

Universidad Nacional de Ingeniería

Facultad de Electrotecnia y Computación -.Ingeniería en Computación

Algoritmización y estructuras de datos Laboratorio No. 7

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	
3.1. Declaraciones y métodos iniciales	
3.2. Métodos de interacción con la interfaz	
4. Ejecución del programa	
5. Conclusiones	

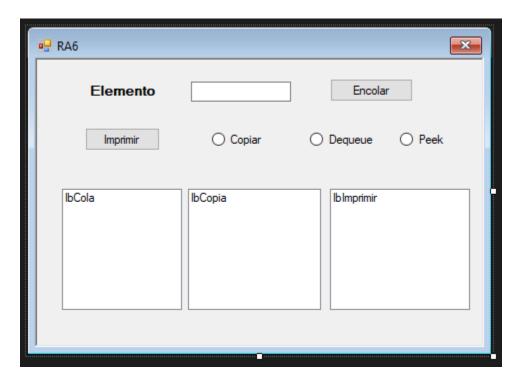
1. Introducción

En esta ocasión, se abordará el concepto de colas. Similarmente al trabajo práctico anterior, se trata de una estructura de datos en la que el orden de los elementos insertados dicta la entrada y salida de datos. Sin embargo, a diferencia de las pilas, las colas persiguen el concepto de **FIFO** (Fist Input First Output), es decir, el primer dato que haya sido ingresado a la estructura será el primero que será devuelto.

A continuación, el reporte de las capturas de pantalla del código practicado.

2. Diseño del form

Se trabajó la interfaz propuesta en clases.



3. Código Fuente

3.1. Declaraciones y métodos iniciales.

Se declara la estructura de tipo **Queue** de forma similar a como se declaraba una fila. Sólo que, esta vez, solamente el nombre del tipo de dato es lo que cambia:

```
Queue<string> ColaNombre = new Queue<string>();
```

La siguiente función nos permitirá agregar a variable de tipo cola los elementos que vayamos insertando. Entonces, los muestra en pantalla por medio de **lbImprimir:**

```
2 referencias
public void Agregar()
{
    ColaNombre.Enqueue(txtElemento.Text);
    lbCola.Items.Clear();

    foreach(var Item in ColaNombre)
    {
        lbCola.Items.Add(Item);
        lbCola.Items.Add("-----");
    }

    txtElemento.Clear();
    txtElemento.Focus();
}
```

Cada vez que se presione la tecla *enter* al estar escribiendo en la caja de texto **txtElemento**, se ingresará el dato llamando a la función anterior:

```
1referencia
private void txtElemento_KeyPress(object sender, KeyPressEventArgs e)
{
    if(e.KeyChar == 13)
        Agregar();
}
```

3.2. Métodos de interacción con la interfaz.

<u>Radiobotón Copiar - checado:</u> Cuando sea checado el radiobotón, se imprimirán en el panel **lbCopia** los elementos existentes en la cola:

```
1referencia
private void rbCopiar_CheckedChanged(object sender, EventArgs e)
{
    if(rbCopiar.Checked == true)
    {
        lbCopia.Items.Clear();
        foreach(var Item in ColaNombre)
        {
            lbCopia.Items.Add(Item);
            lbCopia.Items.Add("-----");
        }
    }
}
```

<u>Radiobotón Dequeue – checado:</u> Evalúa si se ha checado el radiobotón. Si lo fue, elimina de la cola el primer elemento insertado:

```
referencia
private void rbPop_CheckedChanged(object sender, EventArgs e)
{
    if (rbDequeue.Checked == true)
    {
        lbImprimir.Items.Add("Sacar de cola:" + ColaNombre.Dequeue());
        lbCopia.Items.Clear();
        foreach (var Item in ColaNombre)
        {
             lbCopia.Items.Add(Item);
             lbCopia.Items.Add("-----");
        }
    }
}
```

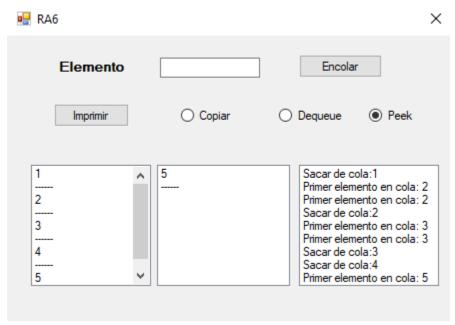
<u>Radiobotón peek - checado:</u> Si ha sido checado, indica en el panel correspondiente el primer elemento en la cola:

```
private void rbPeek_CheckedChanged(object sender, EventArgs e)
{
    if (rbPeek.Checked == true)
    {
        lbImprimir.Items.Add("Primer elemento en cola: " + ColaNombre.Peek());
        lbCopia.Items.Clear();
        foreach (var Item in ColaNombre)
        {
            lbCopia.Items.Add(Item);
            lbCopia.Items.Add("-----");
        }
    }
}
```

Botón Imprimir: Imprime en el panel **lbImprimir** los elementos dentro de la cola:

4. Ejecución del programa

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



Ingresando datos, eliminando y obteniendo el primer dato.

5. Conclusiones

Con esta práctica se culmina con un nuevo tipo de estructura de datos. A mi parecer, la utilidad de las mismas se encuentra en circunstancias bastante particulares, pero no cabe duda que, para manejar los datos en esas precisas circunstancias, lo harían con efecto.