



Universidad Nacional de Ingeniería

.- Facultad de Electrotecnia y Computación -.

Ingeniería en Computación

Algoritmización y estructuras de datos

Laboratorio No. 5

Alumno:

Marcel Enmanuel Díaz Largaespada (2020-1384U)

Docente:

Adilson González López

Fecha de entrega:

Miércoles 25 de agosto del 2021

Índice

1. Introducción.....	3
2. Diseño del <i>form</i>	4
3. Código Fuente.....	5
3.1. Declaraciones y métodos iniciales.	5
3.2. Métodos de interacción con la interfaz.	7
4. Ejecución del programa	10
5. Conclusiones	14

1. Introducción

Para la práctica de laboratorio número 5, se trabajó finalmente con **estructuras de datos**. Dichas estructuras, de no manejarse correctamente, o de no haberse acostumbrado a su manejo, pueden llegar a ser confusas y poco inteligibles, sobre todo si se trata de un arreglo de estructuras. Sin embargo, no se ha tenido complicaciones de ningún tipo.

A continuación, se brinda una descripción breve de las partes del código y, adjuntas a ellas, hay capturas de pantalla.

2. Diseño del *form*

La interfaz que se trabajó fue algo complicada porque, por primera vez, se trabajaba con una cuantiosa cantidad de elementos en la pantalla. Sin embargo, el IDE lo compensó con la facilidad con la que se añaden dichos elementos y lo sencillo que es modificar sus propiedades.

Al mismo tiempo, se intentó conservar la simetría y homogeneidad entre los elementos.

The image shows a screenshot of a Windows application window titled "Form1". The window contains a data entry form with the following elements:

- A label "Cantidad de elementos a ingresar:" followed by a text input field and an "Establecer" button.
- Three text input fields labeled "Camet:", "Nombre:", and "Apellido:".
- A "Sexo" label followed by a dropdown menu.
- A "Fecha de nacimiento" label followed by three dropdown menus labeled "Dia", "Mes", and "Año".
- Three buttons: "Registrar", "Borrar", and "Actualizar".
- An "Imprimir" button.
- A large text area at the bottom with the label "lbImprimir".

3. Código Fuente

3.1. Declaraciones y métodos iniciales.

Se principia declarando las variables de iteración. Luego, se declaran las estructuras con las que trabajaremos y se inicializa un arreglo de las mismas con un tamaño indefinido, de momento.

```
//Declaraciones globales
int Tam, N, i;

//Declaraciones de estructuras
1 referencia
public struct Fecha
{
    public int Dia;
    public string Mes;
    public int Año;
}
2 referencias
public struct Datos
{
    public string Carnet;
    public string Nombres;
    public string Apellidos;
    public string Sexo;
    public Fecha FechaNac;
}
Datos[] Registro; //Se crea un arreglo de la estructura de Datos
```

Cuando nuestro *form* inicialice, se configurarán las cajas de selección y se le añadirán a cada una, las opciones correspondientes.

```
1 referencia
private void Form1_Load(object sender, EventArgs e)
{
    cbSexo.DropDownStyle = ComboBoxStyle.DropDownList;
    cbMes.DropDownStyle = ComboBoxStyle.DropDownList;
    cbDia.DropDownStyle = ComboBoxStyle.DropDownList;

    cbSexo.Items.Add("M");
    cbSexo.Items.Add("F");

    for (int a = 1; a <= 31; a++)
    {
        cbDia.Items.Add(Convert.ToString(a));
    }

    cbMes.Items.Add("Enero");
    cbMes.Items.Add("Febrero");
    cbMes.Items.Add("Marzo");
    cbMes.Items.Add("Abril");
    cbMes.Items.Add("Mayo");
    cbMes.Items.Add("Junio");
    cbMes.Items.Add("Julio");
    cbMes.Items.Add("Agosto");
    cbMes.Items.Add("Septiembre");
    cbMes.Items.Add("Octubre");
    cbMes.Items.Add("Noviembre");
    cbMes.Items.Add("Diciembre");
}
```

Esta función nos permite limpiar las cajas de texto rellenas:

```
//Funciones generales
2 referencias
void Limpiar()
{
    txtCarnet.Clear();
    txtNombres.Clear();
    txtApellidos.Clear();
    txtAño.Clear();
}
```

