

- Este teste é de realização individual;
- Não é permitido qualquer tipo de consulta na sua resolução.
- Resolva cada grupo numa folha separada (note-se que folhas e páginas não são a mesma coisa!) utilizando a linguagem Visual Basic
- Coloque o Nome e Número em todas as folhas
- Duração: 2 horas (sem tolerância)

1. Pretende-se desenvolver uma aplicação informática para calcular o valor da facturação de uma portagem numa SCUD. O programa deverá ler um vector de  $n$  elementos ( $n \leq 15$ ) correspondendo aos nós de entrada na SCUD, em que o  $k$ -ésimo elemento ( $0 < k \leq n$ ) representa a distância desde o nó  $k$  da SCUD até ao nó actual, a saída. Deve depois ser lido uma série de valores correspondendo ao nó de entrada e à classe (1, 2, 3 e 4) de cada um dos veículos que saiu o nó actual. A leitura termina quando o nó tomar um valor negativo ou nulo. O valor da portagem a apagar por cada veículo é o produto da distância percorrida pelo preço dado pela tabela seguinte:

Classe	Preço (€/Km)
1	0,06
2	0,10
3	0,18
4	0,24

O programa deve apresentar o valor pago por cada veículo numa *ListBox* e o total facturado e número de veículos em duas *TextBox*. Elabore o programa pretendido e apresente a sua interface, indicando os nomes de todos os objectos.

2. Elabore uma função que recebe um número inteiro longo e retorna o número de algarismos pares que este contém. Utilizando a função, elabore então um programa que leia uma sequência de números e apresente numa *ListBox* cada número lido e o respectivo número de algarismos pares. A leitura termina quando for introduzido um número sem qualquer algarismo par. Apresente a interface do programa, indicando os nomes de todos os objectos.

3. Elabore uma subrotina que recebe uma frase (*string*) e uma sequência de caracteres como parâmetros e retorne uma frase semelhante à original em que todas as palavra que contenham a sequência de caracteres (não deve ser feita qualquer distinção entre maiúsculas e minúsculas) são eliminadas, bem como os espaços repetidos. A subrotina deve ainda retornar o número de palavras eliminadas. Elabore um programa que leia a frase e a sequência de caracteres e apresente em *TextBoxes* a frase corrigida e o número de palavras eliminadas. Apresente a interface do programa, indicando os nomes de todos os objectos.

Ex: “Abec decsef cBa dk cbAijkl” e “Ec”-> “cBa dk cbAijkl”

