



aprenderaprogramar.com

Abstracción, código y verificación del programa ejercicio algoritmo de ordenación resuelto con Visual Basic. (CU00358A)

Sección: Cursos

Categoría: Curso Visual Basic Nivel I

Fecha revisión: 2029

Autor: Mario R. Rancel

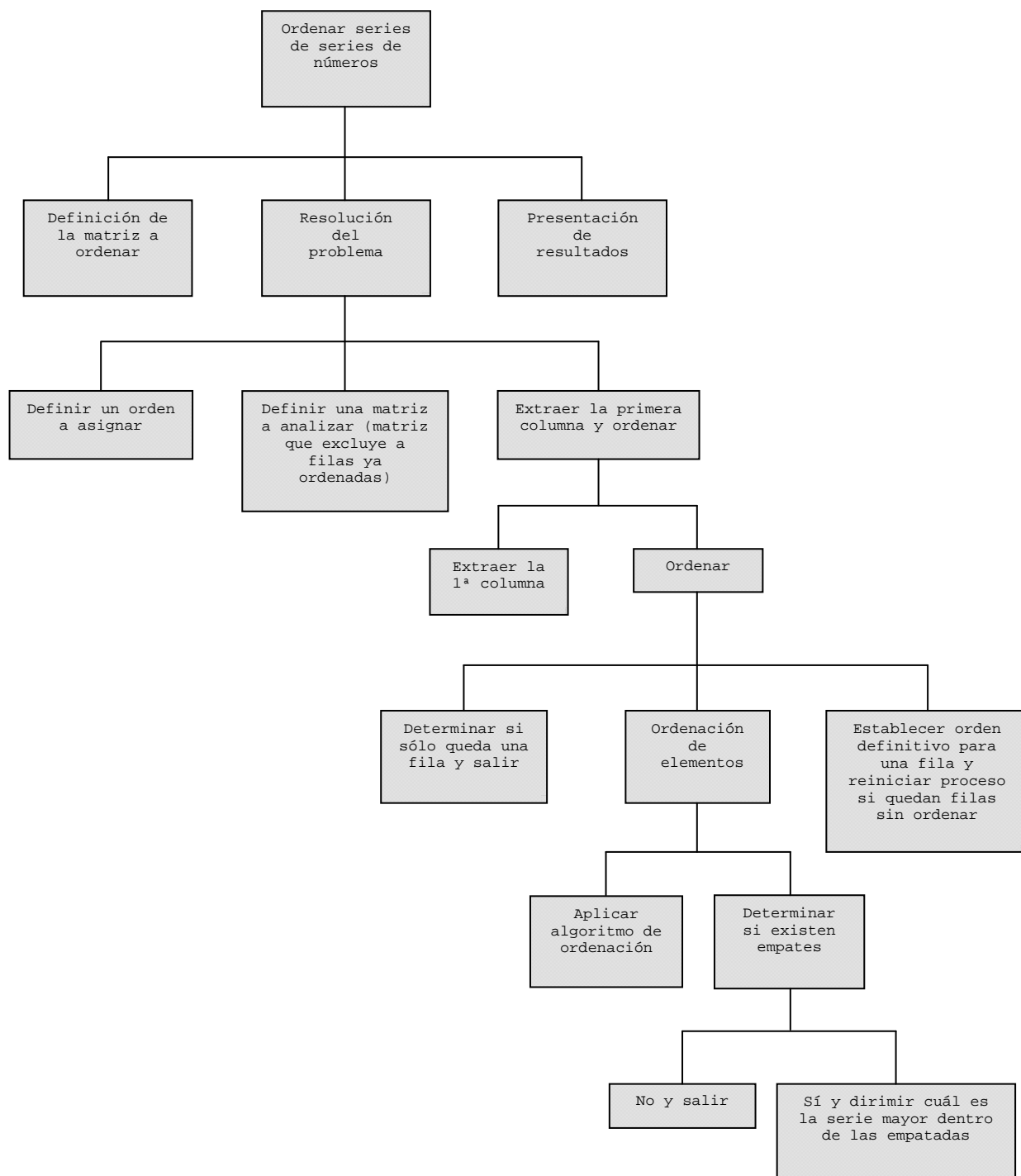
Resumen: Entrega nº57 del Curso Visual Basic Nivel I

29

Ejercicio complementario y opcional para el curso Programación en Visual Basic. Nivel I.

