


Instituto Superior de Economia e Gestao
UNIVERSIDADE TECNICA DE LISBOA

Mestrado Decisão Económica e Empresarial COMPUTAÇÃO

Sumário:

Programação em VBA.
Conceitos básicos. Macros. Variáveis e tipos de dados.

Exercícios da aula anterior

- Escreva o algoritmo para atribuição de mandatos pelo método d'Hondt.
- Represente em binário (com 8 bits) os inteiros 23, -63 e -1.
- Qual o número representado por 11011011 se se tratar de um inteiro sem sinal ? E se se tratar de um inteiro com sinal ?
- Represente em precisão simples 10.125.
- Qual o número representado por 1100 0001 0100 0100 0000 0000 0000 0000 ?

Bom trabalho e até sexta-feira !

Método d'Hondt

O círculo eleitoral "X" elege 7 deputados e concorrem 4 partidos: A, B, C e D. Apurados os votos, a distribuição foi a seguinte: A - 12.000 votos; B - 7.500 votos; C - 4.500 votos; e D - 3.000 votos. Da aplicação do método de Hondt resulta a seguinte série de quocientes:

divisor	Partido			
	A	B	C	D
1	12000	7500	4500	3000
2	6000	3750	2250	1500
3	4000	2500	1500	1000
4	3000	1875	1125	750

http://pt.wikipedia.org/wiki/M%C3%A9todo_D'Hondt

Solução exercícios aula 1

1. Método d'Hondt

Seja V_j o número total de votos obtido pela lista j e s_j o número de mandatos já atribuídos ao partido j o mandato seguinte será atribuído ao partido k tal que

$$R_k = \text{Max}_j \frac{V_j}{s_j + 1}$$

Em caso de empate é escolhido partido menos votado

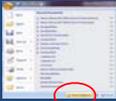
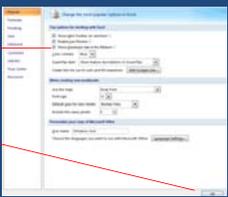
Solução exercícios aula 1

- Represente em binário (com 8 bits) os inteiros 23, -63 e -1.
R: $23 = 0001\ 0111_{(2)}$; $-63 = 1100\ 0001_{(2)}$; $-1 = 1111\ 1111_{(2)}$.
- Qual o número representado por 11011011 se se tratar de um inteiro sem sinal ? E se se tratar de um inteiro com sinal ?
R: 219 ; -37.
- Represente em precisão simples 10.125.
R: 0100 0001 0 010 0010 0000 0000 0000 0000
- Qual o número representado por 1100 0001 0100 0100 00000000 00000000 ?
R: -12.25

Excel/VBA

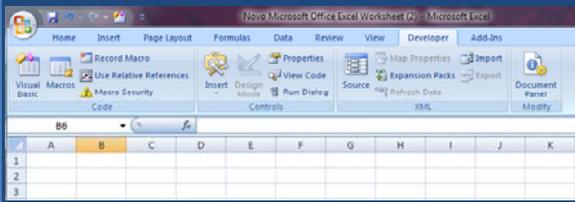
- Excel é uma folha de cálculo, uma **aplicação**, que permite criar e manusear qualquer tipo de lista com funcionalidades das mais simples às mais avançadas. Com a linguagem Visual Basic aumentam-se as potencialidades desta folha de cálculo.
- Microsoft Visual Basic for Applications (VBA) é uma linguagem de programação baseada no Microsoft Visual Basic. Permite escrever código para executar ações num documento e/ou no seu conteúdo.
- Microsoft Visual Basic é um ambiente de programação que vem com a instalação do Excel. Para o tornar facilmente acessível:

VBA acessível no Excel (2007)

1. Botão do Office 
2. Excel options 
3. Show developer tab in the ribbon -> OK 

DEE - Computação 2010/11 | Lápido 7

Tab Developer



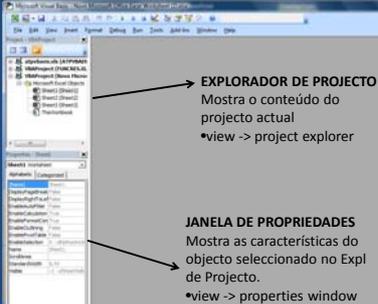
DEE - Computação 2010/11 | Lápido 8

O editor do VBA

- IDE (Integrated Development Environment)
- Para aceder ao IDE partir do Excel
 - (ALT+F11) ou
 - Developer ->  Visual Basic
 -
- Para voltar à folha de cálculo
 - (ALT+F11) (novamente)

