

**UNIVERSIDAD NACIONAL DE INGENIERÍA FACULTAD DE ELECTROTECNIA Y
COMPUTACIÓN**

REPORTE DE LABORATORIO N°3

ALGORITMIZACIÓN Y ESTRUCTURA DE DATOS

PROF. ADILSON GONZALEZ

NOMBRE:

- **ALANIS DEL CARMEN TOBIAS SOTELO 2021-0148U**
- **MADELING REGINA CABRERA ROJAS 2021-0257U**

FECHA:

19/09/2023

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.Runtime.InteropServices;

```

```

namespace RutaDeAcceso
{

```

```

    3 referencias
    public partial class Form1 : Form
    {

```

```

        Stack<String> NombresCarpetas = new Stack<string>();
        Stack<String> RutasDeAcceso = new Stack<string>();
        int CantidadDeCarpetas;

```

```

        1 referencia
        public Form1()
        { InitializeComponent(); }
    }

```

```
/*Funciones del panel Nueva Carpeta.*/
```

```
1 referencia
```

```
private void txtNuevaCarpeta_KeyDown(object sender, KeyEventArgs e)
{ if (e.KeyCode == Keys.Enter) btnAgregarCarpeta_Click(sender, e); }
```

```
2 referencias
```

```
private void btnAgregarCarpeta_Click(object sender, EventArgs e)
{
    if (txtNuevaCarpeta.Text.Trim() == "")
        MessageBox.Show("Ingrese un nombre valido.");
    else
    {
        NombresCarpetas.Push(txtNuevaCarpeta.Text.Trim());
        CantidadDeCarpetas = NombresCarpetas.Count();
        MostrarCarpetas();
        txtNuevaCarpeta.Clear();
    }
}
```

```
1 referencia
```

```
private void txtRuta_KeyDown(object sender, KeyEventArgs e)
{ if (e.KeyCode == Keys.Enter) btnAbrir_Click_1(sender, e); }
```

```
2 referencias
```

```
private void btnAbrir_Click_1(object sender, EventArgs e)
{
    if (cbRutas.SelectedItem != null)
    {
        // Obtén la ruta seleccionada en el ComboBox.
        string rutaSeleccionada = cbRutas.SelectedItem.ToString();

        // Divide la ruta en partes utilizando "\\" como separador.
        string[] partesRuta = rutaSeleccionada.Split('\\');

        // Verifica si hay al menos una parte en la ruta.
        if (partesRuta.Length > 0)
        {
            // Obtiene la última parte de la ruta (nombre de la carpeta).
            string ultimaCarpeta = partesRuta[partesRuta.Length - 1];

            // Concatena el texto deseado con el nombre de la carpeta.
            string textoConcatenado = "Carpeta Abierta: " + ultimaCarpeta;

            // Asigna el texto concatenado al cuadro de texto txtRutaImpresa.
            txtRutaImpresa.Text = textoConcatenado;
        }
    }
    else
    {
        MessageBox.Show("No se ha seleccionado ninguna ruta.");
    }
}
```

```

/*
Función que crear una carpetita(imagen)
para la nueva carpeta en la pila.
*/
1 referencia
public void MostrarCarpetas()
{
    var Titulo = new Label();
    Titulo.Visible = true;
    Titulo.Size = new System.Drawing.Size(227, 20);
    Titulo.Text = "Directorio";
    Titulo.TextAlign = ContentAlignment.TopCenter;
    Titulo.Font = new Font("Microsoft Sans Serif", 10, FontStyle.Bold);
    Titulo.BackColor = Color.FromArgb(255, 128, 0);
    Titulo.Location = new System.Drawing.Point(0, 0);

    /*Ubicación y tamaño de la primera carpeta.*/
    int x_locacion = 14, y_locacion = 28,
        x_tamaño = 25, y_tamaño = 30;
    panel3.Controls.Clear();
    panel3.Controls.Add(Titulo);
    for (int indice = (CantidadDeCarpetas - 1); indice > -1; indice--)
    {
        var imagen = new PictureBox(); /*Para la carpetita.*/
        imagen.Visible = true;
        imagen.Image = Properties.Resources.Carpetita;
        imagen.SizeMode = PictureBoxSizeMode.Zoom;
        imagen.Size = new System.Drawing.Size(x_tamaño, y_tamaño);
        imagen.Location = new System.Drawing.Point(x_locacion, y_locacion);

```