PLANTEAR EL PROCEDIMIENTO DE RESOLUCIÓN. ABSTRACCIÓN. ESQUEMAS DESCENDENTES.

En base a nuestra experiencia como programadores hemos hecho un planteamiento de casos y aplicado una metodología que estimamos se aproxima a un proceso tipo algoritmo, de momento con un lenguaje que no es pseudocódigo. Llega el momento de abstraer esos planteamientos concretos hacia un procedimiento genérico. Es un momento por tanto de reflexión respecto a cómo vamos a enfocar el programa. Analizando los casos anteriores podemos plantear el siguiente esquema descendente:



Este esquema representa una forma de enfocar la resolución del problema. Se entiende que desde un primer planteamiento hasta el esquema definitivo habrá una evolución derivada de la reflexión, mejoras del diseño, constatación de fallos, etc.

Ten en cuenta que si has hecho un correcto planteamiento de casos y organización de la estructura del programa, sólo es cuestión de tiempo llegar a un código satisfactorio. En cambio, si no has sido capaz de elegir casos adecuadamente, analizar los procesos unitarios asimilables a pasos para el ordenador, o si aún habiendo hecho lo anterior no eres capaz de aprehender la información de que dispones para conformar una estructura preliminar de programa, alcanzar un código satisfactorio puede ser un camino tortuoso o imposible.

PSEUDOCÓDIGO Y VERIFICACIONES PARCIALES. CONSTRUCCIÓN DE ABAJO A ARRIBA.

Llega el momento de plantear en pseudocódigo las expresiones o ideas abstractas que hemos ido desarrollando. No vamos a realizar el desarrollo de esta materia ya que resulta un proceso largo (consulta el curso Bases de la programación nivel I de aprenderaprogramar.com para más detalles). Se trata de que el programador plasme una estructura del programa y unos procesos en pseudocódigo para posteriormente ponerse delante del ordenador y de un lenguaje concretos.

¿Cómo empezar? Recuerda siempre que estamos trabajando con la metodología de "dividir y vencer". Por tanto, una buena forma de empezar es no mirar al problema en su conjunto, sino enfrentarnos a partes de él de forma independiente. Si lo estimamos necesario podemos plantear pseudocódigo para programas particulares y a partir de ellos construir un programa genérico.

Del esquema descendente anterior podemos sacar ideas de cuáles pueden ser partes independientes del programa (módulos), como definición de la matriz a ordenar, un algoritmo de ordenación para analizar las columnas que extraigamos, un proceso que permita eliminar empates, etc.

Cada programador puede llegar a distintas formas de solucionar el problema, existiendo métodos variantes por la forma de plantear y agrupar procesos, e incluso métodos distintos de mayor o menor eficiencia que otros.

CÓDIGO Y VERIFICACIONES PARA LA INTEGRACIÓN TOTAL Y ABSTRACCIÓN TOTAL.

Después de atacar el problema por partes que hemos considerado independientes y quizás con cierto grado de particularidad, tendremos que acometer la integración de las distintas partes en un todo, la generalización de las particularidades y la prueba del conjunto. El proceso puede ser relativamente directo o requerir aún de un trabajo importante.

Es de especial relevancia realizar una correcta verificación del programa que hayamos creado hasta constatar su total solidez. Si bien el número de casos utilizados en la búsqueda del método de resolución es limitado, el uso del ordenador nos permitirá hacer una verificación tan exhaustiva como deseemos.

Por motivos de espacio nos limitamos a ofrecer un código que podría ser el resultado al que llegamos después de recorrer el proceso de construcción que hemos omitido. Ten en cuenta que se utilizan alguna herramientas que no hemos abordado en el curso.

CÓDIGO DEL PROGRAMA ORNUMELP PARA VERSIONES MENOS RECIENTES DE VISUAL BASIC

'PROGRAMA ORNUMELP versión 0.1 aprenderaprogramar.com uso exclusivamente didáctico. Prohibido su uso comercial.

