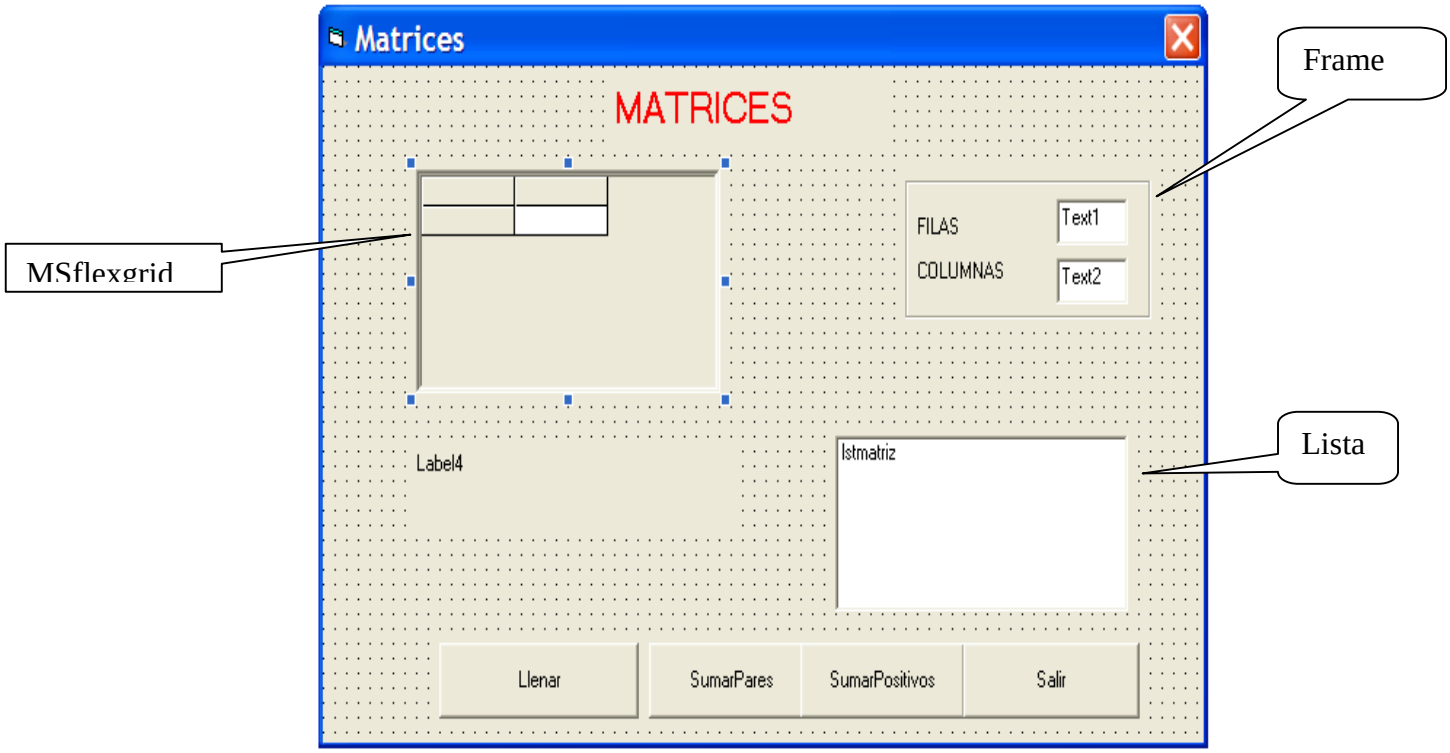
Materia: Algoritmos Avanzados
Tema: Matrices – Estructuras de Datos

La practica podrá ser resuelta en grupos de dos personas como máximo, pero la defebsa es individual. Toda copia será sancionada con la anulación de la nota de practicas (10 pts)
La aplicación 3 será resuelta en laboratorio el día viernes 14 de septiembre y tiene una ponderación de 1 punto. La aplicación 4 se deja como tarea y será entregada en laboratorio el día viernes 21 de septiembre

