

T6. Pilas

¿Qué es una pila?

Una pila es una lista ordenada o estructura de datos que permite almacenar y recuperar datos. Esta estructura se aplica en multitud de supuestos en el área de informática debido a su simplicidad y capacidad de dar respuesta a numerosos procesos.

Para el manejo de los datos cuenta con dos operaciones básicas: apilar (push), que coloca un objeto en la pila, y su operación inversa, retirar (o desapilar, pop), que retira el último elemento apilado.

Operaciones

Una pila cuenta con 2 operaciones imprescindibles: apilar y desapilar, a las que en las implementaciones modernas de las pilas se suelen añadir más de uso habitual.

- **Crear:** se crea la pila vacía. (constructor)
- **Tamaño:** regresa el número de elementos de la pila. (size)
- **Apilar:** se añade un elemento a la pila. (push)
- **Desapilar:** se elimina el elemento frontal de la pila. (pop)
- **Cima:** devuelve el elemento que está en la cima de la pila. (top o peek)
- **Vacía:** devuelve cierto si la pila está vacía o falso en caso contrario (empty).

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Maquina_de_refrescos
{
    class cola
    {
        int[] vec;
        int p, u, tam;

        public cola(int max)
        {
            tam = max;
            vec = new int[tam];
            p = u = -1;
        }

        public int this[int v]
        {
            set
            { vec[v] = Convert.ToInt16(value); }
            get
            { }
        }
    }
}
```

```

        { return vec[v]; }
    }

    public bool esta_llena()
    {
        if (u >= tam - 1)
        {
            return true;
        }
        return false;
    }

    public bool esta_vacia()
    {
        if (p == -1)
        {
            return true;
        }
        return false;
    }

    public bool agregar(int dato)
    {
        if (!esta_llena())
        {
            vec[++u] = dato;
            if (u == 0)
            {
                p = 0;
            }
            return true;
        }
        return false;
    }

    public bool extraer(ref int dato)
    {
        if (!esta_vacia())
        {
            dato = vec[p];
            if (p == u)
            {
                p = -1;
                u = p;
            }
            else
            {
                p++;
            }
            return true;
        }
        return false;
    }
}

```