Option Explicit

Option Base 1

Dim a%, i%, j%, k%, t%, m%, Empates%

Dim TPO As Variant

Dim final As Boolean

'Fijamos el tamaño de la matriz declarando estas constantes

'Admite valores iguales o superiores a 2 (con una sola fila no hay orden que establecer)

Const numfilasmatoriginal = 4

Const numcolumnsmatoriginal = 4

Dim elementooriginal(1 To numfilasmatoriginal, 1 To numcolumnsmatoriginal)

Dim ordenaasignar%

Private Type element

valor As Single

serieaquepertenece As Integer

End Type

Dim elementoanalizar() As element

Dim ordenencolumna() As element

Dim elementomatrizordenada() As element

Dim quématriz() As element

Dim subelemento() As element

Dim elementotemporal() As element

Private Sub Command1_Click()

'Nos permite ordenar la generación de una nueva matriz y su ordenamiento

final = False

ordenaasignar = 0

Label1 = " "

Form_Load

End Sub

Private Sub Form_Load()

Form1.Caption = "Ordenar series de números"

Asignarvaloresmatrizoriginal

Definirordenaasignar

Definirmatrizaanalizar

'Paso a extraer columna 1 y ordenar, indicándole qué matriz usar

ReDim quématriz(UBound(elementoanalizar, 1), UBound(elementoanalizar, 2))

For i = 1 To UBound(elementoanalizar, 1)

For j = 1 To UBound(elementoanalizar, 2)

 quématriz(i, j).valor = elementoanalizar(i, j).valor

 quématriz(i, j).serieaquepertenece = elementoanalizar(i, j).serieaquepertenece

Next j

Next i

Extraercolumna1yordenar

End Sub

Private Sub Asignarvaloresmatrizoriginal()

Show

'Asignación de valores directa por el programador, desactivada

' elementooriginal(1, 1) = 73 elementooriginal(1, 2) = 18 elementooriginal(1, 3) = 21 elementooriginal(1, 4) = 43

' elementooriginal(2, 1) = 85 elementooriginal(2, 2) = 91 elementooriginal(2, 3) = 17 elementooriginal(2, 4) = 12

' elementooriginal(3, 1) = 73 elementooriginal(3, 2) = 18 elementooriginal(3, 3) = 21 elementooriginal(3, 4) = 43

' elementooriginal(4, 1) = 66 elementooriginal(4, 2) = 12 elementooriginal(4, 3) = 17 elementooriginal(4, 4) = 13

'For i = 1 To numfilasmatoriginal

' For j = 1 To numcolumnsmatoriginal

' Label1 = Label1 & " " & elementooriginal(i, j)

' Next j

' Label1 = Label1 & Chr(13)

'Next i

ReDim elementomatrizordenada(numfilasmatoriginal, numcolumnsmatoriginal)

'Generación de una matriz de valores aleatorios; controlamos el número de empates

'con el múltiplo del RND; ACTIVADO

Label1 = "La matriz a ordenar es esta: " & Chr(13) & Chr(13)

