

Mestrado Decisão Económica e Empresarial COMPUTAÇÃO

PERT/CPM
UserForms

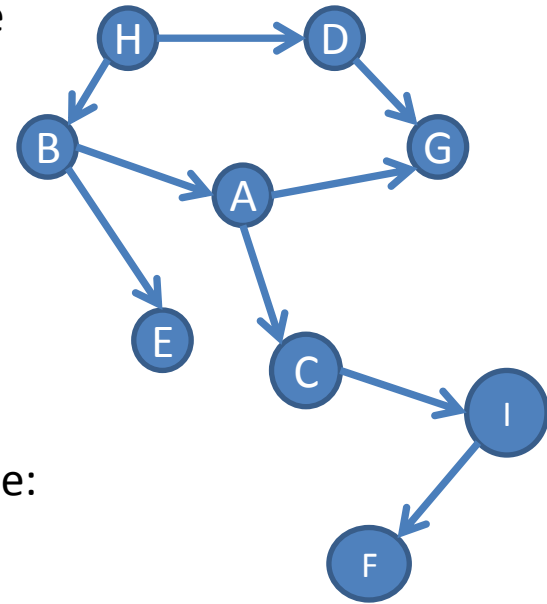
Sumário

- PERT/CPM
- Objectos em VBA

PERT/CPM

Considere-se a rede de tarefas com a seguinte ordenação topológica:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| H | B | D | E | A | G | C | I | F |



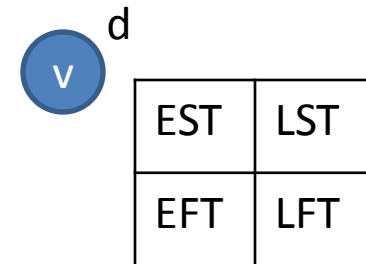
Associado a cada vértice/tarefa v calculam-se:

EFT(v) – Earliest Finish Time

LFT(v) – Latest Finish Time

LST(v) – Latest Start Time

EST(v) – Earliest Start Time



Nota: Assume-se que o projecto inicia-se no instante 0

PERT/CPM

Algoritmo *forwardpass*:

Para todo o vértice **v** segundo a ordenação topológica

Se o vértice **v** não tem predecessores **Então**

$$\text{EST}(v) = 0$$

$$\text{EFT}(v) = \text{duração da tarefa } v$$

Senão

$$\text{EST}(v) = \text{máximo EFT}(w) \text{ de todos os predecessores } w \text{ de } v$$

$$\text{EFT}(v) = \text{EST}(v) + \text{duração da tarefa } v$$

Fim Se

Fim Para

‘Calculo da duração do projecto (**Project Finish Time**)

PFT = máximo **EFT**(v) de todos os vértices **v**

PERT/CPM

Algoritmo *backwardpass*:

Para todo o vértice **v** segundo a ordenação topológica **inversa**

Se o vértice **v** não tem sucessor **Então**

$$LFT(v) = PFT$$

$$LST(v) = LFT(v) - \text{duração da tarefa } v$$

Senão

$$LFT(v) = \text{mínimo } LST(w) \text{ de todos os sucessores } w \text{ de } v$$

$$LST(v) = LFT(v) - \text{duração da tarefa } v$$

Fim Se

Fim Para

PERT/CPM

Indicadores:

- Folga de uma tarefa v é igual a $LST(v) - EST(v)$
- As tarefas com folga igual a zero formam o caminho crítico do projecto.
- O PFT é igual à soma dos tempos das tarefas no caminho crítico
- Para que o projecto termine no tempo previsto, as tarefas têm de iniciar entre $EST(v)$ e $LST(v)$

Comunicação com o utilizador

- Status bar
- Message Boxes
- Input Boxes
- User forms

Status bar

- Mensagens simples podem ser visualizadas na barra de estado do programa.

`Application.StatusBar = "A calcular...por favor aguarde"`

O utilizador pode não prestar atenção à mensagem!!

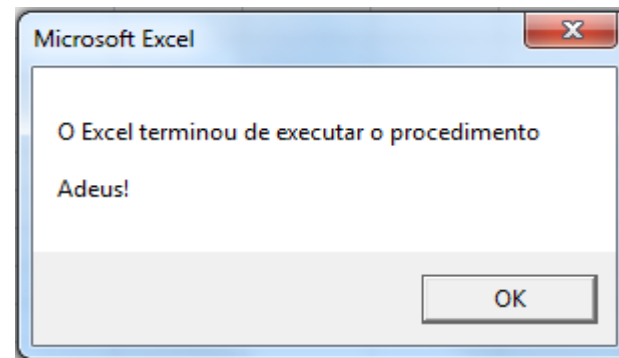
Message Box

- Informar o utilizador sobre a execução de um procedimento
- Pedir ao utilizador que tome uma simples acção: Sim vs. Não
- Avisar o utilizador da existência de algum erro
- Etc...

