

### Class OperacionesMatrices

```
using System.Data;
using System.Numerics;
using System.Security.Cryptography.X509Certificates;
using Microsoft.VisualBasic;

class OperacionesMatrices
{
    public int[,] matriz = new int[0, 0];

    public OperacionesMatrices()
    {
    }

    public void CrearMatriz()
    {
        int cantidadFilas = 0;
        int cantidadCols = 0;

        Console.WriteLine("Ingrese la cantidad de filas de la matriz");
        cantidadFilas = Int32.Parse(Console.ReadLine());
        Console.WriteLine("Ingrese la cantidad de columnas de la matriz");
        cantidadCols = Int32.Parse(Console.ReadLine());

        matriz = new int[cantidadFilas, cantidadCols];
    }

    public void IngresarDatosMatriz()
    {
        //Ejemplos con dos for (hay más formas)
        for (int fila = 0; fila < matriz.GetLength(0); fila++)
        {
            for (int columna = 0; columna < matriz.GetLength(1); columna++)
            {
                Console.WriteLine($"Ingrese valor para la posicion [{fila}][{columna}]");
                matriz[fila, columna] = Int32.Parse(Console.ReadLine());
            }
        }
    }

    public int[,] MultiplicaciónMatrizEscalar(int escalar)
    {
    }
}
```

```
        int[, ] matrizMultiplicada = new int[matriz.GetLength(0),
matriz.GetLength(1)];

        //Ejemplos con dos for (hay más formas)
        for (int fila = 0; fila < matriz.GetLength(0); fila++)
        {
            for (int columna = 0; columna < matriz.GetLength(1); columna++)
            {
                matrizMultiplicada[fila, columna] = matriz[fila, columna] *
escalar;
            }
        }

        return matrizMultiplicada;
    }

    public void ImprimirMatriz(int[, ] matriz)
    {
        for (int i = 0; i < matriz.GetLength(0); i++)
        {
            for (int j = 0; j < matriz.GetLength(1); j++)
            {

                Console.WriteLine(matriz[i, j]);

            }
        }
    }

    public void BuscarDato(int dato)
    {

        for (int fila = 0; fila < matriz.GetLength(0); fila++)
        {
            for (int columna = 0; columna < matriz.GetLength(1); columna++)
            {

                if (dato != matriz[fila, columna])
                {
                    if (dato == matriz[fila, columna])
                    {
                        Console.WriteLine($"Las coordenadas de {dato} son
[{{fila}},{{columna}}]");
                    }
                }
            }
        }
    }
}
```

```
        }
        else
        {
            Console.WriteLine($"Las coordenadas de {dato} son
[{fila},{columna}]");
        }
    }

}

}

public int[] datoPar()
{
    int n = matriz.GetLength(0) * matriz.GetLength(1);
    int t = 0;
    int[] arregloPar = new int[n];

    for (int fila = 0; fila < matriz.GetLength(0); fila++)
    {
        for (int columna = 0; columna < matriz.GetLength(1); columna++)
        {
            if (matriz[fila, columna] % 2 != 0)
            {
                if (matriz[fila, columna] % 2 == 0)
                {
                    arregloPar[t] = matriz[fila, columna];
                    t++;
                }
            }
            else
            {
                arregloPar[t] = matriz[fila, columna];
                t++;
            }
        }
    }

    return arregloPar;
}
```



Josué David Reyes Muñoz  
1029024

```
        break;

        case 'c':
            int[] arregloN = operacionesMatrices.datoPar();

            for(int i=0; i<arregloN.Length; i++){
                Console.WriteLine(arregloN[i]);
            }

            break;
        }
    }
}
```