DEE - Computação 2010/11 | Lápido 9

O editor do VBA

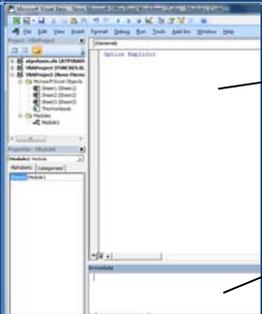


EXPLORADOR DE PROJECTO
Mostra o conteúdo do projecto actual
*view -> project explorer

JANELA DE PROPRIEDADES
Mostra as características do objecto seleccionado no Expl de Projecto.
*view -> properties window

DEE - Computação 2010/11 | Lápido 10

O editor do VBA



MÓDULO
Uma folha em branco para escrever código.
*insert -> Module

IMMEDIATE WINDOW
Para ajudar a testar o código.
*view -> immediate window

DEE - Computação 2010/11 | Lápido 11

Macro

É um programa escrito em VBA, que indica a uma aplicação como o Excel quais os passos a dar para atingir um objectivo específico, dito de outro modo, é a descrição formalizada das tarefas que se pretende automatizar.

- Servem para automatizar tarefas repetitivas.

DEE - Computação 2010/11 | Lápido 12

Criar um Macro

1. Abrir Excel
2. Developer -> Record a Macro
3. Preencher
 - nome,
 - escolher tecla para atalho e
 - escrever descrição
- OK
4. Executar as acções pretendidas
5. Developer -> Stop Record a Macro

DEE - Computação 2010/11

Leção 13

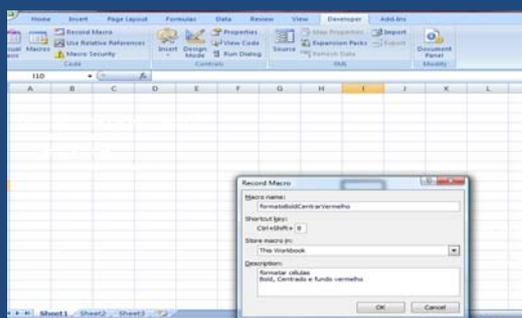
EXEMPLO

- Associar a (Ctrl+Shipt+B)
- a seguinte formatação
 - Bold
 - Centrado
 - Fundo Vermelho

DEE - Computação 2010/11

Leção 14

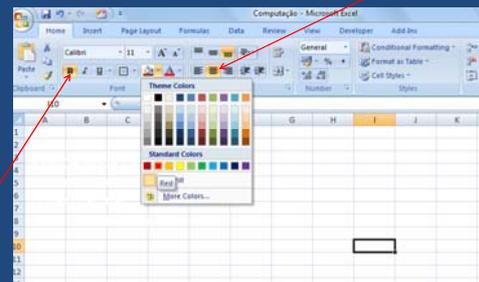
EXEMPLO Criar um Macro (2) (3)



DEE - Computação 2010/11

Leção 15

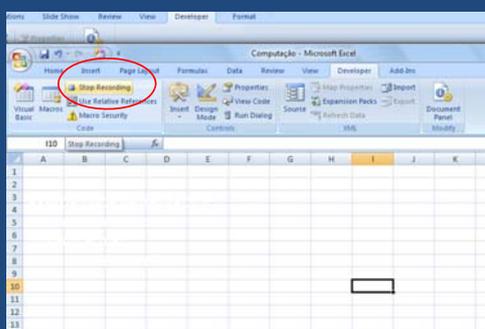
EXEMPLO Criar um Macro (4)



DEE - Computação 2010/11

Leção 16

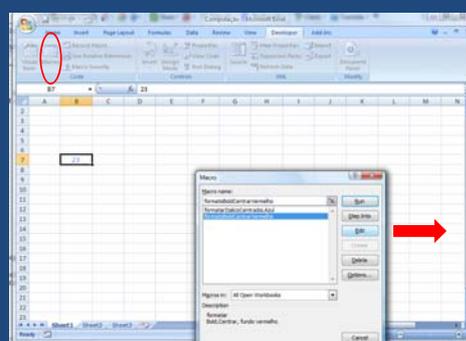
EXEMPLO Criar um Macro (5)



