



1. Considere o programa

```
1 program Teste(input, output);

2 type maiusculas = 'A' .. 'Z';
3   palavra = array [1 .. 50] of maiusculas;
4 var m: integer;
5   w: palavra;

6 function MudarLetra(c: char): maiusculas;
7 begin if (c >= 'a') and (c <= 'z') then
8   MudarLetra := chr(ord('A')+ord(c)-ord('a'))
9   else MudarLetra := c
10 end;

11 procedure LerPalavra(var p: palavra; n: integer);
12 begin n := 0;
13   repeat
14     read('c');
15     if ((c >= 'a') and (c <= 'z')) or ((c >= 'A') and (c <= 'Z')) then
16       begin n := n+1;
17         p[n] := mudarletra(c)
18       end
19     until ((c < 'a') or (c > 'z')) and ((c < 'A') or (c > 'Z'))
20 end;

21 procedure OrdenarLetras(var p: palavra; n: integer);
22 var i, j, posmax: integer;
23   aux: maiusculas;
24 begin for i := n downto 2 do
25   begin posmax := 1;
26     for j := 2 to i do
27       if p[j] > p[posmax] then posmax := j;
28     aux := p[i]; p[i] := p[posmax]; p[posmax] := aux
29   end
30 end;

31 procedure EscreverPalavra(p: palavra; n: integer);
32 var i: integer;
33 begin for i := 1 to n do write(p[i], ' ');
34   writeln
```

```

35 end;

36 begin LerPalavra(w, m);
37     OrdenarLetras(w, m);
38     EscreverPalavra(w, m)
39 end.

```

- (a) Indique e corrija três erros que encontrar no procedimento **LerPalavra**, explicando porque são um erro. Utilize a numeração que identifica as linhas do programa.
 - (b) Descreva o resultado produzido pelo programa e a funcionalidade de cada um dos subprogramas, após a correcção.
 - (c) Após a correcção do programa apresente duas variáveis globais e duas variáveis locais, indicando, para cada uma, o tipo e o bloco em que foi declarada.
 - (d) Compare as passagens de parâmetros por valor e por referência, explicando como funcionam e apresentando as suas diferenças.
 - (e) Qual o método implementado no procedimento **OrdenarLetras**? Exemplifique-o no vector $p = [D \ B \ A \ C]$ ($n = 4$).
2. Escreva um programa que leia uma sequência de números inteiros em $[-100, 100]$, até ser lido um número que é o dobro do anterior e que, após a leitura, indique a soma dos números lidos.

3. Considere as declarações

```

const MaxL = 20;
      MaxC = 30;
type matriz = array [0 .. MaxL, 0 .. MaxC] of integer;

```

- (a) Desenvolva um subprograma que verifique se uma matriz A quadrada, de dimensão $n \times n$ e elementos inteiros, é estritamente diagonal dominante por linhas (EDDL), ou seja, se

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|, \quad i = 1, \dots, n.$$

- (b) Elabore um subprograma que remova a linha i e a coluna j de uma **matriz** ($m \times n$).
- (c) Escreva um programa que guarde uma sequência de inteiros dada pelo utilizador numa variável do tipo **matriz**. Se a **matriz** não for EDDL o programa deve remover uma linha e uma coluna indicadas pelo utilizador e imprimir a tabela modificada no monitor.
Apresente apenas o código novo, substituindo as declarações dos subprogramas já desenvolvidos pelo respectivo cabeçalho e reticências.
- (d) Elabore um subprograma recursivo para construir uma variável P do tipo **matriz**, com o Triângulo de Pascal de uma dimensão dada, atendendo a que

$$\begin{aligned}
 P_{m,0} &= P_{m,m} = 1 \\
 P_{m,n} &= P_{m-1,n-1} + P_{m-1,n}, \text{ se } 0 < n < m
 \end{aligned}$$