

---

# PRACTICA N<sup>o</sup> 3 MATRICES

---

Materia: Estructuras de datos SIS (312)

Docente: Ing. Limber Ruiz Molina.

Estudiante: Univ. Rodriguez Condori Kevin Alexis.

---

Ejercicio 1:

```
package Tarea_3;
import java.util.Scanner;
/**
 * Realice un programa que permita insertar números en una matriz de tamaño M X N,
 * posteriormenmuestre la cantidad de números pares de cada fila.
```

Ejemplo:

```
* Entrada      Salida
  3 4          4 2 1
  4 6 2 2
  6 7 4 3
  3 6 3 1
*/
```

```
public class Example_1 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int M = input.nextInt();
        int N = input.nextInt();
        int[][] A = new int[M][N];
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                A[i][j] = input.nextInt();
            }
        }

        for (int i = 0; i < A.length; i++) {
            int pares = 0;
            for (int j = 0; j < A[i].length; j++) {
                if (A[i][j] % 2 == 0) {
                    pares++;
                }
            }
            System.out.print(pares + " ");
        }
    }
}
```

## Ejercicio 2:

```
package Tarea_3;
import java.util.Scanner;
/**
 * Realice un programa que permita insertar números en una matriz de tamaño M X N,
 * posteriormente muestre el número más grande de cada fila
 * Ejemplo:
 * Entrada      Salida
 * 3 4          6 7 6
 * 4 6 2 2
 * 6 7 4 3
 * 3 6 3 1
 */
public class Example_2 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int M = input.nextInt();
        int N = input.nextInt();
        int[][] A = new int[M][N];
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                A[i][j] = input.nextInt();
            }
        }

        for (int i = 0; i < A.length; i++) {
            int bigNumberRow = 0;
            for (int j = 0; j < A[i].length; j++) {
                if (A[i][j] > bigNumberRow) {
                    bigNumberRow = A[i][j];
                }
            }
            System.out.print(bigNumberRow + " ");
        }
    }
}
```

### Ejercicio 3:

```
package Tarea_3;
import java.util.Scanner;
/**
 *
 * @author krodr
 * Realice un programa que permita insertar números en una matriz de tamaño M X N, posteriormente
 * muestre toda la fila donde se encuentra el número más grande de la matriz
 Ejemplo:
 Entrada      Salida
 3 4          6 7 4 3
 4 6 2 2
 6 7 4 3
 3 6 3 1
 */
public class Example_3 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int M = input.nextInt();
        int N = input.nextInt();
        int[][] A = new int[M][N];

        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                A[i][j] = input.nextInt();
            }
        }

        int bigNumberRow = 0;
        int indexRow = 0;
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                if (A[i][j] > bigNumberRow) {
                    bigNumberRow = A[i][j];
                    indexRow = i;
                }
            }
        }

        for (int i = 0; i < N; i++) {
            System.out.print(A[indexRow][i] + " ");
        }
    }
}
```

#### Ejercicio 4:

```
package Tarea_3;
import java.util.Scanner;
/**
 *
 * @author krodr
 */
public class Example_4 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int N = input.nextInt();
        int A[][] = new int[N][N];

        int positionColumn = N/2;
        int positionRow = 0;
        A[positionRow][positionColumn] = 1;

        int currentPositionRow = positionRow;
        int currentPositionColumn = positionColumn;

        for (int i = 2; i <= (Math.pow(A.length, 2)); i++) {
            positionRow--;
            positionColumn--;

            if (positionRow < 0) {
                positionRow = A.length-1;
            }
            if(positionColumn < 0){
                positionColumn = A.length-1;
            }
            if (A[positionRow][positionColumn] == 0) {
                A[positionRow][positionColumn] = i;
            }else{
                positionRow = currentPositionRow + 1;
                positionColumn = currentPositionColumn;
                A[positionRow][positionColumn] = i;
            }
            currentPositionRow = positionRow;
            currentPositionColumn = positionColumn;
        }
        for (int i = 0; i < A.length; i++) {
            for (int j = 0; j < A[i].length; j++) {
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```