
nomes|        ID |             nomes
    [|    OPBRAK |           XXXXXXXX
    i|        ID |                i
    ,|    COMMA |           XXXXXXXX
    j|        ID |                j
    ]|    CLBRAK |           XXXXXXXX
   <-|    ASSIGN |           XXXXXXXX
nomes|        ID |             nomes
    [|    OPBRAK |           XXXXXXXX
    i|        ID |                i
    -|     ADOP |           MINUS
    1|     INTCT |                1
    ,|    COMMA |           XXXXXXXX
    j|        ID |                j
    ]|    CLBRAK |           XXXXXXXX
    ;|    SCOLON |           XXXXXXXX
   if|       IF |          XXXXXXXX
    (|     OPPAR |          XXXXXXXX
nomes|        ID |             nomes
    [|    OPBRAK |           XXXXXXXX
    i|        ID |                i
    ,|    COMMA |           XXXXXXXX
    j|        ID |                j
    ]|    CLBRAK |           XXXXXXXX
    =|     RELOP |                EQ
 '\0'|    CHARCT |                \0
    )|     CLPAR |          XXXXXXXX
  fim|        ID |                fim
```

```
    <-|     ASSIGN |              XXXXXXXX
  true|       TRUE |              XXXXXXXX
     ;|     SCOLON |              XXXXXXXX
     }|    CLBRACE |               XXXXXXXX
nocorr|         ID |              nocorr
     [|     OPBRAK |              XXXXXXXX
     i|         ID |              i
     ]|     CLBRAK |              XXXXXXXX
    <-|     ASSIGN |              XXXXXXXX
nocorr|         ID |              nocorr
     [|     OPBRAK |              XXXXXXXX
     i|         ID |              i
     -|      ADOP |              MINUS
     1|     INTCT |                   1
     ]|     CLBRAK |              XXXXXXXX
     ;|     SCOLON |              XXXXXXXX
     }|    CLBRACE |               XXXXXXXX
   fim|         ID |              fim
    <-|     ASSIGN |              XXXXXXXX
 false|      FALSE |              XXXXXXXX
     ;|     SCOLON |              XXXXXXXX
   for|       FOR |             XXXXXXXX
     (|      OPPAR |             XXXXXXXX
     j|         ID |              j
    <-|     ASSIGN |              XXXXXXXX
     0|     INTCT |                   0
     ;|     SCOLON |              XXXXXXXX
     !|       NOT |             XXXXXXXX
   fim|         ID |              fim
     ;|     SCOLON |              XXXXXXXX
     j|         ID |              j
    <-|     ASSIGN |              XXXXXXXX
     j|         ID |              j
     +|      ADOP |                PLUS
     1|     INTCT |                   1
     )|      CLPAR |             XXXXXXXX
     {|    OPBRACE |               XXXXXXXX
 nomes|         ID |               nomes
     [|     OPBRAK |              XXXXXXXX
 posic|         ID |              posic
     ,|     COMMA |              XXXXXXXX
     j|         ID |              j
     ]|     CLBRAK |              XXXXXXXX
    <-|     ASSIGN |              XXXXXXXX
palavra|         ID |             palavra
     [|     OPBRAK |              XXXXXXXX
     j|         ID |              j
     ]|     CLBRAK |              XXXXXXXX
```