Message Box

- Message Box simples

Dim mensagem As String
mensagem = "Olá Mundo"
MsgBox mensagem



mensagem= "O Excel terminou de executar o procedimento" _
 & vbCr & vbCr & "Adeus!"

MsgBox mensagem

Message Box

Message Box com botões Sim/Não

```
Dim op As Long
```

```
op = MsgBox("Deseja formatar o computador?", vbYesNo)
```

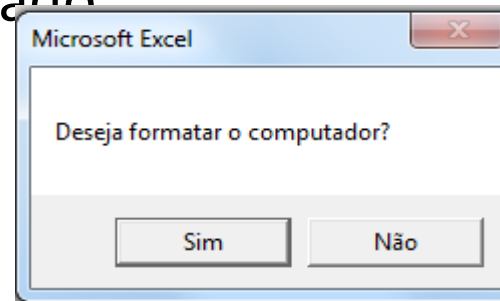
```
If op=vbYes then
```

```
    MsgBox "Contacte o seu programador"
```

```
Else
```

```
    MsgBox "Computador não formatado"
```

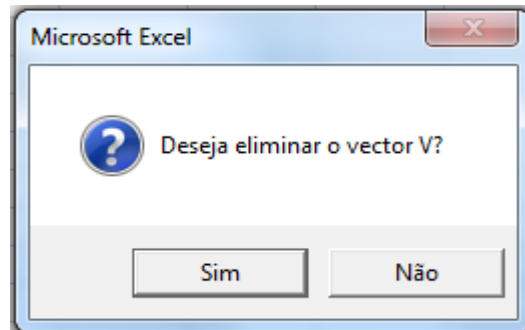
```
End if
```



Message Box

- Colocando um icone numa MsgBox

```
op = MsgBox("Deseja eliminar o vector V?",  
vbYesNo + vbQuestion)
```



Message box

TABLE 13.1: Message Box Types, Controlled by the *buttons* Argument

| VALUE | CONSTANT | BUTTONS |
|-------|--------------------|----------------------|
| 0 | vbOKOnly | OK |
| 1 | vbOKCancel | OK, Cancel |
| 2 | vbAbortRetryIgnore | Abort, Retry, Ignore |
| 3 | vbYesNoCancel | Yes, No, Cancel |
| 4 | vbYesNo | Yes, No |
| 5 | vbRetryCancel | Retry, Cancel |

Message Box

TABLE 13.5: Constants for Selected Buttons

| VALUE | CONSTANT | BUTTON SELECTED |
|-------|----------|-----------------|
| 1 | vbOK | OK |
| 2 | vbCancel | Cancel |
| 3 | vbAbort | Abort |
| 4 | vbRetry | Retry |
| 5 | vbIgnore | Ignore |
| 6 | vbYes | Yes |
| 7 | vbNo | No |

Message Box

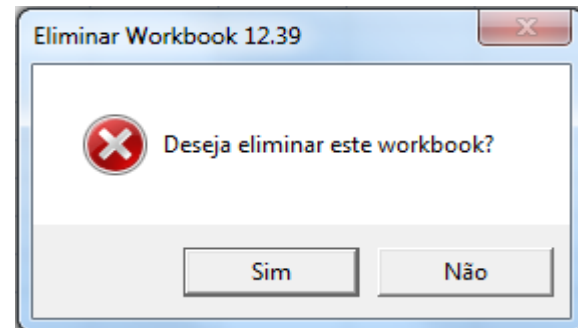
TABLE 13.2: Arguments for Message Box Icons

| VALUE | CONSTANT | DISPLAYS |
|-------|----------------------------|------------------------|
| 16 | <code>vbCritical</code> | Stop icon |
| 32 | <code>vbQuestion</code> | Question mark icon |
| 48 | <code>vbExclamation</code> | Exclamation point icon |
| 64 | <code>vbInformation</code> | Information icon |

