

GUIA PRACTICA 01

Competencia:

Mediante la elaboración de esta Guía Práctica podrás reforzar tus conocimientos de la estructuras básicas de programación del lenguaje C# , y específicamente con el manejo de matrices y números aleatorios.

Descripción:

La guía se basa en el proyecto Lab03_Estructuras (entregado como recurso en la plataforma virtual). La guía se divide en 2 secciones:

- a) Un laboratorio guiado, paso a paso con la solución de un problema sobre arreglos y generación de números aleatorios.
- b) Un ejercicio propuesto que tendrás que resolver en base a lo que hayas asimilado de lo aprendido en esta semana

Finalmente empaqueta la carpeta con tu aplicativo ya resuelto y envíala mediante la plataforma, hasta la fecha indicada por tu instructor. Esto será tu tarea de esta semana. ¡A trabajar entonces!!

PARTE 1: LABORATORIO GUIADO

El escenario es simular la generación de 100 números elegidos al azar por la propia maquina (llamados números aleatorios) en un rango de 100 a 1000. Los números deben ser almacenados en una matriz y mostrarlos en un control listbox. Luego de ello ingresar un valor numérico y finalmente, de los números almacenados en la matriz, cuántos son mayores, menores e iguales al número ingresado.

Paso 1:

Descarga el empaquetado con el proyecto **Lab03_Estructuras** y ábrelo en el VS 2022. Una vez abierto el proyecto selecciona el formulario **frmArreglos1**

Paso 2:

Accede a la ventana de código y a nivel de formulario efectúa las siguientes declaraciones

```
public partial class frmArreglos1 : Form
{
    // Se declara la instancia objRandom de la clase Random que permite generar numeros aleatorios
    Random objRandom = new Random();
    // y la matriz de numeros enteros
    int[] intNumeros = new int[100];
}
```

Paso 3:

Ahora procederemos a generar los 100 números aleatorios. Tal como tu instructor te explico en la sesión presencial, accede al evento clic del botón **btnGenerar** de tu formulario y codifica lo siguiente:

```
private void btnGenerar_Click(object sender, EventArgs e)
{
    // Generamos los 100 numeros....
    for (short intValor = 0; intValor < 100; intValor++)
    { // entre 100 y 1000, con el metodo Next de la instancia objRandom
        intNumeros[intValor] = objRandom.Next(100, 1001);
    }
    //Mostramos en la lista los valores generados
    lstNumeros.DataSource = null; // se limpia la lista
    lstNumeros.DataSource = intNumeros; // se enlaza la lista a la matriz intNumeros
}
```

Paso 4:

Por último codificaremos el evento clic del botón **btnBuscar** para que se muestren la cantidad de valores mayores, menores e iguales al valor ingresado en la caja de texto **txtBuscar**

```
private void btnBuscar_Click(object sender, EventArgs e)
{
    try
    {
        int intBuscar = Convert.ToInt16(txtBuscar.Text);
        int intMenores = 0;
        int intMayores = 0;
        int intIguales = 0;
        // Empezamos la busqueda
        for (short intZ = 0; intZ < 100; intZ++)
        {
            if (intNumeros[intZ] < intBuscar)
            {
                intMenores += 1;
            }
            else if (intNumeros[intZ] > intBuscar)
            {
                intMayores += 1;
            }
            else
            {
                intIguales += 1;
            }
        }
        // Mostramos Resultados
        lblMenores.Text = intMenores.ToString();
        lblMayores.Text = intMayores.ToString();
        lblIguales.Text = intIguales.ToString();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error....." + ex.Message);
    }
}
```

Paso 5:

Por último en la clase **Program** de su proyecto indique que la ejecución del mismo se hará desde el formulario **frmArreglos1** y compílelo. De haber algún error sintáctico debes corregirlo para luego, una vez subsanado cualquier error ejecutar su proyecto y probar la funcionalidad del ejercicio desarrollado.

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new frmArreglos1());
}
```

PARTE 2: EJERCICIO PROPUESTO

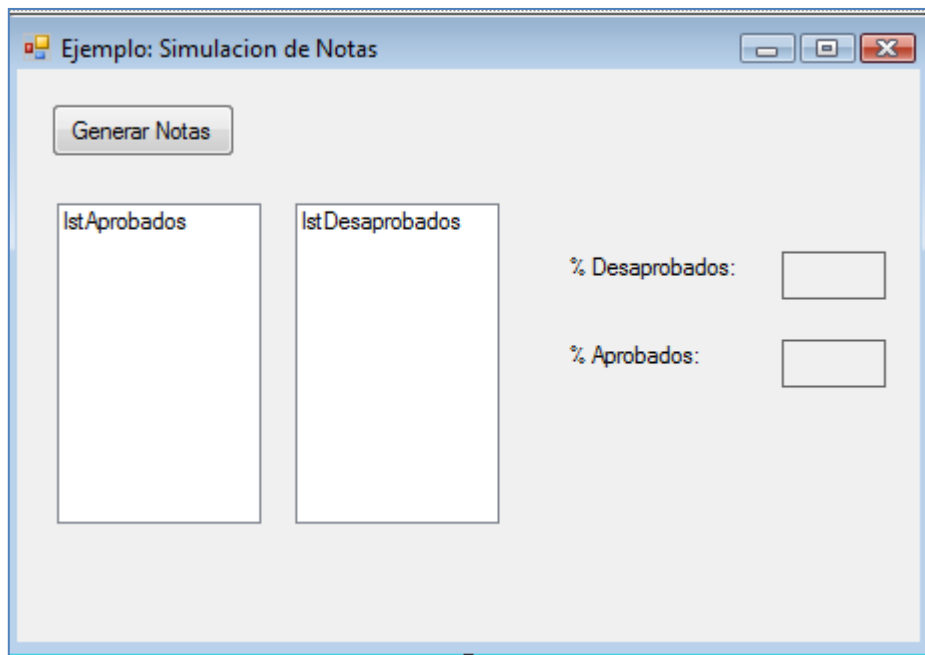
Planteamiento del problema:

En este caso debe resolver el siguiente problema: determinar el porcentaje de aprobados y desaprobados en una simulación de 100 exámenes (con notas entre 0 y 20) generados aleatoriamente al hacer clic en el botón **btnGenerar**, con las siguientes características:

- a) Se sabe que la nota mínima aprobatoria es 13
- b) Todas las notas aprobatorias cárguelas a la lista **lstAprobados** y las desaprobatorias a la lista **lstDesaprobados**
- c) En los label del lado derecho muestra el porcentaje (no la cantidad) de desaprobados y aprobados en el examen simulado.
- d) Cada vez que se haga clic en el botón **btnGenerar** se deben limpiar las listas y label, dando la impresión de que se trata de un nuevo proceso de examen.

Paso 1:

En el proyecto **Lab03_Estructuras** agregue el siguiente formulario, nombrándolo previamente como **frmArreglos2**:



Ejemplo: Simulacion de Notas

Generar Notas

IstAprobados

IstDesaprobados

% Desaprobados:

% Aprobados:

Paso 2:

Codifique el evento clic del botón **btnGenerar** para dar solución al problema planteado.

iii A programar!!!