Components

Documenting Components

Objective

Some times is usual that some calculus are implemented as basic code. Mathcad is a usefull tool to translate and analyze the behavior of these calculations. The objective of this worksheets is show how to implement a simple but traceable component for use basic language inside Mathcad.

Components

The component (or control) of our interest, is the TextBox component. You can find the help about controls under QuickSheets/Programming.

In the insert menu there are the controls provided within Mathcad Software, as is showing in the figure 1.

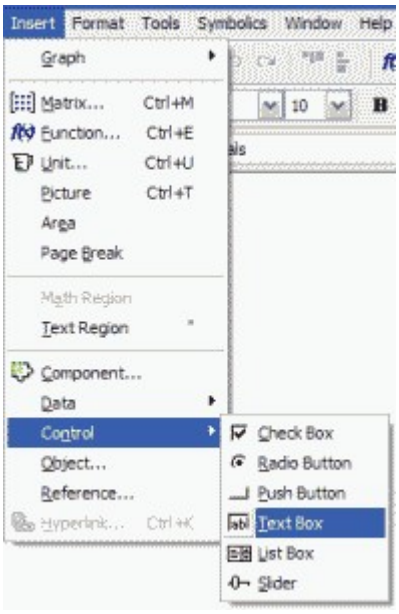


Fig. 1. Inserting a Control.

With the contextual menu (figure 2), we can access to the Component Properties (figure 2) and the script code

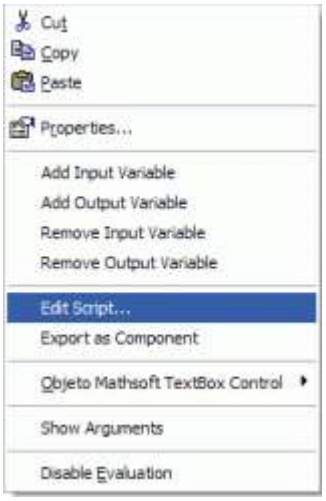


Fig. 2. Contextual Menu

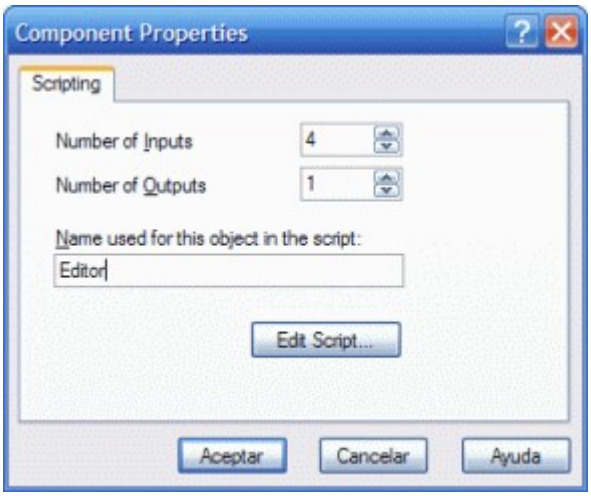


Fig. 3. Seting up Control Properties

Example

For example, this component

vbs(a, b, c, d) :=

30

(a b c d)

do this computation:

$$f(a,b,c,d) := a + 2\cdot b + 3\cdot c + 4\cdot d$$

$$f(1,2,3,4) = 30 \qquad vbs(1,2,3,4) = 30$$

The component internal code

This schema is usefull for developing purposes, but the final work looks *undocumented*. To show what code is executed, we insert a TextBox named **Editor** with this internal code

```
Sub EditorEvent_Start()
   On error resume Next
   ExecuteGlobal CStr(Editor.text)
End Sub

Sub EditorEvent_Change(Inputs,Outputs)
   Editor.Recalculate()
End Sub

Sub EditorEvent_Stop()
  Rem TODO: Add your code here
End Sub
```

Whith the `ExecuteGlobal` statement we can uses the control text as the program. Here is the result