```

        var etiqueta = new Label(); /*Para el nombre.*/
        etiqueta.Visible = true;
        etiqueta.AutoSize = true;
        etiqueta.Text = NombresCarpetas.ElementAt(indice);
        etiqueta.ForeColor = Color.Blue;
        etiqueta.Font = new Font("Microsoft Sans Serif", 10, FontStyle.Bold);
        etiqueta.Location = new System.Drawing.Point(x_locacion + 30, y_locacion + 7);

        panel3.Controls.Add(imagen);
        panel3.Controls.Add(etiqueta);

        x_locacion += 31;
        y_locacion += 36;
    }
    MostrarRutas();
}

```

/*Devuelve la ruta, el nombre*/

0 referencias

public string Busqueda(string Dato)

```
{
    string resultado = null;
    for (int j = 0; j < CantidadDeCarpetas; j++)
    {
        if (string.Equals(RutasDeAcceso.ElementAt(j), Dato))
        { resultado = NombresCarpetas.ElementAt(j); break; }
    }
    for (int j = 0; j < CantidadDeCarpetas; j++)
    {
        if (string.Equals(NombresCarpetas.ElementAt(j), Dato))
        { resultado = RutasDeAcceso.ElementAt(j); break; }
    }
    return resultado;
}
```

/*Le asigna la elección del ComboBox a txtRuta.*/

1 referencia

private void cbRutas_SelectedIndexChanged(object sender, EventArgs e)

```
{
    txtRuta.Text = cbRutas.Text;
    cbRutas.Text = "";
}
```

}

/*Muestra las direcciones de cada carpeta en el ComboBox.*/

1 referencia

void MostrarRutas()

```
{
    cbRutas.Items.Clear();
    if (CantidadDeCarpetas > 0)
    {
        string cadena = NombresCarpetas.ElementAt(CantidadDeCarpetas - 1);
        cbRutas.Items.Add(cadena + "\\");
        RutasDeAcceso.Push(cadena + "\\");
        for (int indice = (CantidadDeCarpetas - 2); indice > -1; indice--)
        {
            cadena += "\\ " + NombresCarpetas.ElementAt(indice);

            RutasDeAcceso.Push(cadena);
            cbRutas.Items.Add(cadena);
        }
    }
}
```

Form1

Nueva Carpeta

NOTAS

Agregar

Directorio

C

IISEMESTRE

UNI

ADILSON

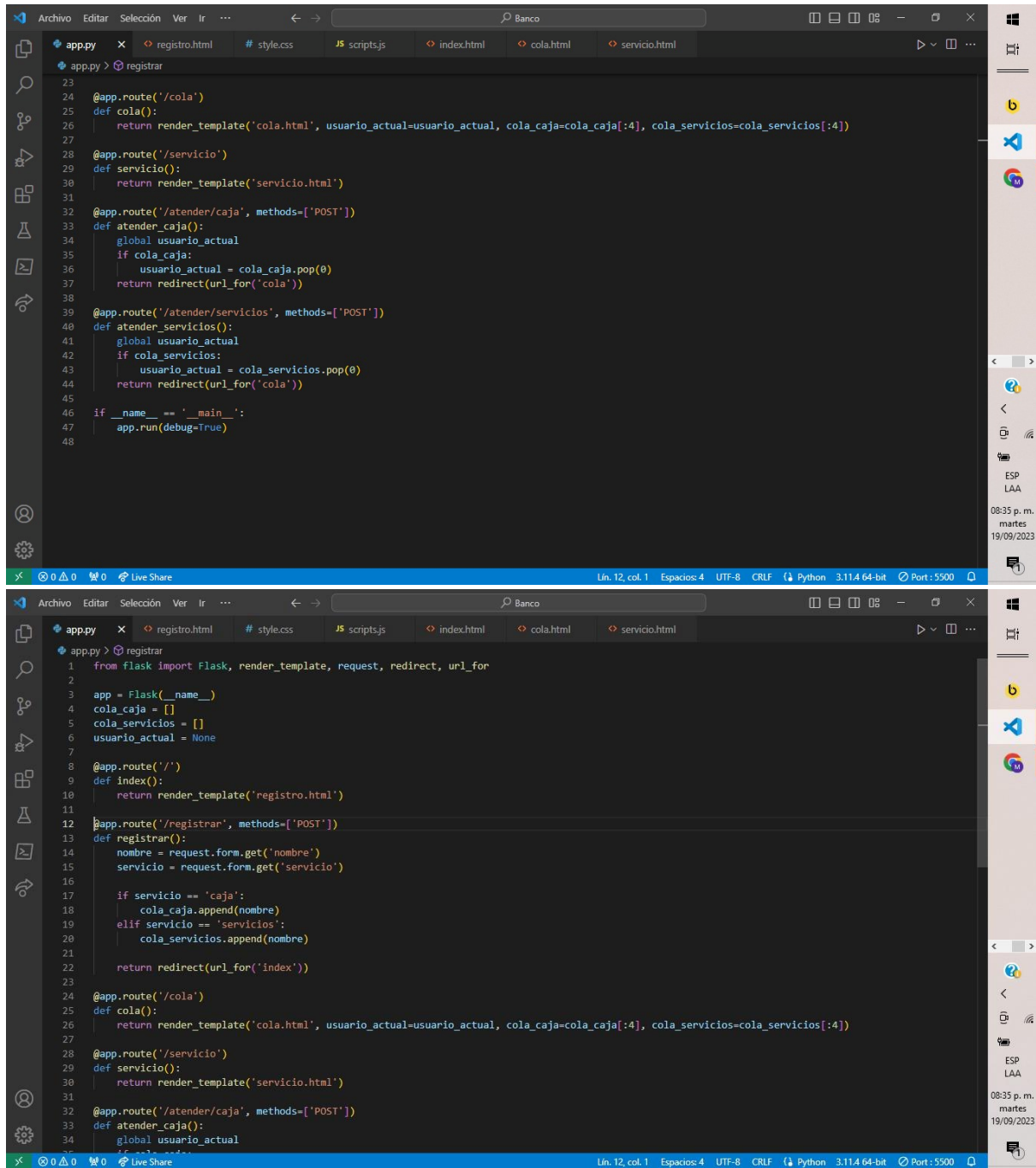
Buscar

Ruta:

C\
C\IISEMESTRE
C\IISEMESTRE\UNI
C\IISEMESTRE\UNI\ADILSON

Abrir

SEGUNDO EJERCICIO