3.2. Métodos de interacción con la interfaz.

Botón Establecer: Asigna un tamaño a la estructura de registro determinado por la caja de texto correspondiente:

```
//Boton establecer
1 referencia
private void btnEstablecer_Click(object sender, EventArgs e)
{
    Tam = int.Parse(txtCantidad.Text);
    Registro = new Datos[Tam]; //Se le asigna un tamaño al arreglo de registro
    N = 0;
}
```

Botón Registrar: Rellena los datos del último elemento del arreglo de la estructura de registro con la información correspondiente para cada variable.

```
//Boton Registrar
1 referencia
private void btnRegistrar_Click(object sender, EventArgs e)
{
    if (N <= Tam - 1)
    {
        Registro[N].Carnet = txtCarnet.Text;
        Registro[N].Nombres = txtNombres.Text;
        Registro[N].Apellidos = txtApellidos.Text;
        Registro[N].Sexo = cbSexo.Text;
        Registro[N].FechaNac.Dia = int.Parse(cbDia.Text);
        Registro[N].FechaNac.Año = int.Parse(txtAño.Text);
        Registro[N].FechaNac.Mes = cbMes.Text;
        Limpiar();
        N++;
        MessageBox.Show("El estudiante ha sido ingresado");
    }
    else
    {
        MessageBox.Show("No hay espacio.");
    }
}
```

Botón Eliminar: Busca entre los elementos del arreglo de la estructura de registros por una coincidencia con la información ingresada en la caja de texto del carnet. De no encontrarla, se notifica y, de lo contrario, se procede a eliminar dicho elemento y traslada a los posteriores un elemento menos.

```
//Botón Eliminar
1 referencia
private void btnBorrar_Click(object sender, EventArgs e)
{
    int i = 0;
    string x;
    x = txtCarnet.Text;

    while (i < N && x != Registro[i].Carnet)
    {
        i++;
    }
    if (i >= N)
    {
        MessageBox.Show(x + "no está registrado.");
    }
    else
    {
        for (int k = i; k < N - 1; k++)
        {
            Registro[k].Carnet = Registro[k + 1].Carnet;
            Registro[k].Nombres = Registro[k + 1].Nombres;
            Registro[k].Apellidos = Registro[k + 1].Apellidos;
            Registro[k].Sexo = Registro[k + 1].Sexo;
            Registro[k].FechaNac.Dia = Registro[k + 1].FechaNac.Dia;
            Registro[k].FechaNac.Mes = Registro[k + 1].FechaNac.Mes;
            Registro[k].FechaNac.Año = Registro[k + 1].FechaNac.Año;
        }
        N--;
        MessageBox.Show("El estudiante con carnet " + x + " se ha eliminado.");
    }
}
```


Botón Actualizar: Realiza la misma búsqueda del **botón Eliminar**. Si no encuentra la coincidencia, se notifica; si la encuentra, se reemplaza la información en ese elemento por lo que se encuentra actualmente en las cajas de texto correspondientes a la información del estudiante.

```
1 referencia
private void btnActualizar_Click(object sender, EventArgs e)
{
    int i = 0;
    string x;
    x = txtCarnet.Text;

    while (i < N && x != Registro[i].Carnet)
    {
        i++;
    }
    if (i >= N)
    {
        MessageBox.Show(x + " no está registrado.");
    }
    else
    {
        Registro[i].Nombres = txtNombres.Text;
        Registro[i].Apellidos = txtApellidos.Text;
        Registro[i].Sexo = cbSexo.Text;
        Registro[i].FechaNac.Dia = int.Parse(cbDia.Text);
        Registro[i].FechaNac.Mes = cbMes.Text;
        Registro[i].FechaNac.Año = int.Parse(txtAño.Text);
        Limpiar();
        MessageBox.Show("El estudiante con carnet " + x + " se ha actualizado.");
    }
}
```