For i = 1 To numfilasmatoriginal

For j = 1 To numcolumnsmatoriginal

```

Randomize
elementooriginal(i, j) = Int(Rnd * 9) + 1
Label1 = Label1 & elementooriginal(i, j) & " "
Next j
Label1 = Label1 & Chr(13)
Next i
End Sub

Private Sub Definirordenaasignar()
ordenaasignar = ordenaasignar + 1
End Sub

Private Sub Definirmatrizaanalizar()
Show
If ordenaasignar > numfilasmatoriginal Then presentaresultados: Exit Sub
If ordenaasignar = 1 Then
ReDim elementoaaanalizar(numfilasmatoriginal, numcolumnasmatoriginal)
For i = 1 To numfilasmatoriginal
For j = 1 To numcolumnasmatoriginal
elementoaaanalizar(i, j).valor = elementooriginal(i, j)
elementoaaanalizar(i, j).serieaquepertenece = i
Next j
Next i
End If
If ordenaasignar > 1 Then
ReDim elementotemporal(UBound(elementoaaanalizar, 1) - 1, UBound(elementoaaanalizar, 2))
a = 0
For i = 1 To UBound(elementoaaanalizar, 1)
'este if se cierra en una línea
If elementoaaanalizar(i, 1).serieaquepertenece = ordenencolumna(1).serieaquepertenece Then GoTo 999
a = a + 1
For j = 1 To UBound(elementoaaanalizar, 2)
elementotemporal(a, j).valor = elementoaaanalizar(i, j).valor
elementotemporal(a, j).serieaquepertenece = elementoaaanalizar(i, j).serieaquepertenece
Next j
999 Next i
'Ya tenemos construída la matriz temporal
ReDim elementoaaanalizar(UBound(elementoaaanalizar, 1) - 1, UBound(elementoaaanalizar, 2))
'Hemos destruído el contenido de elemento a analizar Asignamos a elemento a analizar el contenido de elemento temporal
For i = 1 To UBound(elementotemporal, 1)
For j = 1 To UBound(elementotemporal, 2)
elementoaaanalizar(i, j).valor = elementotemporal(i, j).valor
elementoaaanalizar(i, j).serieaquepertenece = elementotemporal(i, j).serieaquepertenece
Next j
Next i
End If
End Sub

Private Sub Extraercolumna1yordenar()
If UBound(quématriz, 1) > 1 Then
ReDim ordenencolumna(UBound(quématriz, 1))
Else
final = True: Establecerordendefinitivo: Exit Sub
End If
For a = 1 To UBound(quématriz, 1)
For m = 1 To a
If m = a Then ordenencolumna(m).valor = quématriz(a, 1).valor: _
ordenencolumna(m).serieaquepertenece = quématriz(a, 1).serieaquepertenece: Exit For
If quématriz(a, 1).valor > ordenencolumna(m).valor Then Desplazar: Exit For
Next m
Next a
'Proceso para comprobaciones. Desactivado.
For i = 1 To UBound(ordenencolumna, 1)
'Print "ordenencolumna"; i; "="; ordenencolumna(i).valor; "con serie"; ordenencolumna(i).serieaquepertenece
Next i
Empates = 1
For i = 1 To UBound(quématriz, 1)
If i = 1 Then
If ordenencolumna(i).valor > ordenencolumna(i + 1).valor Then
Establecerordendefinitivo
Exit For 'Se sale sin empates

```

```

    End If
  End If
  If i < UBound(quématriz, 1) Then
    If ordenencolumna(i).valor > ordenencolumna(i + 1).valor Then Exit For ' Se sale hay empates
  End If
  If i < UBound(quématriz, 1) Then Empates = Empates + 1
