

LABORATORIO 08

El laboratorio consiste en una serie de ejercicios para practicar la resolución de problemas usando el método visto en la clase de Introducción a la Programación, listas, matrices y ordenamientos.

El laboratorio es individual, deberán subir un archivo con el nombre Laboratorio8.py, en la asignación propia del GitHub. Se entrega el **viernes 28 de mayo hasta las 10 pm**. Para estas funciones haremos del uso del **While y For**

Ejercicio #1 (40 puntos)

Escriba una función **encontrarNumerosDivisibles(matriz, num)**. Esta función recibe una **matriz** de tamaño $m \times n$, consiste en hacer el recorrido **columna por columna**, si en dicho recorrido se encuentra un número que es divisible por el parámetro **num** no modificar dicho número, de lo contrario sustituirlo por **CERO**. La función debe comportarse de la siguiente manera:

$$\text{Matriz1} = \begin{bmatrix} 1 & 5 & 3 & 7 \\ 2 & 43 & 6 & 8 \\ 11 & 23 & 3 & 4 \\ 7 & 8 & 9 & 10 \end{bmatrix}$$

$$\text{Resultado} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 2 & 0 & 6 & 8 \\ 0 & 0 & 0 & 4 \\ 0 & 8 & 0 & 10 \end{bmatrix}$$

```
>>> matriz = [[1,5,3,7], [2,43,6,8], [11,23,3,4], [7,8,9,10]]  
>>> encontrarNumerosDivisibles (matriz, 2)  
[[0,0,0,0], [2,0,6,8], [0,0,0,4], [0,8,0,10]]
```

Las listas no deben ser vacías y deben ser números enteros

Hacer una implementación, haciendo uso del **While** o **For**

Use el método de resolución de problemas visto en clases para encontrar una solución. **Escriba las entradas, salidas, restricciones del problema de una forma técnica**. Implemente la solución en Python. Documente propiamente su código y use nombres de variables significativos, y utilice **validación de argumentos** para hacer su función más robusta. Hacer uso de la cláusula **CONTINUE**.

Ejercicio #2 (30 puntos)

Escriba una función **convertirBinarioDecimal(numero)** que reciba un parámetro denominado "numero", el cual es una representación binaria y deberá convertir a su representación con base

a 10. La función debe comportarse de la siguiente manera:

```
>>> convertirBinarioDecimal (1100)
12
>>> convertirBinarioDecimal (1108)
'No corresponde a una representación binaria'
>>> convertirBinarioDecimal (0)
0
```

Ejercicio #3 (30 puntos)

Escriba una función **convertirDecimalHex(numero)** que reciba un parámetro denominado “numero”, el cual es una representación en base 10 y deberá convertir a su representación con base a 16. La función debe comportarse de la siguiente manera:

```
>>> convertirDecimalHex (16)
10
>>> convertirDecimalHex (31)
1F
>>> convertirDecimalHex (7000)
1B58
```

Enlace de referencia: <https://cual-es-mi-ip.online/herramientas/conversores-numericos/conversor-decimal-a-hexadecimal/>