

Informe Desafío 1

Andrés Felipe Sepúlveda Rivillas

71.363.392

a) Análisis del problema y consideraciones para la alternativa de solución propuesta.

El problema propuesto para el Desafío 1 del curso de Informática II, se basa en una imagen BMP (Bitmap), la cual es un formato gráfico para almacenar imágenes digitales, la cual permite un acceso directo y rápido a cada pixel. En este caso, cada pixel está representado mediante valores de color en forma de tripletas RGB (red, green, blue) con un tamaño de 3 bytes.

Por otro lado, se tendrá una imagen distorsionada a través de transformaciones a nivel de bits, que pueden incluir desplazamientos, rotaciones y operaciones XOR con una imagen de distorsión aleatoria, todas estas transformaciones se aplican en un orden que se desconoce para el programador. Tras cada una de las etapas de transformación, se emplea una técnica de enmascaramiento que consiste en suma una porción de la imagen distorsionada con una imagen de color, esta porción de la imagen se determina a partir de una posición inicial, denominada semilla, la cual está contenida en los archivos de rastreo, mediante la suma:

$$S(k) = ID(k + s) + M(k) \text{ para } 0 \leq k < i \times j \times 3$$

Los archivos .TXT, contienen además de la semilla, información resultante del enmascaramiento realizado, determinado por conjuntos de valores enteros que representan la suma de los canales RGB, píxel a píxel.

El objetivo del desafío, es aplicar Ingeniería inversa para recuperar la imagen original que fue distorsionada mediante las transformaciones y las máscaras, usando los temas vistos en el curso, como son estructuras de control, operaciones a nivel de bits, punteros, arreglos dinámicos y funciones en C++.

Para abordar la solución al problema planteado, se tienen a disposición tres códigos elaborados en C++ con los cuales se puede acceder a los valores de los píxeles de una imagen BMP, abrir los archivos que contienen el enmascaramiento y exportar la información contenida en los arreglos dinámicos como imágenes. Mediante estos códigos y las funciones adicionales que se desarrollen, se podrá cumplir con el objetivo propuestos en el desafío, como los son:

- Función para revertir el enmascaramiento.
- Función para realizar operaciones a nivel de bit.
- Función para realizar la operación XOR con la imagen, en caso de ser necesario.

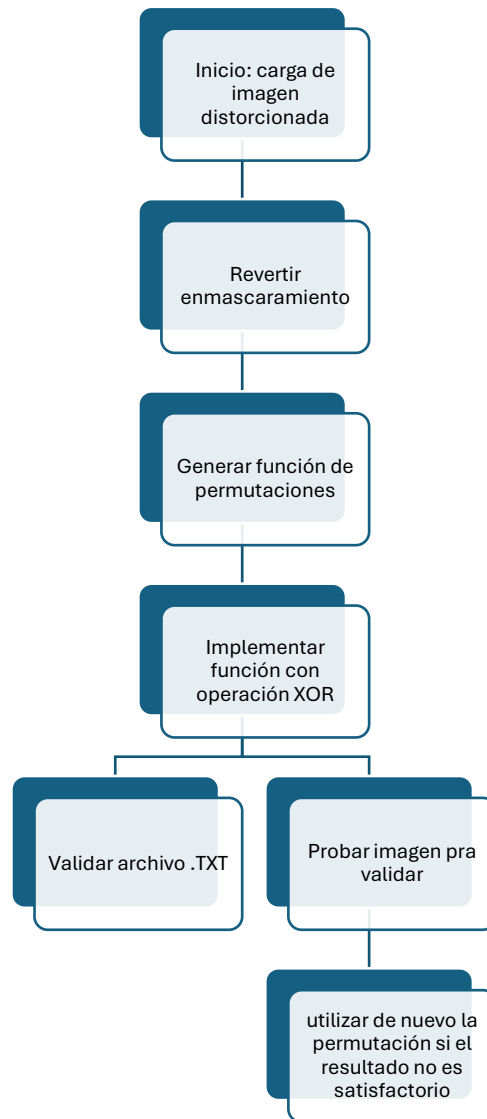
b) Esquema donde describa las tareas que usted definió en el desarrollo de los algoritmos.

Para el desarrollo del algoritmo, de manera inicial, ya que apenas estoy haciendo una aproximación al problema, considero que se pueden definir varias etapas, las cuales tendrán inmersas ciertas tareas para llegar a la solución del desafío, las etapas con sus tareas se describen a continuación:

1. Cargar las imágenes y los datos iniciales:
 - i. Cargar la imagen transformada I_D .
 - ii. Cargar la imagen que sirve de máscara M .
 - iii. Cargar la imagen para las operaciones XOR I_M .
 - iv. Cargar los archivos de rastreo.
2. Revertir el enmascaramiento y generar las permutaciones necesarias para determinar qué tipo de transformaciones se realizaron a la imagen:
 - i. Crear la función para revertir el enmascaramiento.
 - ii. Generar la función que permita aplicar las permutaciones necesarias para determinar las transformaciones, teniendo en cuenta los tipos de operaciones a nivel de bits que se pueden usar.
3. Realizar las pruebas y visualizar las imágenes de salida:
 - i. Validar que las operaciones de permutación realizadas resulten correctas, mediante la comparación con los archivos .TXT de entrada.
 - ii. Exportar la imagen generada luego de revertir el enmascaramiento y realizar las permutaciones, para validar visualmente el resultado.

c) Algoritmos implementados.

Según las consideraciones hechas en el ítem anterior, puedo plantear el siguiente esquema básico para implementar la solución del problema, al igual que en el punto anterior, es algo inicial que muy seguramente tendrá variaciones en la medida que pueda realizar un análisis más profundo del problema.



d) Problemas de desarrollo que afrontó.

1. Gestión de la memoria: debido al uso de los arreglos dinámicos y los punteros, se debe tener especial cuidado al momento de hacer la reserva de la memoria y especialmente en la correcta liberación de esta.
2. Se debe verificar el tamaño de las imágenes para garantizar que los tamaños de la imagen distorsionada y la imagen máscara para las operaciones XOR coincida, por otro lado, se debe garantizar que el tamaño de la imagen utilizada en el proceso de enmascaramiento sea igual o menor a la imagen distorsionada.
3. El tema de la validación al momento de realizar las operaciones de permutación se complica un poco, ya que de manera visual no se puede determinar si el proceso realizado es correcto o se encamina a la solución.
4. Se debe controlar de manera efectiva la aparición de valores negativos al momento de realizar las operaciones que revierten el enmascaramiento.

e) Evolución de la solución y consideraciones para tener en cuenta en la implementación.

Se irá diligenciando a medida que se realicen las funciones y se encuentren problemas en la implementación.