The documented component

vbs(a, b, c, d) :=

```
Sub EditorEvent_Exec(Inputs,Outputs)
   A = Inputs(0).Value
   B = Inputs(1).Value
   C = Inputs(2).Value
   D = Inputs(3).Value
   ans = 1*A+2*B+3*C+4*D
   Outputs(0).Value = ans
End Sub
```

(a b c d)

vbs(1, 2, 3, 4) = 30

Application Example

A vbscript Runge-Kutta Solver

This example is Runge-Kutta solver writing in vbscript. Notice that the code inside the Class declaration generates an error in the Sub `EditorEvent_Start`, because the class name can't be repeated. Then, if you modify this part of the code is needed to 'modify' also the internal script (selecting Edit Script from contextual menu, and exiting) for telling Mathcad that something is happend.

RK(x) :=

```
Sub EditorEvent_Exec(Inputs,Outputs)
   args = Inputs(0).Value
   Set rk4 = New RungeKutta4
   rk4.y1 = args(0) : rk4.x1 = args(1)
   rk4.x2 = args(2) : rk4.n = args(3)
   rk4.ODE = args(4)
   Outputs(0).Value = rk4.solve
End Sub

Class RungeKutta4
Dim m_y1, m_x1, m_x2, m_h, m_n
Property Let y1(val) : m_y1 = val : End Property
Property Let x1(val) : m_x1 = val : End Property
Property Let x2(val) : m_x2 = val : End Property
Property Let n(val)
   m_n = val
   m_h = (m_x2 - m_x1)/val
End property
Property Let ODE(equation)
   ExecuteGlobal( Join( Array("function D(x,y)", _
      "D = CDbl( "  equation _" )",  _
      "End function"), vbLf ) )
End property
Function solve
   Dim rk(), k(4), i
   ReDim rk(m_n,2)
   x = m_x1 : y = m_y1
   rk(0,0) = x : rk(0,1) = y
   For i = 1 to m_n
      k(1) = D(x, y)
      k(2) = D(x + .5*m_h, y + .5*m_h*k(1))
      k(3) = D(x + .5*m_h, y + .5*m_h*k(2))
      k(4) = D(x + m_h, y + m_h*k(3))
      y = y + m_h/6*(k(1) + 2*k(2) + 2*k(3) + k(4))
      x = m_x1 + (m_x2 - m_x1)/m_n*i
      rk(i,0) = x : rk(i,1) = y
   Next
   solve = rk
End Function
End Class
```

x

Parameters

ODE, y' = d(x,y)

$d\$:= "\sin(x) + 1/y"$ script notation

$d(x, y) := \sin(x) + \frac{1}{y}$ mathcad notation

Initial Conditions $x_1 := 0$ $y_1 := 2$

Endpoint and Intervals $x_2 := 25$ $n := 100$

Solutions

Odesolve solution

Given $y'(x) = d(x, y(x))$ $y(x_1) = y_1$

$f := \text{Odesolve}(x, x_2)$

Script solution

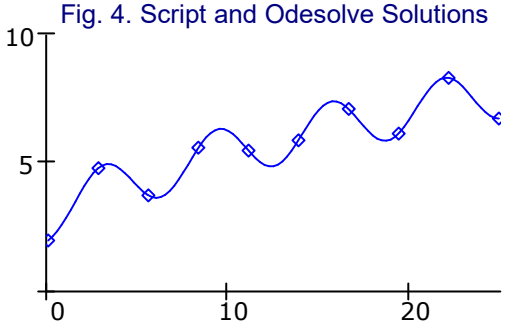
$F := \text{RK}(\text{stack}(y_1, x_1, x_2, n, d\$))$

Comparison of results

Plots

$x := x_1, x_1 + \frac{x_2 - x_1}{9} .. x_2$

Fig. 4. Script and Odesolve Solutions



Note

- Inside the TextBox Copy and Paste are called with Ctrl-Shift-C and Ctrl-Shift-V respectively.

References

- Elementary Differential Equations and Boundary Value Problems, Boyce, DiPrima. Willey, ISBN: 978-0-471-43338-5
- Microsoft Corporation (1997) Visual Basic 5, Component Tools Guide
- Mathcad Documentation

Alvaro Díaz, mathcalc@msn.com