Next i
If final = False And Empates > 1 Then Dirimirempate
End Sub

Private Sub Desplazar()
  t = a + 1
  For k = 1 To a - m
    t = t - 1
    ordenencolumna(t).valor = ordenencolumna(t - 1).valor
    ordenencolumna(t).serieaquepertenece = ordenencolumna(t - 1).serieaquepertenece
  Next k
  ordenencolumna(m).valor = quématriz(a, 1).valor
  ordenencolumna(m).serieaquepertenece = quématriz(a, 1).serieaquepertenece
End Sub

Private Sub Dirimirempate()
  'Entra en dirimir empate con número de empates definido por la variable Empates
  'Caso de que haya iterado, sólo queda una columna y dos o más series de cabeza son absolutamente iguales
  If UBound(quématriz, 2) = 1 And Empates > 1 Then Establecerordendefinitivo: Exit Sub
  ReDim subelemento(Empates, UBound(quématriz, 2) - 1)
  For m = 1 To Empates
    For i = 1 To UBound(quématriz, 1)
      For j = 2 To UBound(quématriz, 2)
        If ordenencolumna(m).serieaquepertenece = quématriz(i, j).serieaquepertenece Then
          subelemento(m, j - 1).valor = quématriz(i, j).valor
          subelemento(m, j - 1).serieaquepertenece = quématriz(i, j).serieaquepertenece
        End If
      Next j
    Next i
  Next m
  'asigna a quématriz el contenido de subelemento para mandarlo a pasar extraercolumna1yordenar
  ReDim quématriz(UBound(subelemento, 1), UBound(subelemento, 2))
  For i = 1 To UBound(subelemento, 1)
    For j = 1 To UBound(subelemento, 2)
      quématriz(i, j).valor = subelemento(i, j).valor
      quématriz(i, j).serieaquepertenece = subelemento(i, j).serieaquepertenece
    Next j
  Next i
  Extraercolumna1yordenar
End Sub

Private Sub Establecerordendefinitivo()
  'Entramos en establecer orden definitivo con orden a asignar definido por la variable ordenaasignar
  a = 0
  For i = 1 To UBound(elementoaaanalizar, 1)
    For j = 1 To UBound(elementoaaanalizar, 2)
      If elementoaaanalizar(i, j).serieaquepertenece = ordenencolumna(1).serieaquepertenece Then
        a = a + 1
        elementomatrizordenada(ordenaasignar, a).valor = elementoaaanalizar(i, j).valor
        elementomatrizordenada(ordenaasignar, a).serieaquepertenece = _
          elementoaaanalizar(i, j).serieaquepertenece
        'Control desactivado Print "elemento mat ordenada"; ordenaasignar; ","; a; "vale"; _
          elementomatrizordenada(ordenaasignar, a).valor; "y pertenece a la serie"; _
          elementomatrizordenada(ordenaasignar, a).serieaquepertenece
      End If
    Next j
  Next i
  If final = True Then
    For i = 1 To UBound(quématriz, 2)
      elementomatrizordenada(ordenaasignar, i).valor = quématriz(1, i).valor
      elementomatrizordenada(ordenaasignar, i).serieaquepertenece = quématriz(1, i).serieaquepertenece
    Next i
  End If
  Definirordenaasignar
  Definirmatrizaanalizar