```
                ;|     SCOLON |                XXXXXXXX
               if|         IF |              XXXXXXXX
                (|      OPPAR |                XXXXXXXX
          palavra|         ID |                palavra
                [|     OPBRAK |                XXXXXXXX
                j|         ID |                j
                ]|     CLBRAK |                XXXXXXXX
                =|      RELOP |                     EQ
             '\0'|     CHARCT |                     \0
                )|      CLPAR |                XXXXXXXX
              fim|         ID |                fim
               <-|     ASSIGN |                XXXXXXXX
             true|       TRUE |                XXXXXXXX
                ;|     SCOLON |                XXXXXXXX
                }|    CLBRACE |                 XXXXXXXX
           nocorr|         ID |                nocorr
                [|     OPBRAK |                XXXXXXXX
            posic|         ID |                posic
                ]|     CLBRAK |                XXXXXXXX
               <-|     ASSIGN |                XXXXXXXX
                1|      INTCT |                     1
                ;|     SCOLON |                XXXXXXXX
                }|    CLBRACE |                 XXXXXXXX
             void|       VOID |               XXXXXXXX
     ExibirTabela|         ID |              ExibirTabela
                (|      OPPAR |                XXXXXXXX
                )|      CLPAR |                XXXXXXXX
                {|    OPBRACE |                  XXXXXXXX
            local|      LOCAL |                XXXXXXXX
                :|      COLON |               XXXXXXXX
              int|        INT |              XXXXXXXX
                i|         ID |                i
                ;|     SCOLON |                XXXXXXXX
            logic|      LOGIC |                XXXXXXXX
              fim|         ID |                fim
                ;|     SCOLON |                XXXXXXXX
       statements| STATEMENTS |                       XXXXXXXX
                :|      COLON |               XXXXXXXX
            write|      WRITE |                XXXXXXXX
                (|      OPPAR |               XXXXXXXX
        "        "|     STRING |           "          "
                ,|      COMMA |                XXXXXXXX
"Palavra            "|     STRING |      "Palavra            "
                ,|      COMMA |                XXXXXXXX
  "   Num. de ocorr."|     STRING |        "   Num. de ocorr."
                )|      CLPAR |               XXXXXXXX
                ;|     SCOLON |                XXXXXXXX
              for|        FOR |               XXXXXXXX
```

```
                (|      OPPAR |                XXXXXXXX
                i|        ID |                  i
               <-|    ASSIGN |                  XXXXXXXX
                1|     INTCT |                     1
                ;|    SCOLON |                  XXXXXXXX
                i|        ID |                  i
               <=|     RELOP |                    LEQ
               50|     INTCT |                    50
                ;|    SCOLON |                  XXXXXXXX
                i|        ID |                  i
               <-|    ASSIGN |                  XXXXXXXX
                i|        ID |                  i
                +|     ADOP |                    PLUS
                1|     INTCT |                     1
                )|     CLPAR |                XXXXXXXX
            write|     WRITE |                   XXXXXXXX
                (|     OPPAR |                XXXXXXXX
              "-"|    STRING |                     "-"
                )|     CLPAR |                XXXXXXXX
                ;|    SCOLON |                  XXXXXXXX
              for|       FOR |                XXXXXXXX
                (|     OPPAR |                XXXXXXXX
                i|        ID |                  i
               <-|    ASSIGN |                  XXXXXXXX
                1|     INTCT |                     1
                ;|    SCOLON |                  XXXXXXXX
                i|        ID |                  i
               <=|     RELOP |                    LEQ
             ntab|       ID |                  ntab
                ;|    SCOLON |                  XXXXXXXX
                i|        ID |                  i
               <-|    ASSIGN |                  XXXXXXXX
                i|        ID |                  i
                +|     ADOP |                    PLUS
                1|     INTCT |                     1
                )|     CLPAR |                XXXXXXXX
                {|   OPBRACE |                   XXXXXXXX
            write|     WRITE |                   XXXXXXXX
                (|     OPPAR |                XXXXXXXX
"\n           "|    STRING |           "\n          "
                )|     CLPAR |                XXXXXXXX
                ;|    SCOLON |                  XXXXXXXX
              fim|       ID |                  fim
               <-|    ASSIGN |                  XXXXXXXX
            false|     FALSE |                  XXXXXXXX
                ;|    SCOLON |                  XXXXXXXX
              for|       FOR |                XXXXXXXX
                (|     OPPAR |                XXXXXXXX
```

