

INTRODUÇÃO

1)

```
using System;
```

```
using System.Collections;
```

```
class Fila
```

```
{
```

```
    private ArrayList elementos;
```

```
    public Fila()
```

```
    {
```

```
        elementos = new ArrayList();
```

```
    }
```

```
    public void Inserir(object elemento)
```

```
    {
```

```
        elementos.Add(elemento);
```

```
    }
```

```
    public object Remover()
```

```
    {
```

```
        if (elementos.Count > 0)
```

```
        {
```

```
        object elementoRemovido = elementos[0];
        elementos.RemoveAt(0);
        return elementoRemovido;
    }
    else
    {
        throw new InvalidOperationException("A fila está vazia.");
    }
}
```

```
public void MostrarElementos()
{
    foreach (object elemento in elementos)
    {
        Console.WriteLine(elemento);
    }
}
```

```
class Program
{
    static void Main(string[] args)
```

```
{  
    Fila minhaFila = new Fila();  
  
    minhaFila.Inserir("Elemento 1");  
    minhaFila.Inserir("Elemento 2");  
    minhaFila.Inserir("Elemento 3");  
  
    Console.WriteLine("Elementos na fila:");  
    minhaFila.MostrarElementos();  
  
    Console.WriteLine("Removendo elemento: " +  
minhaFila.Remover());  
  
    Console.WriteLine("Elementos na fila após remoção:");  
    minhaFila.MostrarElementos();  
}  
}
```

2)

```
using System;  
using System.Collections;
```

```
class Pilha
```

```
{
```

```
private ArrayList elementos;
```

```
public Pilha()
```

```
{
```

```
    elementos = new ArrayList();
```

```
}
```

```
public void Inserir(object elemento)
```

```
{
```

```
    elementos.Add(elemento);
```

```
}
```

```
public object Remover()
```

```
{
```

```
    if (elementos.Count > 0)
```

```
    {
```

```
        int ultimoIndice = elementos.Count - 1;
```

```
        object elementoRemovido = elementos[ultimoIndice];
```

```
        elementos.RemoveAt(ultimoIndice);
```

```
        return elementoRemovido;
```

```
    }
```

```
    else
```

```
    {
```

```
        throw new InvalidOperationException("A pilha está vazia.");
```

```
    }
```

```
}
```

```
public void MostrarElementos()
```

```
{
```

```
    for (int i = elementos.Count - 1; i >= 0; i--)
```

```
    {
```

```
        Console.WriteLine(elementos[i]);
```

```
    }
```

```
}
```

```
}
```

```
class Program
```

```
{
```

```
    static void Main(string[] args)
```

```
    {
```

```
        Pilha minhaPilha = new Pilha();
```

```
        minhaPilha.Inserir("Elemento 1");
```

```
        minhaPilha.Inserir("Elemento 2");
```

```
        minhaPilha.Inserir("Elemento 3");
```

```
Console.WriteLine("Elementos na pilha:");  
minhaPilha.MostrarElementos();
```

```
Console.WriteLine("Removendo elemento: " +  
minhaPilha.Remover());
```

```
Console.WriteLine("Elementos na pilha após remoção:");  
minhaPilha.MostrarElementos();
```

```
}
```

```
}
```

3)

```
using System;
```

```
using System.Collections.Generic;
```

```
class Fila
```

```
{
```

```
    private Stack<object> entrada;
```

```
    private Stack<object> saida;
```

```
    public Fila()
```

```
    {
```

```
        entrada = new Stack<object>();
```

```
        saida = new Stack<object>();
```

```
}
```

```
public void Inserir(object elemento)
```

```
{
```

```
    entrada.Push(elemento);
```

```
}
```

```
public object Remover()
```

```
{
```

```
    if (saida.Count == 0)
```

```
    {
```

```
        while (entrada.Count > 0)
```

```
        {
```

```
            saida.Push(entrada.Pop());
```

```
        }
```

```
    }
```

```
    if (saida.Count > 0)
```

```
    {
```

```
        return saida.Pop();
```

```
    }
```

```
    else
```

```
    {
```

```
        throw new InvalidOperationException("A fila está vazia.");
```

```
    }
```

```
}
```

```
public void MostrarElementos()
```

```
{
```

```
    foreach (object elemento in saida)
```

```
    {
```

```
        Console.WriteLine(elemento);
```

```
    }
```

```
Stack<object> temp = new Stack<object>(saida);
```

```
while (temp.Count > 0)
```

```
{
```

```
    entrada.Push(temp.Pop());
```

```
}
```

```
}
```

```
}
```

```
class Program
```

```
{
```



```

static void Main(string[] args)
{
    Fila minhaFila = new Fila();

    minhaFila.Inserir("Elemento 1");
    minhaFila.Inserir("Elemento 2");
    minhaFila.Inserir("Elemento 3");

    Console.WriteLine("Elementos na fila:");
    minhaFila.MostrarElementos();

    Console.WriteLine("Removendo elemento: " +
minhaFila.Remove());

    Console.WriteLine("Elementos na fila após remoção:");
    minhaFila.MostrarElementos();
}
}