ISEP, 2005-01-21  
Ângelo Martins, Luís Vieira Lobo, Pedro Fortuna

Exercício	Cotação (valores)
1	7
2	8
3	5
Total	20

## Exercício 1

Facturação SCUT

Facturação de Portagem - SCUT

Nº de Nós  Listagem de Utentes

Nº de Utilizadores

Facturação

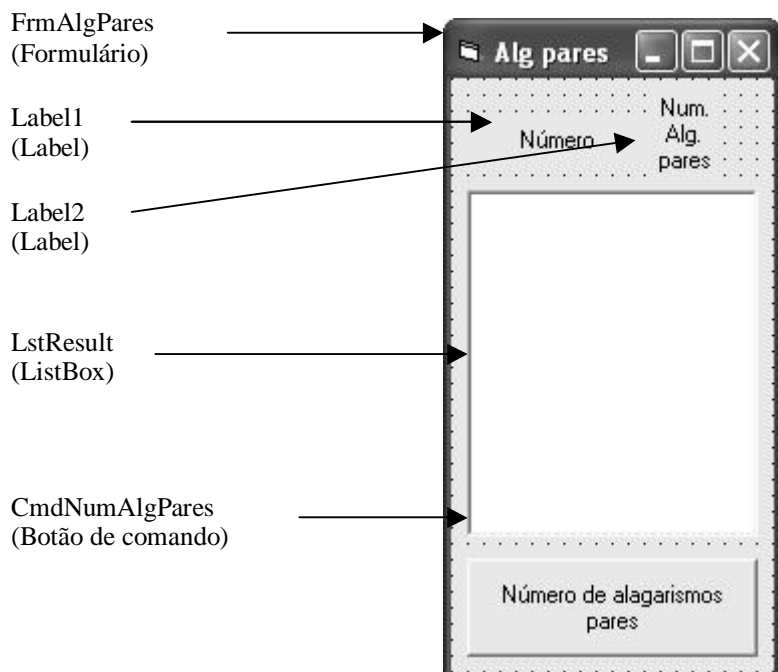
Calcular

1-7,2  
2-0,78  
3-0,84  
4-2,52  
5-1,56

Option Explicit

```
Private Sub cmdCalc_Click()  
    Dim v(15) As Single, nNos As Integer, nV As Integer  
    Dim vprecos(4) As Single, totPago As Single, x As Integer  
    Dim no As Integer, classe As Integer  
  
    vprecos(1) = 0.06  
    vprecos(2) = 0.12  
    vprecos(3) = 0.18  
    vprecos(4) = 0.24  
    lstUtil.Clear  
  
    nNos = Val(txtNNos)  
    If nNos < 1 Or nNos > 15 Then  
        MsgBox "Número de nós errado!"  
        Exit Sub  
    End If  
    For x = 1 To nNos  
        Do  
            v(x) = Val(InputBox("Distância do nó " & x))  
        Loop While v(x) <= 0  
    Next  
    nV = 0  
    totPago = 0  
    Do  
        no = Val(InputBox("Nº do nó de entrada", "Viatura nº" & (nV + 1)))  
        If no > 0 And no <= nNos Then  
            nV = nV + 1  
            Do  
                classe = Val(InputBox("Classe da viatura", "Viatura nº" & nV))  
            Loop Until classe >= 1 And classe <= 4  
            lstUtil.AddItem nV & " - " & vprecos(classe) * v(no)  
            totPago = totPago + vprecos(classe) * v(no)  
        End If  
    Loop While no > 0  
    txtNUtil.Text = nV  
    txtFact.Text = totPago  
End Sub
```

## Exercício 2



Option Explicit

```
Function NumAlgPares(ByVal n As Long) As Integer
    Dim alg As Integer, k As Integer
    n = Abs(n)
    Do
        alg = n Mod 10
        If alg Mod 2 = 0 Then
            k = k + 1
        End If
        n = n \ 10
    Loop Until n = 0
    NumAlgPares = k
End Function
```

```
Private Sub CmdNumAlgPares_Click()
    Dim n As Long, k As Integer
    LstResult.Clear
    Call LerNúmero(n)
    k = NumAlgPares(n)
    Do While k > 0
        ' É inserido um ou dois Tabs de modo a alinhar sempre o número
        ' de algarismos pares do respectivo número, independentemente
        ' do tamanho deste. O tipo de letra considerado é MS Sans Serif
        ' de tamanho 8.
        ' Repare também que vbTab faz o mesmo efeito que Chr(13).
        If Abs(n) < 10000000# Then
            LstResult.AddItem n & vbTab & vbTab & k
        Else
            LstResult.AddItem n & vbTab & k
        End If
        Call LerNúmero(n)
        k = NumAlgPares(n)
    Loop
End Sub
```

```

Sub LerNúmero(ByRef n As Long)
  Dim aux As String, NumAux As Double, NumCorrecto As Boolean
  NumCorrecto = False
  Do
    ' O valor de defeito 1 definido na função de InputBox destina-se
    ' a facilitar o fim da entrada de dados, atendendo a que, desta forma,
    ' o utilizador só precisa de carregar no Ok para terminar a entrada
    ' de dados.
    aux = InputBox("Introduza um número com algarismos pares " & _
      "(Para terminar introduza um número sem algarismos pares)", _
      "Entrada de dados", 1)
    If IsNumeric(aux) Then
      NumAux = Val(aux)
      If NumAux < -2147483648# Or NumAux > 2147483647 Then
        MsgBox "Fora dos limites de um número inteiro longo", _
          vbCritical, "Erro nos dados"
      ElseIf NumAux <> Fix(NumAux) Then
        MsgBox "Não é número inteiro", vbCritical, _
          "Erro nos dados"
      Else
        n = NumAux: NumCorrecto = True
      End If
    Else
      MsgBox "Não é número", vbCritical, "Erro nos dados"
    End If
  Loop Until NumCorrecto
End Sub

```