Message Box

Títulos para MsgBox

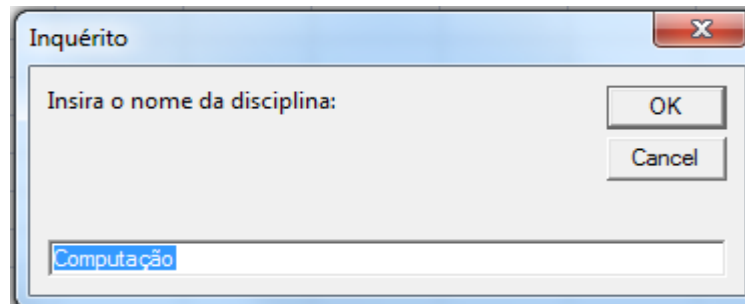
```
op = MsgBox("Deseja eliminar este workbook?", _  
           vbYesNo + vbCritical, "Eliminar Workbook 12.39")
```



Input Box

Dim str As String

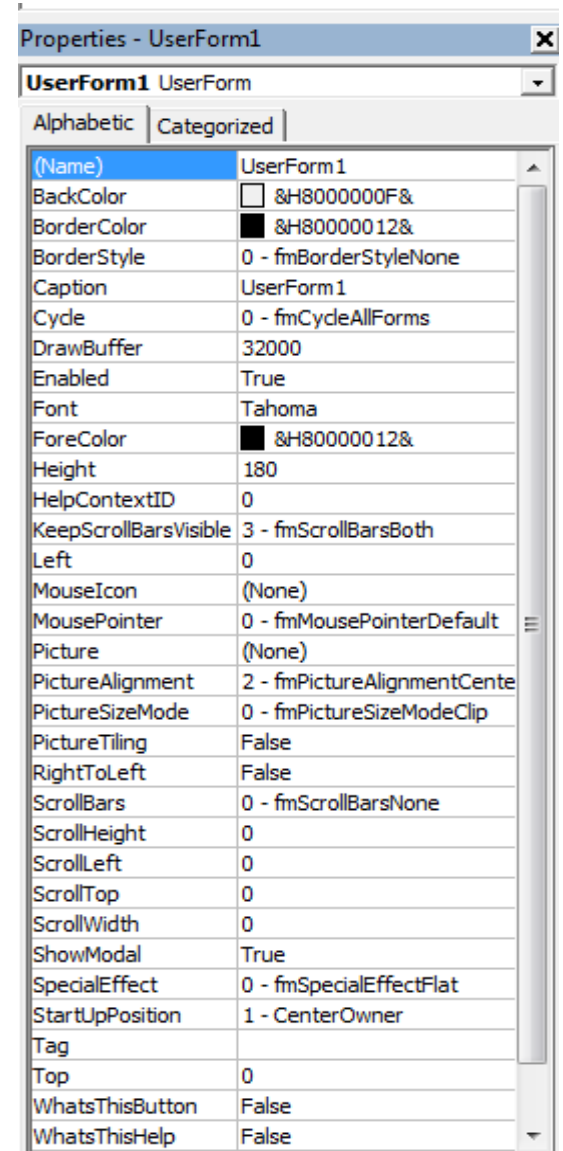
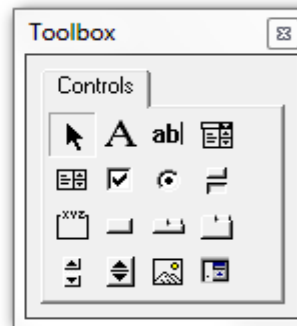
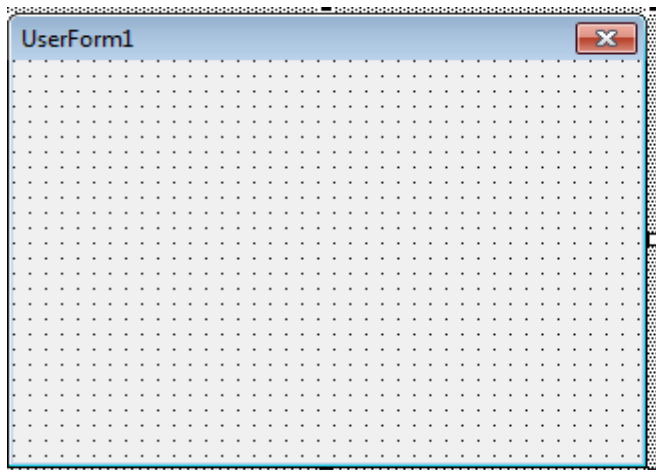
```
str = InputBox("Insira o nome da disciplina:", _  
    "Inquérito", "Computação")
```