```

```
'Asignamos a quématriz el contenido de elementoaaanalizar para mandarlo a extraer columna 1 y ordenar
ReDim quématriz(UBound(elementoaaanalizar, 1), UBound(elementoaaanalizar, 2))
For i = 1 To UBound(elementoaaanalizar, 1)
  For j = 1 To UBound(elementoaaanalizar, 2)
    quématriz(i, j).valor = elementoaaanalizar(i, j).valor
    quématriz(i, j).serieaquepertenece = elementoaaanalizar(i, j).serieaquepertenece
  Next j
Next i
If final = True Then Exit Sub
Extraercolumna1yordenar
End Sub

Private Sub presentaresultados()
'Entramos en presentaresultados
Label2 = "La matriz ordenada es esta:" & Chr(13) & Chr(13)
For i = 1 To numfilasmatoriginal
  For j = 1 To numcolumnasmatoriginal
    TPO = TPO & " " & elementomatrizordenada(i, j).valor
  Next j
  Label2 = Label2 & TPO & Chr(13)
  TPO = ""
Next i
End Sub
```

CÓDIGO DEL PROGRAMA ORNUMELP PARA VERSIONES MÁS RECIENTES DE VISUAL BASIC

```
'PROGRAMA ORNUMELP versión 0.1 aprenderaprogramar.com uso exclusivamente didáctico. Prohibido su uso
comercial.
Option Explicit On
Public Class Form1

  Dim a%, m%, t%
  Dim Empates%
  Dim TPO As Object
  Dim final As Boolean
  'Fijamos el tamaño de la matriz declarando estas constantes
  'Admite valores iguales o superiores a 2 (con una sola fila no hay orden que establecer)
  Const numfilasmatoriginal = 4
  Const numcolumnasmatoriginal = 4
  Dim elementooriginal(numfilasmatoriginal, numcolumnasmatoriginal)
  Dim ordenaasignar%

  Private Structure element
    Dim valor As Single
    Dim serieaquepertenece As Integer
  End Structure

  Dim elementoaaanalizar(,) As element
  Dim ordenen columna() As element
  Dim elementomatrizordenada(,) As element
  Dim quématriz(,) As element
  Dim subelemento(,) As element
  Dim elementotemporal(,) As element

  Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
    'Nos permite ordenar la generación de una nueva matriz y su ordenamiento
    'Nos permite ordenar la generación de una nueva matriz y su ordenamiento
    final = False
    ordenaasignar = 0
    Label1.Text = " "
    Me.Form1_Load(Me, New System.EventArgs)
  End Sub
```

```

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Dim i As Integer, j As Integer
    Me.Text = "Ordenar series de números"
    Asignarvaloresamatrizoriginal()
    Definirordenaasignar()
    Definirmatrizaanalizar()
    'Paso a extraer columna 1 y ordenar, indicándole qué matriz usar
    ReDim quematriz(UBound(elementoaaanalizar, 1), UBound(elementoaaanalizar, 2))
    For i = 1 To UBound(elementoaaanalizar, 1)
        For j = 1 To UBound(elementoaaanalizar, 2)
            quematriz(i, j).valor = elementoaaanalizar(i, j).valor
            quematriz(i, j).serieaquepertenece = elementoaaanalizar(i, j).serieaquepertenece
        Next j
    Next i
    Extraercolumnalyordenar()
End Sub

Private Sub Asignarvaloresamatrizoriginal()
    Dim i As Integer, j As Integer
    'Asignación de valores directa por el programador, ACTIVADA
    elementooriginal(1, 1) = 7 : elementooriginal(1, 2) = 6 : elementooriginal(1, 3) = 4 :
elementooriginal(1, 4) = 8
    elementooriginal(2, 1) = 6 : elementooriginal(2, 2) = 4 : elementooriginal(2, 3) = 7 :
elementooriginal(2, 4) = 9
    elementooriginal(3, 1) = 6 : elementooriginal(3, 2) = 6 : elementooriginal(3, 3) = 3 :
elementooriginal(3, 4) = 8
    elementooriginal(4, 1) = 4 : elementooriginal(4, 2) = 8 : elementooriginal(4, 3) = 1 :
elementooriginal(4, 4) = 3
    For i = 1 To numfilasmatoriginal
        For j = 1 To numcolumnasmatoriginal
            Label1.Text = Label1.Text & " " & elementooriginal(i, j)
        Next j
        Label1.Text = Label1.Text & Chr(13)
    Next i
    ReDim elementomatrizordenada(numfilasmatoriginal, numcolumnasmatoriginal)
    'Generación de una matriz de valores aleatorios; controlamos el número de empates
    'con el múltiplo del RND; DESACTIVADO
    'Label1.Text = "La matriz a ordenar es esta: " & Chr(13) & Chr(13)
    'For i = 1 To numfilasmatoriginal
    'For j = 1 To numcolumnasmatoriginal
    'Randomize()
    'elementooriginal(i, j) = Int(Rnd() * 9) + 1
    'Label1.Text = Label1.Text & elementooriginal(i, j) & " "
    'Next j
    'Label1.Text = Label1.Text & Chr(13)
    'Next i
End Sub

Private Sub Definirordenaasignar()
    ordenaasignar = ordenaasignar + 1
End Sub

Private Sub Definirmatrizaanalizar()
    Dim i As Integer, j As Integer
    If ordenaasignar > numfilasmatoriginal Then presentarResultados() : Exit Sub
    If ordenaasignar = 1 Then
        ReDim elementoaaanalizar(numfilasmatoriginal, numcolumnasmatoriginal)
        For i = 1 To numfilasmatoriginal
            For j = 1 To numcolumnasmatoriginal
                elementoaaanalizar(i, j).valor = elementooriginal(i, j)
                elementoaaanalizar(i, j).serieaquepertenece = i
            Next j
        Next i
    End If
    If ordenaasignar > 1 Then
        ReDim elementotemporal(UBound(elementoaaanalizar, 1) - 1, UBound(elementoaaanalizar, 2))
        Me.a = 0
        For i = 1 To UBound(elementoaaanalizar, 1)
            'este if se cierra en una línea
            If elementoaaanalizar(i, 1).serieaquepertenece = ordenencolumna(1).serieaquepertenece Then
                Me.a = Me.a + 1
                For j = 1 To UBound(elementoaaanalizar, 2)