DEE - Computação 2010/11

Leção 17

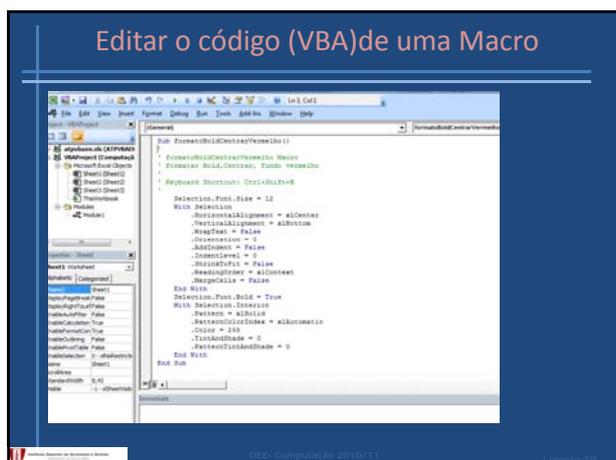
Visualizar as Macros



DEE - Computação 2010/11

Leção 18

Editar o código (VBA) de uma Macro



Executar uma Macro

- Usando o atalho definido

ou,

- Developer -> Macros
 - Seleccionar
 - RUN

Escrever código legível

- Indentação
 - Deve permitir perceber facilmente a estrutura do código;
 - Usar os *tabs*
conselho predefinir *tab* a 4
(Menu VB -> tools -> options ...)
- Comentários
 - Devem ser inseridos para facilitar a compreensão do código
 - Uma linha que comece por ' (plica) ou Rem é ignorada pelo compilador

Variáveis

Localização de memória em que a informação pode ser guardada de modo a ser usada por um macro.

Caracterizada por

- Nome
- Tipo de dados que pode armazenar e
- Âmbito

Variáveis - declaração

Chama-se declaração à criação de uma variável.

Atribui-se-lhe: nome, tipo de dados e fixa-se o seu âmbito.

Antes de utilizar uma variável esta deve ser declarada.

Dim variável **As** tipo de dado

Variável - nome

Regras:

- Deve começar com uma letra ou com *underscore*
- Não pode contêr ponto;
- Não pode exceder 255 caracteres.
- Deve ser único na área que é usado
- Não pode coincidir com palavras reservadas

Variável - não pode ter os nomes:

And (Bitwise) And (Condition) As Boolean ByRef Byte
 ByVal Call Case CBool CByte CDate
 CDbI CLnt CLng Const CSng CStr
 Date Dim Do Double Each Else
 Elseif End EndIf Error False For
 Function Get GoTo If Integer Let
 Lib Long Loop Me Mid Mod
 New Next Not Nothing Option Or (Bitwise)
 Or (Condition) Private Public ReDim REM Resume
 Select Set Single Static Step String
 Sub Then To True Until vbCrLf
 vbTab With While XOR



DEE - Computação 2010/11

Lição 26

Option explicit

- Existe a possibilidade de não declarar explicitamente uma variável. Nesta caso a primeira vez que o nome é usado a variável é implicitamente declarada.

NÃO É SEGURO por isso convem trabalhar com a opção Option explicit

- Escrever no início do código
- Developer -> VB -> tool -> options -> require variable declaration



DEE - Computação 2010/11

Lição 26

Variável – atribuição de valores

```
Sub exemplo()
  Dim valor As Integer
  valor = 5
End Sub
```

= Afectação

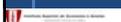


DEE - Computação 2010/11

Lição 27

Tipos de dados

- Byte
 - Integer
 - Long
 - Single
 - Double
 - String
 - Currency
 - Date
 - Variant
- inteiros
- Virgula flutuante



DEE - Computação 2010/11

Lição 28

Tipo Byte

Números naturais entre 0 e 255

```
Sub ExemploByte()
  Dim Idade As Byte
End Sub
```

```
Sub ErroByte()
  Dim Valor As Byte
  Valor=777
End Sub
```



DEE - Computação 2010/11

Lição 29

Tipo Integer

- Inteiros entre -32768 e 32767

```
Sub ExemploInt()
  Dim Nalunos As Integer
End Sub
```

```
Sub ExemploInt()
  Dim Nalunos%
End Sub
```



DEE - Computação 2010/11

Lição 30

Tipo Long

Inteiros entre $-2,147,483,648$ e $2,147,483,647$

```
Sub ExemploLong()
    Dim Populacao As Long
End Sub
```

```
Sub ExemploLong()
    Dim Populacao@
End Sub
```



DEE - Computação 2010/11

Lição 21

Tipo Single

Negativos de $1.401298e^{-45}$ a $3.402823e^{38}$
Positivos de $1.401298e^{-45}$ a $3.402823e^{38}$

```
Sub ExemploSingle()
    Dim Distancia As Single
End Sub
```

```
Sub ExemploSingle()
    Dim Distancia!
End Sub
```



DEE - Computação 2010/11

Lição 22

Tipo Double

Negativos de $-1.79769313486231e^{308}$ a
 $-4.94065645841247e^{-324}$

Positivos de $4.94065645841247e^{-324}$ a
 $1.79769313486231e^{308}$

```
Sub ExemploDouble()
    Dim Distancia As Double
End Sub
```

```
Sub ExemploDouble()
    Dim Distancia#
End Sub
```



DEE - Computação 2010/11

Lição 23

Tipo String

- É um caracter ou um conjunto de caracteres (texto)

```
Sub ExemploString()
    Dim Nome As String
End Sub
```

```
Sub ExemploString()
    Dim Nome$
End Sub
```



DEE - Computação 2010/11

Lição 24

Tipo Currency

- Entre $922,337,203,685,477.5808$ e $922,337,203,685,477.5807$.