```
23
24 @app.route('/cola')
25 def cola():
26     return render_template('cola.html', usuario_actual=usuario_actual, cola_caja=cola_caja[:4], cola_servicios=cola_servicios[:4])
27
28 @app.route('/servicio')
29 def servicio():
30     return render_template('servicio.html')
31
32 @app.route('/atender/caja', methods=['POST'])
33 def atender_caja():
34     global usuario_actual
35     if cola_caja:
36         usuario_actual = cola_caja.pop(0)
37     return redirect(url_for('cola'))
38
39 @app.route('/atender/servicios', methods=['POST'])
40 def atender_servicios():
41     global usuario_actual
42     if cola_servicios:
43         usuario_actual = cola_servicios.pop(0)
44     return redirect(url_for('cola'))
45
46 if __name__ == '__main__':
47     app.run(debug=True)
48
```

```
1 from flask import Flask, render_template, request, redirect, url_for
2
3 app = Flask(__name__)
4 cola_caja = []
5 cola_servicios = []
6 usuario_actual = None
7
8 @app.route('/')
9 def index():
10     return render_template('registro.html')
11
12 @app.route('/registrar', methods=['POST'])
13 def registrar():
14     nombre = request.form.get('nombre')
15     servicio = request.form.get('servicio')
16
17     if servicio == 'caja':
18         cola_caja.append(nombre)
19     elif servicio == 'servicios':
20         cola_servicios.append(nombre)
21
22     return redirect(url_for('index'))
23
24 @app.route('/cola')
25 def cola():
26     return render_template('cola.html', usuario_actual=usuario_actual, cola_caja=cola_caja[:4], cola_servicios=cola_servicios[:4])
27
28 @app.route('/servicio')
29 def servicio():
30     return render_template('servicio.html')
31
32 @app.route('/atender/caja', methods=['POST'])
33 def atender_caja():
34     global usuario_actual
35     if cola_caja:
36         usuario_actual = cola_caja.pop(0)
37     return redirect(url_for('cola'))
38
39 @app.route('/atender/servicios', methods=['POST'])
40 def atender_servicios():
41     global usuario_actual
42     if cola_servicios:
43         usuario_actual = cola_servicios.pop(0)
44     return redirect(url_for('cola'))
45
```

Document

127.0.0.1:5000/servicio

En pausa

Navbar Registro Servicio Clientes Disabled

Notificar Siguiente Atención

Atender Cliente en Caja:

Atender Siguiente en Caja

Atender Cliente en Servicios Bancarios:

Atender Siguiente en Servicios Bancarios

En pausa

ESP LAA

08:34 p. m.
martes
19/09/2023

Document

127.0.0.1:5000/cola

En pausa

Navbar Registro Servicio Clientes Disabled

Cola de Atención

Usuario siendo atendido:

None

Próximos 4 a seguir en Caja:

- Madeling
- Regina
- Madeling

Próximos 4 a seguir en Servicios Bancarios:

En pausa

ESP LAA

08:34 p. m.
martes
19/09/2023

