Fundamentos de Informática

Examen – 9 de Junio de 2008

Resolución

1. *(1 punto)*

Num. Base 2 (8 bits)	–Num base 2 (8 bits)	Num. base 10	Num. base 16
00010001	11101111	17	**11**
01100101	10011011	**101**	65
00000101	11111011	5	5

$\mathbf{11}_{16} = 1\cdot16^{1} + 1\cdot16^{10} = \mathbf{17}_{\mathbf{10}}$

$\mathbf{11}_{16} = 0001\cdot16^{1} + 0001\cdot16^{10} = \mathbf{00010001}_{\mathbf{2\ con\ 8\ bits}}$

00010001_{2} Complementado a 1 es $11101110_{Ca1}+1 = \mathbf{11101111}_{Ca2}$

```
101 |2
 01 50|2
  1 10 25|2
     0 05 12|2
        1  0 6|2
           0 3|2
             1 1|2
               1 0
```

$\mathbf{101}_{10}= 1100101_{2} = \mathbf{01100101}_{\mathbf{2\ con\ 8\ bits}}$

$01100101_{2\text{ con 8 bits}} = 0110\cdot 16^{1} + 0101\cdot 16^{0} = \mathbf{6}\cdot 16^{1} + \mathbf{5}\cdot 16^{0} = \mathbf{65}_{\mathbf{16}}$

$01100101_{2\text{ con 8 bits}}$ Complementado a 1 es $10011010_{Ca1}+1 = \mathbf{10011011}_{Ca2}$

$\mathbf{00000101}_{2\text{ con 8 bits}} = 1\cdot2^{2} + 1\cdot2^{0} = 4 + 1 = \mathbf{5}_{\mathbf{10}}$

$\mathbf{00000101}_{2\text{ con 8 bits}} = 0000\cdot 16^{1} + 0101\cdot 16^{0} = \mathbf{0}\cdot 16^{1} + \mathbf{5}\cdot 16^{0} = \mathbf{5}_{\mathbf{16}}$

$\mathbf{00000101}_{2\text{ con 8 bits}}$ Complementado a 1 es $11111010_{Ca1}+1 = \mathbf{11111011}_{Ca2}$

2. *(1,5 puntos)*

```
Sub cmd1_Click()
  Dim x As Integer, y As Integer
  If Not IsNumeric(txt1.Text) Or Not IsNumeric(txt2.Text) Then
    MsgBox "Los operandos han de ser numéricos"
  Else
    x = Val(txt1.Text)
    y = Val(txt2.Text)
    If x = 0 Or y = 0 Then
      pct1.Print "Nulo"
    ElseIf x > 0 And y < 0 Or x < 0 And y > 0 Then
      pct1.Print "Negativo"
    Else
      pct1.Print "Positivo"
    End If
  End If
End Sub
```

Alternativa (sin ElseIf):

```
    If x = 0 Or y = 0 Then
      pct1.Print "Nulo"
    Else
      If x > 0 And y < 0 Or x < 0 And y > 0 Then
        pct1.Print "Negativo"
      Else
        pct1.Print "Positivo"
      End If
    End If
```

3. *(3 puntos)*

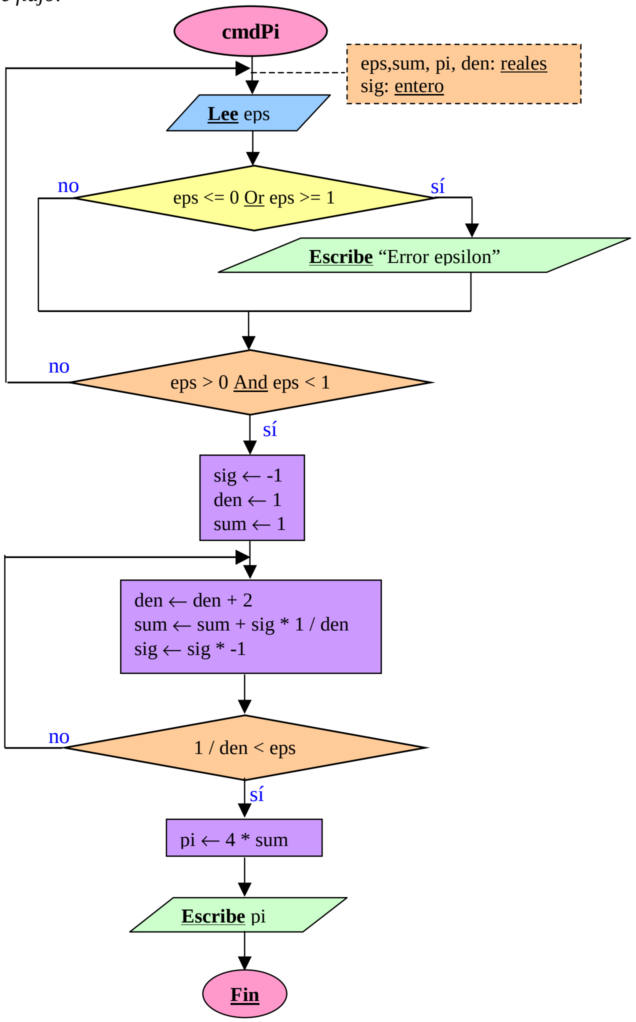
```
Function CompruebaPass(ByVal pass As String) As Integer
  Dim i As Integer, n As Integer
  Dim c As String
  Dim cd As Integer, cl As Integer, co As Integer
  n = Len(pass)
  If n < 4 Or n > 10 Then
    CompruebaPass = 1
  Else
    cd = 0 ' Contador de dígitos
    cl = 0 ' Contador de letras
    co = 0 ' Contador de otros
    For i = 1 To n Step 1
      c = Mid(pass, i, 1)
      If c >= "0" And c <= "9" Then
        cd = cd + 1
      ElseIf c >= "a" And c <= "z" Or c >= "A" And c <= "Z" Then
        cl = cl + 1
      Else 'i.e. Not (c>="0" And c<="9" Or c>="a" And c<="z" Or c>="A" And c<="Z")
        co = co + 1
      End If
    Next i
    If cd = 0 Then
      CompruebaPass = 2
    ElseIf cl = 0 Then
      CompruebaPass = 3
    ElseIf co = 0 Then
      CompruebaPass = 4
    Else
      CompruebaPass = 0
    End If
  End If
End Function
```