- As vírgulas são para facilitar a leitura
- Não colocar € ou outro símbolo

```
Sub ExemploCurrency()
    Dim Salario As Currency
End Sub
```



DEE - Computação 2010/11

Lição 25

Tipo Date

```
Sub ExemploDate()
    Dim DataNascimento As Date
    DataNascimento = #1/10/2010#
End Sub
```



DEE - Computação 2010/11

Lição 26

Tipo Variant

```
Sub ExemploVaR()
    Dim Nome As Variant
    Dim Contracto As Variant
    Dim SalarioHora As Variant
    Dim InicioContr As Variant
```

```
    Nome = "Manuel Joaquim"
    Contracto = 0
    SalarioHora = 10
    InicioContr = #1/1/1980#
End Sub
```

Não é boa prática
Exige reserva de
muito espaço

Âmbito de uma variável

- Tempo de vida de uma variável

Globais/locais

Option Explicit

Variável Global

```
Dim NomeF As String
```

Sub ExemploGlobalLocal() Variável Local

```
    Dim NomeP As String
    NomeP = "Maria"
    NomeF = "Silva"
End Sub
```

Variável -Private/Public

```
Private NomeF As String
```

```
Sub PrivPub()
    Dim NomeP As String
```

```
    NomeP = "Maria"
    NomeF = "Silva"
End Sub
```

Só pode ser acedida
por código do
mesmo módulo

Variável -Private/Public

Option Explicit

```
Private NomeF As String
Public NomeC As String
```

```
Sub ExemploPrivPubOK()
    Dim NomeP As String
```

```
    NomeP = "Maria"
    NomeF = "Silva"
    NomeC = NomeP & " " & NomeF
End Sub
```

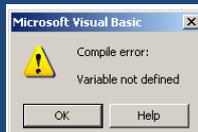
Só pode ser acedida
por código do
mesmo módulo ou
fora dele

Variável -Private/Public

```
MÓDULO1:
Option Explicit
Private NomeC As String
```

```
MÓDULO2:
Option Explicit
Private NomeF As String
Private NomeP As String
```

```
Sub ExemploPrivPubKO()
    Dim NomeP As String
    NomeP = "Maria"
    NomeF = "Silva"
    NomeC = NomeP & " " & NomeF
End Sub
```



Equação de 2º grau

```
Option Explicit
Public Sub Eq2Grau()
    Dim a As Double
    Dim b As Double
    Dim c As Double
    Dim x1 As Double
    Dim x2 As Double
    Dim delta As Double
    a = Range("a2")
    b = Range("b2")
    c = Range("c2")
    Range("a3:c7").Clear
    Range("a4") = " Vou calcular as raizes da equação "
    If a <> 0 Then
        delta = b * b - 4 * a * c
        If delta > 0 Then
            x1 = (-b + Sqr(delta)) / (2 * a)
            x2 = (-b - Sqr(delta)) / (2 * a)
            Range("a5") = " Neste caso há 2 raizes reais "
            Range("b5") = "x1:"
            Range("c5") = x1
            Range("b7") = "x2:"
            Range("c7") = x2
        ElseIf delta = 0 Then
```

```
            x1 = (-b) / (2 * a)
            Range("a5") = " Neste caso só há uma raiz, tem multiplicidade 2"
            Range("b5") = "x1:"
            Range("c5") = x1
        Else
            Range("a5") = " Neste caso as raiz são complexas!!!!!"
            Range("b5") = "x1:"
            Range("c5") = "-b / (2 * a) & "+ " & Sqr(-delta) / (2 * a) & "i"
            Range("b7") = "x2:"
            Range("c7") = "-b / (2 * a) & "+ " & Sqr(-delta) / (2 * a) & "i"
        End If
    ElseIf b <> 0 Then
        Range("a5") = " Trata-se de uma equação de 1º grau"
        Range("b5") = "x:"
        Range("c5") = -b / c
    ElseIf c <> 0 Then
        Range("a5") = " Não tem solução"
        Range("b5") = "0 = " & c & " !!!!!!!!!!!!!!!!"
    Else
        Range("a5") = " Condição Universal"
        Range("b5") = "0 = 0 !!!!!!!!!!!!!!!!"
    End If
End Sub
```