```

4)

```

using System;
using System.Collections.Generic;

class Pilha

```

```
{  
    private Queue<object> entrada;  
    private Queue<object> saida;  
  
    public Pilha()  
    {  
        entrada = new Queue<object>();  
        saida = new Queue<object>();  
    }  
  
    public void Inserir(object elemento)  
    {  
        entrada.Enqueue(elemento);  
    }  
  
    public object Remover()  
    {  
        if (saida.Count == 0)  
        {  
            while (entrada.Count > 1)  
            {  
                saida.Enqueue(entrada.Dequeue());  
            }  
        }  
    }  
}
```

```
}

if (saida.Count > 0)
{
    return saida.Dequeue();
}
else if (entrada.Count > 0)
{
    return entrada.Dequeue();
}
else
{
    throw new InvalidOperationException("A pilha está vazia.");
}

}

public void MostrarElementos()
{
    Queue<object> temp = new Queue<object>(saida);

    while (temp.Count > 0)
    {
```

```
        Console.WriteLine(temp.Dequeue());
    }

    foreach (object elemento in entrada)
    {
        Console.WriteLine(elemento);
    }
}

class Program
{
    static void Main(string[] args)
    {
        Pilha minhaPilha = new Pilha();

        minhaPilha.Inserir("Elemento 1");
        minhaPilha.Inserir("Elemento 2");
        minhaPilha.Inserir("Elemento 3");

        Console.WriteLine("Elementos na pilha:");
        minhaPilha.MostrarElementos();
    }
}
```

```
Console.WriteLine("Removendo elemento: " +  
minhaPilha.Remover());
```

```
Console.WriteLine("Elementos na pilha após remoção:");  
minhaPilha.MostrarElementos();  
}  
}
```

EXERCÍCIOS RESOLVIDOS

1)

```
using System;
```

```
public class Lista
```

```
{
```

```
    private int[] elementos;
```

```
    private int tamanho;
```

```
    public Lista(int capacidade)
```

```
    {
```

```
        elementos = new int[capacidade];
```

```
        tamanho = 0;
```

```
    }
```

```
public void InserirInicio(int elemento)
{
    if (tamanho < elementos.Length)
    {
        for (int i = tamanho; i > 0; i--)
        {
            elementos[i] = elementos[i - 1];
        }
        elementos[0] = elemento;
        tamanho++;
    }
    else
    {
        Console.WriteLine("A lista está cheia.");
    }
}
```

```
public void InserirFim(int elemento)
{
    if (tamanho < elementos.Length)
    {
```

```
        elementos[tamanho] = elemento;
        tamanho++;
    }
    else
    {
        Console.WriteLine("A lista está cheia.");
    }
}
```

```
public void Inserir(int elemento, int posicao)
{
    if (posicao >= 0 && posicao <= tamanho && tamanho <
    elementos.Length)
    {
        for (int i = tamanho; i > posicao; i--)
        {
            elementos[i] = elementos[i - 1];
        }
        elementos[posicao] = elemento;
        tamanho++;
    }
    else
```

```
    {  
        Console.WriteLine("Posição inválida ou a lista está  
cheia.");  
    }  
}
```

```
public int RemoverInicio()  
{  
    if (tamanho > 0)  
    {  
        int valorRemovido = elementos[0];  
        for (int i = 0; i < tamanho - 1; i++)  
        {  
            elementos[i] = elementos[i + 1];  
        }  
        tamanho--;  
        return valorRemovido;  
    }  
    else  
    {  
        Console.WriteLine("A lista está vazia.");  
        return -1;  
    }  
}
```



```
    }  
}  
  
public int RemoverFim()  
{  
    if (tamanho > 0)  
    {  
        int valorRemovido = elementos[tamanho - 1];  
        tamanho--;  
        return valorRemovido;  
    }  
    else  
    {  
        Console.WriteLine("A lista está vazia.");  
        return -1;  
    }  
}  
  
public int Remover(int posicao)  
{  
    if (posicao >= 0 && posicao < tamanho)  
    {
```

```
        int valorRemovido = elementos[posicao];  
        for (int i = posicao; i < tamanho - 1; i++)  
        {  
            elementos[i] = elementos[i + 1];  
        }  
        tamanho--;  
        return valorRemovido;  
    }  
    else  
    {  
        Console.WriteLine("Posição inválida.");  
        return -1;  
    }  
}
```

```
public void Mostrar()  
{  
    Console.WriteLine("Lista: ");  
    for (int i = 0; i < tamanho; i++)  
    {  
        Console.Write(elementos[i] + " ");  
    }  
}
```

```

        Console.WriteLine();
    }
}

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("==== LISTA LINEAR ====");
        Lista lista = new Lista(6);
        int x1, x2, x3;
        lista.InserirInicio(1);
        lista.InserirFim(7);
        lista.InserirFim(9);
        lista.InserirInicio(3);
        lista.Inserir(8, 3);
        lista.Inserir(4, 2);
        lista.Mostrar();
        x1 = lista.RemoverInicio();
        x2 = lista.RemoverFim();
        x3 = lista.Remover(2);
        Console.WriteLine(x1 + ", " + x2 + ", " + x3);
    }
}

```

```
        lista.Mostrar();  
    }  
}
```