Alternativa (con variables lógicas en vez de contadores y primera condicional negada):

```
Function CompruebaPass(ByVal pass As String) As Integer
  Dim i As Integer, n As Integer
  Dim c As String
  Dim dig As Boolean, ltr As Boolean, otr As Boolean
  n = Len(pass)
  If n >= 4 And n <= 10 Then
    dig = False 'Hay al menos un dígito
    ltr = False 'Hay al menos una letra
    otr = False 'Hay al menos un carácter que no es ni dígito ni letra
    For i = 1 To n Step 1
      c = Mid(pass, i, 1)
      If c >= "0" And c <= "9" Then
        dig = True
      ElseIf c >= "a" And c <= "z" Or c >= "A" And c <= "Z" Then
        ltr = True
      Else 'i.e. Not (c>="0" And c<="9" Or c>="a" And c<="z" Or c>="A" And c<="Z")
        otr = True
      End If
    Next i
    If Not dig Then
      CompruebaPass = 2
    ElseIf Not ltr Then
      CompruebaPass = 3
    ElseIf Not otr Then
      CompruebaPass = 4
    Else
      CompruebaPass = 0
  Else
    CompruebaPass = 1
  End If
End Function
```

Alternativa propuesta (no resuelta): prueba a definir (1) una función para contar letras y (2) otra función para contar dígitos. Si hay al menos una letra y un dígito la suma de dígitos y letras no debe dar la longitud completa para que haya algún carácter que no es ni letra ni dígito.

4. *(3,5 puntos)*
Diagrama de flujo:



Programa VB:

```
Sub cmdPi_Click()
  Dim eps As Double 'Epsilon
  Dim pi As Double
  Dim sum As Double 'Sumatorio
  Dim den As Double 'Denominador
  Dim sig As Integer ' Signo
  Do
    eps = InputBox("Introduce epsilon")
    If eps <= 0 Or eps >= 1 Then
      MsgBox "El valor introducido no está en (0, 1)"
    End If
  Loop Until eps > 0 And eps < 1
  sig = -1
  den = 1
  sum = 1
  Do
    den = den + 2
    sum = sum + sig * 1 / den
    sig = sig * -1
  Loop Until 1 / den < eps
  pi = 4 * sum
  MsgBox pi
End Sub
```

Alternativa (puntos suspensivos para código repetido – no se proporciona DdF):

```
Sub cmdPi_Click()
  Dim eps As Double 'Epsilon
  Dim pi As Double
  Dim sum As Double 'Sumatorio
  Dim ter As Double 'Término
  Dim i As Integer ' Contador
…  'Comprobación de epsilon idéntica
  sum = 1
  i = 1
  Do
    i = i + 1
    ter = 1 / (2 * i – 1)
    If i Mod 2 = 0 Then
      sum = sum - ter
    Else
      sum = sum + ter
    End If
  Loop Until ter < eps
  pi = 4 * sum
  MsgBox pi
End Sub
```

5. *(1 punto)*

```
Function Desviacion(ByVal n As Integer, v() As Double, ByVal m As Double) _
                   As Double
  Dim i As Integer
  Dim s As Double
  s = 0
  For i = 1 To n Step 1
    s = s + (v(i) - m) ^ 2
  Next i
  Desviacion = Sqr(s / (n - 1))
End Function
```