---

APLICACIÓN 3

Diseñe el siguiente formulario



Para sacar un control MSFLEXGrid

- Clic derecho sobre la barra de herramientas
- Componentes
- Microsoft FlexGrid Control 6.0
- Aplicar
- Cerrar

Los controles deben tener los siguientes nombres:

control	name	control	Name
Label1	Lbltitulo	Command2	CmdSumaPar
Label2	Tblmatriz	Command3	CmdSumaPos

List1	LstMatriz	Command4	CmdSalir
Command1	CmdLlenar	Msflexgrid1	MSFMatriz

Luego haga doble click sobre el formulario y escriba el siguiente código

```
Private Sub Form_Load()
     MSFmatriz.FixedRows = 0
     MSFmatriz.FixedCols = 0
     bandera = False
End Sub
```

En la ventana de código defina las siguientes variables globales

```
Option explicit
Dim mat(10, 10) As Integer
Dim nf As Integer, nc As Integer
Dim bandera As Boolean
```

Option explicit verifica que todas las variable han sido definidas. A continuación escriba el siguiente código, para llenar la matriz y luego mostrarla en los controles Msflexgrid, label y list:

```
Private Sub llenaMatriz(ByRef m() As Integer, ByVal nn As
Integer, ByVal mm As Integer)
  Dim i As Integer, j As Integer, aux As String
  MSFmatriz.Clear  ' Limpiar el control msfmatriz
  MSFmatriz.Rows = nn
  MSFmatriz.Cols = mm
  For i = 1 To nn
    For j = 1 To mm
      m(i, j) = InputBox("Ingrese elemento  ")
      ' Para mostrar los elementos de la matriz en el MSFLEXGRID
      MSFmatriz.TextMatrix(i - 1, j - 1) = m(i, j)
    Next j
  Next i

  ' Limpiamos los controles LISTA y LABEL
    lstmatriz.Clear
    lblMatriz.Caption = ""

  For i = 1 To nn
    For j = 1 To mm
      lblMatriz.Caption = lblMatriz.Caption & m(i, j) & ", "
    Next j
    lblMatriz.Caption = lblMatriz.Caption & Chr(13)
  Next i

  aux = ""
  For i = 1 To nn
    For j = 1 To mm
      aux = aux & m(i, j) & ", "
    Next j
    lstmatriz.AddItem aux
    aux = ""
  Next i
```

```
End Sub
```

Después de hacer doble click sobre el botón llenar

```
Private Sub cmdLlenar_Click()
nf = Val(txtFilas.Text)
nc = Val(txtColumnas.Text)
bandera = True
llenaMatriz mat, nf, nc
End Sub
```

Función que suma los elementos pares de la matriz

```
Private Function SumaParMatriz(ByRef m() As Integer, ByVal nn As
Integer, ByVal mm As Integer) As Integer
  Dim i As Integer, j As Integer, s As Integer
  For i = 1 To nn
    For j = 1 To mm
      If m(i, j) Mod 2 = 0 Then
        s = s + m(i, j)
       End If
    Next j
  Next i
  SumaParMatriz = s
End Function
```

Después de hacer doble click sobre el botón sumaPares

```
Private Sub cmdSumaPar_Click()
If bandera = True Then
   MsgBox "La suma de los números pares de la matriz es : " &
          SumaParMatriz(mat, nf, nc), vbInformation, "Algoritmos
          Avanzados"
Else
   MsgBox "La matriz no tiene datos " & Chr(13) & "Imposible
          mostrar resultados", vbExclamation, "Mensajillo"
End If
End Sub
```

La función sumar los elementos pares queda como practica

APLICACIÓN 4. Implementar las siguientes operaciones sobre matrices:

a. Buscar un elemento en una matriz

b. ***Generar la matriz caracol***