UserForms

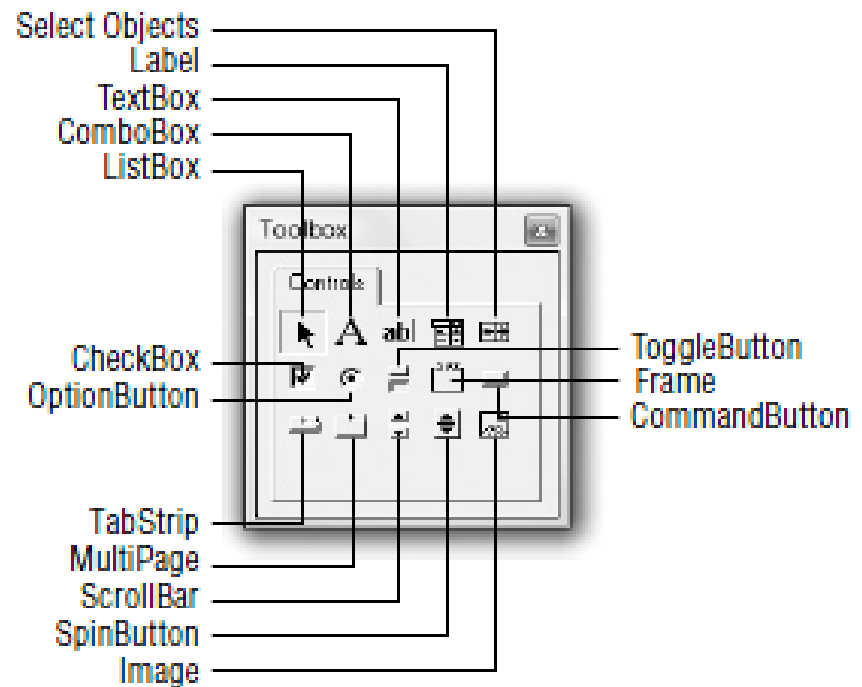
No developer:

- Insert // UserForm
- View // Toolbox
- View // Properties Window



Controles do UserForm

FIGURE 14.4
Use the Toolbox to add controls to the user form.



Descrição dos controlos

- **Select Objects**

Altera o ponteiro do rato para o modo de selecção.

- **Label**

Cria uma caixa de texto informativa.

- **TextBox**

Cria uma caixa de texto editável.

- **ListBox**

Cria uma lista de itens. O utilizador pode apenas escolher um item da lista.

- **ComboBox**

Cria um controlo que combina uma TextBox com uma ListBox

Descrição dos controlos

- **CheckBox**

Cria um quadrado com uma label que permite ao utilizador activar ou desactivar uma determinada opção.

- **OptionButton**

Cria um círculo com uma label que permite ao utilizador activar ou desactivar uma determinada opção.

- **ToggleButton**

Cria um Button com dois estados: On/Off

- **Frame**

Cria uma moldura com uma label que permite agrupar controlos.

Descrição dos controlos

- **CommandButton**

Cria um button que activa um certo comando/procedimento.

- **TabStrip**

Permite exibir diferentes conjuntos de informação para controles relacionados.

- **MultiPage**

Este controlo permite criar diversos userforms divididos por separadores. Cada separador tem o seu próprio layout.

- **ScrollBar**

Este controlo é pouco usado uma vez que ComboBox e ListBox vêm com ScrollBar.

Descrição dos controlos

- **SpinButton**

Este controlo consiste em duas setas (verticais ou horizontais) que permitem (por exemplo) incrementar ou decrementar valores numa TextBox.

- **Image**

Permite inserir uma imagem no userform.

UserForms

- *Show/Hide* (mostra/esconde userform)

```
Sub iniciar()
```

```
form1.Show 'mostra a form1
```

```
End Sub
```

```
Sub terminar()
```

```
form1.Hide 'esconde form1
```

```
End Sub
```


Eventos

É algo que acontece a um objecto;

- Para que um objecto possa reagir a determinado evento deverá existir, previamente programado, um procedimento especial designado por *event handler*, que especifica estas reacções.

Eventos

- **Principais eventos**

Click – Quando clica sobre um objecto

Dblclick – Quando clica duas vezes sobre um objecto

Activate – Quando um objecto é activado

GotFocus – Quando um objecto fica com o cursor

KeyPress – Quando se pressiona uma tecla

MouseMove – Quando se move o ponteiro do rato