## **Desafío 1**

2025-1

Andrés Eduardo Santos Ramos

Cesar Luis Álvarez Sánchez

## 

## 

## 

## 

## 

## 

## 

## 

## **Introducción**

El presente informe tiene como objetivo documentar el análisis y desarrollo de un sistema capaz de reconstruir una imagen original que ha sido transformada mediante operaciones a nivel de bits y enmascaramientos. El ejercicio corresponde a una aplicación real en seguridad de la información, donde una imagen es modificada deliberadamente como una forma de cifrado o protección. El sistema debe ser capaz de revertir estas transformaciones utilizando pistas parciales, tales como archivos de suma enmascarada y una imagen auxiliar generada aleatoriamente.

## **Descripción del Problema**

Se proporciona una imagen final transformada (I\_M.bmp) y una máscara (M.bmp). Además, se entregan archivos de texto M1.txt, M2.txt, ..., cada uno de los cuales contiene:

* Una semilla de desplazamiento s.
* Un conjunto de valores RGB resultantes de aplicar:

S(k) = ID(k + s) + M(k)

Donde ID es una versión intermedia de la imagen antes del enmascaramiento, y M es la máscara de la misma dimensión. El resultado es la suma canal por canal (R, G, B), cuyos valores están guardados como tripletas de enteros en los archivos .txt.

## **Plan de Solución**

1. Cargar las dos imágenes (I\_M, M) y los .txt proporcionados con QT creator.
2. Extraer los píxeles de la imagen distorsionada para su posterior utilización.
3. Leer y operar en los archivos .txt para obtener la ubicación de la máscara.
4. Aplicar una de las transformaciones en toda la imagen y luego enmascaramiento en una zona específica para hacer la verificación.
5. Comparar el enmascaramiento con los datos proporcionados en el .txt para verificar que la transformación realizada fue la correcta.
6. Si fue correcta, Repetir el proceso para cada archivo .txt desde el último hasta el primero.

.

## **Funciones Relevantes a utilizar**

### **1. Desplazamiento de Bits**

* **Entradas:** uint8\_t value, int bits
* **Salida:** Realiza un desplazamiento lógico de bits a izquierda o derecha. (de 1 a 8 bits)

### **2. XOR entre Píxeles RGB**

* **Entradas:** pixeles1, pixeles2
* **Salida:** Aplica una operación XOR bit a bit entre cada canal RGB

### **3. Rotación de Bits a la Izquierda**

* **Entradas:** uint8\_t value,
* **Salida:** uint8\_t rotado circularmente a la izquierda en bits

### **4. Rotación de Bits a la Derecha**

* **Entradas:** uint8\_t value
* **Salida:** uint8\_t rotado a la derecha en bits

## **6. Resultados Esperados**

* Una imagen reconstruida con alta fidelidad respecto a la original.
* Validación visual y numérica del éxito del proceso.
* Comprensión de los efectos de operaciones a nivel de bits sobre la información de una imagen.
* Un Listado con el paso a paso de operaciones a nivel de bits para reconstruir la imagen

## **7. Conclusiones**

Este proyecto permite estudiar de forma práctica cómo ciertas operaciones a nivel de bit pueden servir como mecanismos de transformación, ocultamiento o cifrado de información. La posibilidad de revertirlas correctamente depende de conocer y aplicar correctamente cada etapa del proceso, validando tanto la teoría como la programación paso a paso.