```
    j|        ID |                 j
   <-|    ASSIGN |              XXXXXXXX
    0|    INTCT |                   0
    ;|    SCOLON |              XXXXXXXX
    !|       NOT |             XXXXXXXX
  fim|        ID |                 fim
    ;|    SCOLON |              XXXXXXXX
    j|        ID |                 j
   <-|    ASSIGN |              XXXXXXXX
    j|        ID |                 j
    +|     ADOP |                PLUS
    1|    INTCT |                   1
    )|    CLPAR |             XXXXXXXX
    {|   OPBRACE |               XXXXXXXX
   if|       IF |             XXXXXXXX
    (|    OPPAR |              XXXXXXXX
nomes|       ID |                nomes
    [|    OPBRAK |              XXXXXXXX
    i|        ID |                 i
    ,|    COMMA |              XXXXXXXX
    j|        ID |                 j
    ]|    CLBRAK |              XXXXXXXX
    =|     RELOP |                  EQ
 '\0'|    CHARCT |                  \0
    )|    CLPAR |              XXXXXXXX
  fim|        ID |                 fim
   <-|    ASSIGN |              XXXXXXXX
 true|      TRUE |              XXXXXXXX
    ;|    SCOLON |              XXXXXXXX
 else|      ELSE |             XXXXXXXX
write|     WRITE |               XXXXXXXX
    (|    OPPAR |              XXXXXXXX
nomes|       ID |                nomes
    [|    OPBRAK |              XXXXXXXX
    i|        ID |                 i
    ,|    COMMA |              XXXXXXXX
    j|        ID |                 j
    ]|    CLBRAK |              XXXXXXXX
    )|    CLPAR |              XXXXXXXX
    ;|    SCOLON |              XXXXXXXX
    }|   CLBRACE |               XXXXXXXX
write|     WRITE |               XXXXXXXX
    (|    OPPAR |              XXXXXXXX
" | "|    STRING |                " | "
    ,|    COMMA |              XXXXXXXX
nocorr|       ID |               nocorr
    [|    OPBRAK |              XXXXXXXX
    i|        ID |                 i
```

```
                ]|     CLBRAK |                XXXXXXXX
                )|      CLPAR |               XXXXXXXX
                ;|     SCOLON |                XXXXXXXX
                }|    CLBRACE |                 XXXXXXXX
                }|    CLBRACE |                 XXXXXXXX
             main|       MAIN |                 XXXXXXXX
                {|    OPBRACE |                 XXXXXXXX
            local|      LOCAL |                 XXXXXXXX
                :|      COLON |               XXXXXXXX
              int|        INT |              XXXXXXXX
                i|         ID |                 i
                ,|      COMMA |                 XXXXXXXX
            posic|         ID |                 posic
                ;|     SCOLON |                 XXXXXXXX
             char|       CHAR |                 XXXXXXXX
                c|         ID |                  c
                ;|     SCOLON |                 XXXXXXXX
            logic|      LOGIC |                 XXXXXXXX
              fim|         ID |                 fim
                ;|     SCOLON |                 XXXXXXXX
       statements| STATEMENTS |                       XXXXXXXX
                :|      COLON |               XXXXXXXX
             ntab|         ID |                 ntab
               <-|     ASSIGN |                 XXXXXXXX
                0|      INTCT |                    0
                ;|     SCOLON |                 XXXXXXXX
            write|      WRITE |                  XXXXXXXX
                (|      OPPAR |               XXXXXXXX
"Nova palavra? (s/n): "|     STRING |     "Nova palavra? (s/n): "
                )|      CLPAR |               XXXXXXXX
                ;|     SCOLON |                 XXXXXXXX
             read|       READ |                 XXXXXXXX
                (|      OPPAR |               XXXXXXXX
                c|         ID |                  c
                )|      CLPAR |               XXXXXXXX
                ;|     SCOLON |                 XXXXXXXX
            while|      WHILE |                  XXXXXXXX
                (|      OPPAR |               XXXXXXXX
                c|         ID |                  c
                =|      RELOP |                     EQ
              's'|     CHARCT |                      s
               |||         OR |              XXXXXXXX
                c|         ID |                  c
                =|      RELOP |                     EQ
              'S'|     CHARCT |                      S
                )|      CLPAR |               XXXXXXXX
                {|    OPBRACE |                  XXXXXXXX
            write|      WRITE |                  XXXXXXXX
```

(| OPPAR | XXXXXXXX
"\nDigite a palavra: "| STRING | "\nDigite a palavra: "
)| CLPAR | XXXXXXXX
;| SCOLON | XXXXXXXX
fim| ID | fim
<-| ASSIGN | XXXXXXXX
false| FALSE | XXXXXXXX
;| SCOLON | XXXXXXXX
for| FOR | XXXXXXXX
(| OPPAR | XXXXXXXX
i| ID | i
<-| ASSIGN | XXXXXXXX
0| INTCT | 0
;| SCOLON | XXXXXXXX
!| NOT | XXXXXXXX
fim| ID | fim
;| SCOLON | XXXXXXXX
i| ID | i
<-| ASSIGN | XXXXXXXX
i| ID | i
+| ADOP | PLUS
1| INTCT | 1
)| CLPAR | XXXXXXXX
{| OPBRACE | XXXXXXXX
read| READ | XXXXXXXX
(| OPPAR | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
)| CLPAR | XXXXXXXX
;| SCOLON | XXXXXXXX
if| IF | XXXXXXXX
(| OPPAR | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX
i| ID | i
]| CLBRAK | XXXXXXXX
=| RELOP | EQ
'\n'| CHARCT | \n
)| CLPAR | XXXXXXXX
{| OPBRACE | XXXXXXXX
fim| ID | fim
<-| ASSIGN | XXXXXXXX
true| TRUE | XXXXXXXX
;| SCOLON | XXXXXXXX
palavra| ID | palavra
[| OPBRAK | XXXXXXXX