Botón Imprimir: Imprime todos los datos de todos los elementos del arreglo de la estructura en orden de adición.

```
1 referencia
private void btnImprimir_Click(object sender, EventArgs e)
{
    lblImprimir.Items.Clear();
    lblImprimir.Items.Add("Carnet" + "\t" + "Nombres" + "\t" + "Apellidos" + "\t" + "Sexo" + "\t" + "Nacimiento");
    for (int i = 0; i < N; i++)
    {
        lblImprimir.Items.Add(Registro[i].Carnet + "\t" + Registro[i].Nombres + "\t" + Registro[i].Apellidos + "\t" + Registro[i].Sexo + "\t" + Registro[i].FechaNac.Mes + "\t" + Registro[i].FechaNac.Dia + "\t" + Registro[i].FechaNac.Año);
    }
}
```

A favor de que se visualizara la plenitud de la función, se redujo al mínimo posible. No es necesario leer porque lo que importa es que funciona.

4. Ejecución del programa

Finalmente, el resultado de nuestro laboratorio arrojó la siguiente interfaz:

Carnet	Nombres	Apellidos	Sexo	Nacimiento
2020-1384U	Marcel	Díaz	M	Febrero 5 2002

Se asignó un tamaño de dos elementos al arreglo de la estructura. Inmediatamente, se rellenaron los campos y se ingresó un estudiante; se mostró en pantalla.

Form1

Cantidad de elementos a ingresar: Establecer

Camet: Nombre: Apellido:

Sexo:

Fecha de nacimiento: Día: Mes: Año:

Registrar Borrar Actualizar

Imprimir

Camet	Nombres	Apellidos	Sexo	Nacimiento
2020-1384U	Marcel	Díaz	M	Febrero 5 2002
2020-0505U	Engel	Reyes	F	Mayo 10 2003

Se ingresó un segundo estudiante y se mostró en pantalla

Form1

Cantidad de elementos a ingresar: Establecer

Camet: Nombre: Apellido:

Sexo:

Fecha de nacimiento: Día: Mes: Año:

Registrar Borrar Actualizar

Imprimir

Camet	Nombres	Apellidos	Sexo	Nacimiento
2020-1384U	Marcel	Díaz	M	Febrero 5 2002
2020-0505U	Engel	Reyes	F	Mayo 10 2003

Cerciorándonos de que la validación funcione

Form1

Cantidad de elementos a ingresar: Establecer

Camet: Nombre: Apellido:

Sexo: Fecha de nacimiento:

Registrar Borrar Actualizar

Imprimir

Camet	Nombres	Apellidos	Sexo	Nacimiento
2020-1384U	Marcel	Díaz	M	Febrero 5 2002
2020-0505U	Engel	Reyes	F	Mayo 10 2003

Se eliminó un estudiante y se volvieron a imprimir los datos existentes en pantalla

Form1

Cantidad de elementos a ingresar: Establecer

Camet: Nombre: Apellido:

Sexo: Fecha de nacimiento:

Registrar Borrar Actualizar

Imprimir

Camet	Nombres	Apellidos	Sexo	Nacimiento
2020-0505U	Engel	Reyes	F	Mayo 10 2003

Form1

Cantidad de elementos a ingresar: Establecer

Carnet:
 Nombre:
 Apellido:

×

El estudiante con carnet 2020-0505U se ha actualizado.

Aceptar

Carnet	Nombres	Apellidos	Sexo	Nacimiento
2020-0505U	Engel	Reyes	F	Mayo 10 2003

Actualizando los datos de un estudiante

Form1

Cantidad de elementos a ingresar: Establecer

Carnet:
 Nombre:
 Apellido:

Sexo:
 Fecha de nacimiento:

Día:
 Mes:
 Año:

Registrar Borrar Actualizar

Imprimir

Carnet	Nombres	Apellidos	Sexo	Nacimiento
2020-0505U	Ángela	Reynosa	M	Mayo 10 3002

Volviendo a imprimir los datos en pantalla

5. Conclusiones

La práctica de laboratorio culminó de forma efectiva. Si bien el manejo y la administración de los datos en una estructura de datos puede llegar a ser confusa y poco inteligible, sobre todo si se trata de un arreglo de las mismas, en esta ocasión fueron trabajadas de una forma sencilla y muy bien abstraída, por lo que todo lo que ha sido practicado con este laboratorio fue aprehendido con efecto.