

Instituto Superior de Economia e Gestão
UNIVERSIDADE TÉCNICA DE LISBOA

Mestrado Decisão Económica e Empresarial

COMPUTAÇÃO

Sumário:

Programação em VBA.
Variáveis indexadas. Funções e procedimentos.

Exercícios aula 3 Dúvidas ?

1. Escreva um macro que escreve 3 inteiros por ordem decrescente.
2. Programe um macro a determinação do custo de uma encomenda sabendo que o preço base é de 100 e que é efectuado um desconto em função da quantidade de acordo com a seguinte tabela:

quantidade	desconto
>= 150	10%
>= 300	15%
>= 500	20%
>= 1000	25%
3. Escreva um macro para determinar se um número é primo.
4. Escreva um macro para determinar o máximo divisor comum.
5. Escreva um macro que informa se um número é par ou ímpar

Variáveis indexadas

- **Array (vector)**
Permitem armazenar numa variável vários valores desde que sejam todos do mesmo tipo

33 ← a

Agora variável b

12	15	18	18	12	15	16	10	13	14
0	1	2	3	4	5	6	7	8	9

b(0)=12
b(1)=15
b(9)=14

Valor do índice deve ser inteiro e por omissão começa em zero

Variáveis indexadas

- **Array (vector)**
Permitem armazenar numa variável vários valores desde que sejam todos do mesmo tipo

Não necessariamente numéricos

variável nomes

Pedro	Mariana	Joana	António	Guilherme	Maria	Manuel
0	1	2	3	4	5	6

nomes(0)="Pedro"
nomes(1)="Mariana"
nomes(6)="Manuel"

Valor do índice deve ser inteiro e por omissão começa em zero

Variáveis indexadas

- **Declaração Estática I**
vector
Dim nome_do_vector(num_de_elementos) **As** Tipo

exemplos:

Dim idade(8) As Integer (de 0 à 7)
Dim Medias(12) As Double (de 0 à 11)
Dim nomes(16) As String (de 0 à 15)

Adicionalmente com Dim os valores numéricos são inicializados a zeros e os alfanuméricos a strings nulas

Variáveis indexadas

- **Declaração Estática II**
vector
Dim nome_do_vector(menorIndice To maiorIndice) **As** Tipo

exemplos:

Dim alunosNovos(100 To 200) As String
Dim Medias(-100 To 100) As Integer

Adicionalmente com Dim os valores numéricos são inicializados a zeros e os alfanuméricos a strings nulas

Variáveis indexadas

- Declaração Dinâmica**
Quando não sabemos a dimensão no vector no momento da sua declaração
Dim nome_do_vector() As Tipo
Dim dados() As Integer
Quando é conhecida a dimensão pretendida
ReDim nome_do_vector()
ReDim dados(100)
ReDim Preserve nome_do_vector()
para ajustar dimensão sem perder os valores já existentes no vector

Variáveis indexadas

- Utilidade**
Exemplo: uma empresa de produtos lacteos tem 100 produtos cada um com o seu preço e pretende actualizar o IVA que era de 6% e passou a ser de 23%.

```

100 variáveis
Declaração
Dim preço1 As Single
Dim preço2 As Single
...
Dim preço100 As Single
Dim actualiza As Single
actualiza = 0.23/0.06
Actualização
preço1 = actualiza*preço1
preço2 = actualiza*preço2
...
preço100 = actualiza*preço100
    
```

```

1 variável indexada com 100 elementos
Declaração
Dim preços(1 To 100) As Single
Dim actualiza As Single
Dim i As Integer
actualiza = 0.23/0.06
Actualização
For i=1 To 100
    Preços(i) = actualiza*preços(i)
Next i
    
```

Variáveis indexadas

- Exemplo 1**
Public Sub Xbarra()
Dim dados(1 To 10) As Integer
Dim i As Integer
Dim Soma As Integer
Dim Média As Double

For i = 1 To 10
dados(i) = ActiveSheet.Cells(i, 1)
Next i
Soma = 0
For i = 1 To 10
Soma = Soma + dados(i)
Next i
MsgBox ("Soma=" & Soma)
Média = Soma / 10
MsgBox ("Média=" & Média)
End Sub

Variáveis indexadas

- Exemplo 2**
Public Sub Xbarra()
Dim dados() As Integer
ReDim dados(1 To 10) As Integer
Dim i As Integer
Dim Soma As Integer
Dim Média As Double
For i = 1 To 10:dados(i) = ActiveSheet.Cells(i, 1):Next i
Soma = 0
For i = 1 To 10:Soma = Soma + dados(i):Next i
MsgBox ("Soma=" & Soma)
Média = Soma / 10
MsgBox ("Média=" & Média)
ReDim Preserve dados(1 To 20)
For i = 11 To 20:dados(i) = ActiveSheet.Cells(i, 1):Next i
Soma=0
For i = 1 To 20:Soma = Soma + dados(i):Next i
MsgBox ("Soma=" & Soma)
Média = Soma / 20
MsgBox ("Média=" & Média)
End Sub

Variáveis indexadas

- Matrizes**
Dim nome_matriz(num_linhas,num_colunas) As Tipo
exemplo
Dim Mat1(4,3) As Integer

0	0	0
0	0	0
0	0	0
0	0	0

.....