```



```

                elementotemporal(Me.a, j).valor = elementoanalizar(i, j).valor
                elementotemporal(Me.a, j).serieaquepertenece = elementoanalizar(i,
j).serieaquepertenece
                Next j
999:           Next i
                'Ya tenemos construída la matriz temporal
                ReDim elementoanalizar(UBound(elementoanalizar, 1) - 1, UBound(elementoanalizar, 2))
                'Hemos destruído el contenido de elemento a analizar Asignamos a elemento a analizar el
contenido de elemento temporal
                For i = 1 To UBound(elementotemporal, 1)
                    For j = 1 To UBound(elementotemporal, 2)
                        elementoanalizar(i, j).valor = elementotemporal(i, j).valor
                        elementoanalizar(i, j).serieaquepertenece = elementotemporal(i,
j).serieaquepertenece
                    Next j
                Next i
            End If
        End Sub

        Private Sub Extraercolumnalyordenar()
            Dim i As Integer
            If UBound(quématriz, 1) > 1 Then
                ReDim ordenencolumna(UBound(quématriz, 1))
            Else
                final = True : Establecerordendefinitivo() : Exit Sub
            End If
            For Me.a = 1 To UBound(quématriz, 1)
                For Me.m = 1 To Me.a
                    If Me.m = Me.a Then ordenencolumna(Me.m).valor = quématriz(Me.a, 1).valor _
                        : ordenencolumna(Me.m).serieaquepertenece = quématriz(Me.a, 1).serieaquepertenece :
Exit For
                    If quématriz(Me.a, 1).valor > ordenencolumna(Me.m).valor Then Desplazar() : Exit For
                Next Me.m
            Next Me.a
            'Proceso para comprobaciones. Desactivado.
            'For i = 1 To UBound(ordenencolumna, 1)
            'Print "ordenencolumna"; i; "="; ordenencolumna(i).valor; "con serie";
ordenencolumna(i).serieaquepertenece
            'Next i
            Empates = 1
            For i = 1 To UBound(quématriz, 1)
                If i = 1 Then
                    If ordenencolumna(i).valor > ordenencolumna(i + 1).valor Then
                        Establecerordendefinitivo()
                        Exit For 'Se sale sin empates
                    End If
                End If
                If i < UBound(quématriz, 1) Then
                    If ordenencolumna(i).valor > ordenencolumna(i + 1).valor Then Exit For ' Se sale hay
empates
                End If
                If i < UBound(quématriz, 1) Then Empates = Empates + 1
            Next i
            If final = False And Empates > 1 Then Dirimirempate()
        End Sub

        Private Sub Desplazar()
            Dim k As Integer
            Me.t = Me.a + 1
            For k = 1 To Me.a - Me.m
                Me.t = Me.t - 1
                ordenencolumna(Me.t).valor = ordenencolumna(Me.t - 1).valor
                ordenencolumna(Me.t).serieaquepertenece = ordenencolumna(Me.t - 1).serieaquepertenece
            Next k
            ordenencolumna(Me.m).valor = quématriz(Me.a, 1).valor
            ordenencolumna(Me.m).serieaquepertenece = quématriz(Me.a, 1).serieaquepertenece
        End Sub

        Private Sub Dirimirempate()
            Dim i As Integer, j As Integer
            'Entra en dirimir empate con número de empates definido por la variable Empates
            'Caso de que haya iterado, sólo queda una columna y dos o más series de cabeza son absolutamente
iguales
            If UBound(quématriz, 2) = 1 And Empates > 1 Then Establecerordendefinitivo() : Exit Sub