```
                    i|        ID |                    i
                    ]|    CLBRAK |                XXXXXXXX
                   <-|    ASSIGN |                XXXXXXXX
                  '\0'|   CHARCT |                   \0
                    ;|    SCOLON |                XXXXXXXX
                    }|   CLBRACE |                 XXXXXXXX
                    }|   CLBRACE |                 XXXXXXXX
                 posic|       ID |                posic
                   <-|    ASSIGN |                XXXXXXXX
                Procura|       ID |              Procura
                    (|     OPPAR |              XXXXXXXX
                    )|     CLPAR |              XXXXXXXX
                    ;|    SCOLON |                XXXXXXXX
                   if|       IF |              XXXXXXXX
                    (|     OPPAR |              XXXXXXXX
                 posic|       ID |                posic
                    >|     RELOP |                   GT
                    0|     INTCT |                   0
                    )|     CLPAR |              XXXXXXXX
                nocorr|       ID |              nocorr
                    [|    OPBRAK |              XXXXXXXX
                 posic|       ID |              posic
                    ]|    CLBRAK |              XXXXXXXX
                   <-|    ASSIGN |              XXXXXXXX
                nocorr|       ID |              nocorr
                    [|    OPBRAK |              XXXXXXXX
                 posic|       ID |              posic
                    ]|    CLBRAK |              XXXXXXXX
                    +|     ADOP |                 PLUS
                    1|     INTCT |                   1
                    ;|    SCOLON |                XXXXXXXX
                 else|     ELSE |               XXXXXXXX
                 call|     CALL |              XXXXXXXX
               Inserir|       ID |             Inserir
                    (|     OPPAR |              XXXXXXXX
                    ~|      NEG |              XXXXXXXX
                 posic|       ID |              posic
                    ,|    COMMA |                XXXXXXXX
                    i|       ID |                i
                    )|     CLPAR |              XXXXXXXX
                    ;|    SCOLON |                XXXXXXXX
                write|     WRITE |                 XXXXXXXX
                    (|     OPPAR |              XXXXXXXX
"\n\nNova palavra? (s/n): "|   STRING | "\n\nNova palavra? (s/n): "
                    )|     CLPAR |              XXXXXXXX
                    ;|    SCOLON |                XXXXXXXX
                read|      READ |                XXXXXXXX
                    (|     OPPAR |              XXXXXXXX
```

```
           c|        ID |                  c
           )|     CLPAR |               XXXXXXXX
           ;|    SCOLON |                 XXXXXXXX
           }|   CLBRACE |                  XXXXXXXX
        call|      CALL |               XXXXXXXX
ExibirTabela|        ID |            ExibirTabela
           (|     OPPAR |               XXXXXXXX
           )|     CLPAR |               XXXXXXXX
           ;|    SCOLON |                 XXXXXXXX
           }|   CLBRACE |                  XXXXXXXX
           }|   CLBRACE |                  XXXXXXXX
```