1	5	7
3	2	4
11	10	6
8	12	13

Mat1(2,2)=6
Mat1(0,0)=1
Mat1(3,2)=13

Variáveis indexadas

- Manipulação de Matriz ciclos encadeados**
Public Sub mat()
Dim mat(1 To 4, 1 To 4) As Integer
Dim i As Integer
Dim j As Integer
Dim k As Integer
For i = 1 To 4
For j = 1 To 4
mat(i, j) = ActiveSheet.Cells(i, j)
Next j
Next i
k = 6
For i = 1 To 4
For j = 1 To 4
ActiveSheet.Cells(i + k, j + k) = mat(i, j)
Next j
Next i
End Sub

Procedimentos e Funções

- Os **procedimentos** ou subrotinas destinam-se a realizar um conjunto de tarefas mas não têm necessariamente que devolver qualquer resultado
- Uma **função** destina-se a realizar um conjunto de tarefas e a devolver um resultado

DEE - Computação 2010/11 | Lápido 13

Funções

Function Nome (arg1,arg2,...) Tipo de dado

' Lista de instruções

Nome=resultado

End Function

Exemplo:

```
Function txLucro(v As Single,c As Single) As single
txLucro= (v-c)/v
End Function
```

DEE - Computação 2010/11 | Lápido 14

Função factorial

```
Private Sub Combina()
Dim n As Integer
Dim p As Integer
Dim Comb As Double
n = InputBox("no total quantos elementos")
p = InputBox("grupos de quantos elementos")
If n > 0 And p > 0 And n >= p Then
    Comb = factorial(n) / factorial(n - p) / factorial(p)
    MsgBox (" num. de combinações " & Comb)
Else
    MsgBox (" Erro, dados inválidos ")
End If
End Sub

Public Function factorial(n As Integer) As Double
Dim i As Integer, f As Double
f = 1
For i = 2 To n
    f = f * i
Next i
factorial = f
End Function
```

DEE - Computação 2010/11 | Lápido 15

Procedimentos

Sub Nome (arg1,arg2,...)

' Lista de instruções

End Sub

Call Nome()

Exemplo:

```
Public Sub exemplo()
Dim x As Integer
Dim y As Integer
x = InputBox("introduza um inteiro")
y = InputBox("introduza um inteiro")
MsgBox ("(" & x & ", " & y & ")=>" & " Soma:" & x + y)

Public Sub chamada()
Call exemplo
End Sub
```

DEE - Computação 2010/11 | Lápido 16

Passagem de parâmetros

- ByVal** O que é passado à função ou ao procedimento é o valor corrente da variável. Dentro da função ou procedimento os valores são alterados mas no retorno ao programa os valores são iguais aos que eram na altura da chamada
- ByRef** O que é passado à função ou ao procedimento é o endereço da variável, uma autorização para alterar o seu valor. Dentro da função ou procedimento os valores são alterados no retorno ao programa ficam alterados de acordo como dentro do bloco

DEE - Computação 2010/11 | Lápido 17

Exemplo de passagem de argumentos

```
Public Sub NãoTroca(ByVal a As Integer, ByVal b As Integer)
Dim temp As Integer
temp = a:a = b:b = temp
End Sub

Public Sub troca(ByRef a As Integer, ByRef b As Integer)
Dim temp As Integer
temp = a:a = b:b = temp
End Sub

Public Sub principal()
Dim x As Integer
Dim y As Integer
x = 1:y = 2
MsgBox ("antes de nãoTroca (x,y)=( " & x & ", " & y & ")")
Call NãoTroca(x, y)
MsgBox ("depois de nãoTroca (x,y)=( " & x & ", " & y & ")")
Call troca(x, y)
MsgBox ("depois de Troca (x,y)=( " & x & ", " & y & ")")
End Sub
```

DEE - Computação 2010/11 | Lápido 18

Usar as funções do excel

Definir nome => dados



```
Med = Application.worksheetFunction.Average(Range("dados"))
```

exercícios

1. Reescrever os programas da aula anterior usando funções ou procedimentos.
2. Escrever um macro que calcula a média de um conjunto de dados armazenado num vector.
3. Escrever um macro para determinar o produto de duas matrizes.