```

```

ReDim subelemento(Empates, UBound(quématriz, 2) - 1)
For Me.m = 1 To Empates
  For i = 1 To UBound(quématriz, 1)
    For j = 2 To UBound(quématriz, 2)
      If ordenecolumna(Me.m).serieaquepertenece = quématriz(i, j).serieaquepertenece Then
        subelemento(Me.m, j - 1).valor = quématriz(i, j).valor
        subelemento(Me.m, j - 1).serieaquepertenece = quématriz(i, j).serieaquepertenece
      End If
    Next j
  Next i
Next Me.m
'asigna a quématriz el contenido de subelemento para mandarlo a pasar extraercolumnayordenar
ReDim quématriz(UBound(subelemento, 1), UBound(subelemento, 2))
For i = 1 To UBound(subelemento, 1)
  For j = 1 To UBound(subelemento, 2)
    quématriz(i, j).valor = subelemento(i, j).valor
    quématriz(i, j).serieaquepertenece = subelemento(i, j).serieaquepertenece
  Next j
Next i
Extraercolumnayordenar()
End Sub

Private Sub Establecerordendefinitivo()
  Dim i As Integer, j As Integer
  'Entramos en establecer orden definitivo con orden a asignar definido por la variable
ordenaasignar
  Me.a = 0
  For i = 1 To UBound(elementoaaanalizar, 1)
    For j = 1 To UBound(elementoaaanalizar, 2)
      If elementoaaanalizar(i, j).serieaquepertenece = ordenecolumna(1).serieaquepertenece Then
        Me.a = Me.a + 1
        elementomatrizordenada(ordenaasignar, Me.a).valor = elementoaaanalizar(i, j).valor
        elementomatrizordenada(ordenaasignar, Me.a).serieaquepertenece = _
          elementoaaanalizar(i, j).serieaquepertenece
      End If
    Next j
  Next i
  If final = True Then
    For i = 1 To UBound(quématriz, 2)
      elementomatrizordenada(ordenaasignar, i).valor = quématriz(1, i).valor
      elementomatrizordenada(ordenaasignar, i).serieaquepertenece = quématriz(1,
i).serieaquepertenece
    Next i
  End If
  Definirordenaasignar()
  Definirmatrizaanalizar()
  'Asignamos a quématriz el contenido de elementoaaanalizar para mandarlo a extraer columna 1 y ordenar
ReDim quématriz(UBound(elementoaaanalizar, 1), UBound(elementoaaanalizar, 2))
For i = 1 To UBound(elementoaaanalizar, 1)
  For j = 1 To UBound(elementoaaanalizar, 2)
    quématriz(i, j).valor = elementoaaanalizar(i, j).valor
    quématriz(i, j).serieaquepertenece = elementoaaanalizar(i, j).serieaquepertenece
  Next j
Next i
If final = True Then Exit Sub
Extraercolumnayordenar()
End Sub

Private Sub presentaresultados()
  Dim i As Integer
  Dim j As Integer
  'Entramos en presentaresultados
Label2.Text = "La matriz ordenada es esta:" & Chr(13) & Chr(13)
For i = 1 To numfilasmatoriginal
  For j = 1 To numcolumnasmatoriginal
    TPO = TPO & " " & elementomatrizordenada(i, j).valor
  Next j
  Label2.Text = Label2.Text & TPO & Chr(13)
  TPO = ""
Next i
End Sub
End Class

```

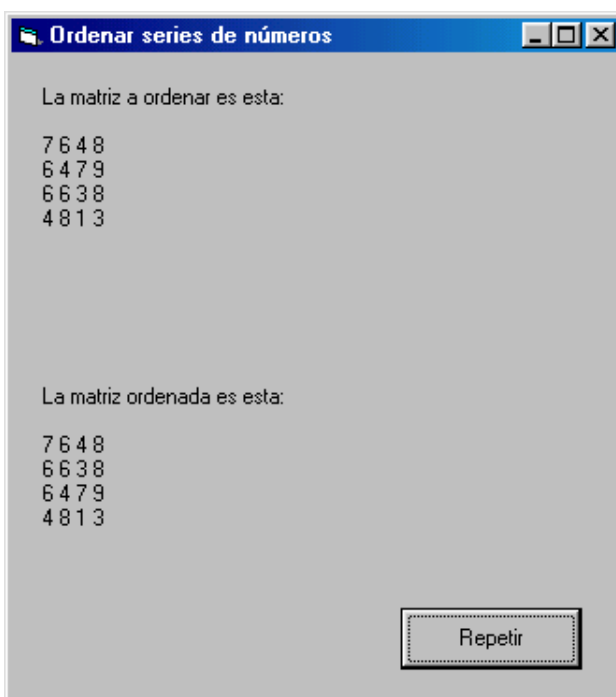
Comentarios: El programador controla la dimensión de la matriz a ordenar a través de la declaración `Const numfilasmatoriginal = 4` y `Const numcolumnasmatoriginal = 4`.

Por otro lado, se controla el intervalo de valores para la generación de datos aleatorios a través de la expresión.

$$\text{elementooriginal}(i, j) = \text{Int}(\text{Rnd} * 9) + 1$$

Esta expresión constituye una forma de regular el número de empates que queremos que aparezcan. Si el número que multiplica a `Rnd` es bajo habrá un mayor número de repeticiones en los elementos de la matriz.

El resultado del programa será de este tipo:



Próxima entrega: CU00359A

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:
http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=37&Itemid=61