

Análisis y diseño de la solución del problema

En este desafío, se propone desarrollar en C++ para rescatar una foto BMP (IO) de una foto final (ID) que ha sufrido alteraciones en sus bits. El problema está en hacer un proceso inverso, ya que no se sabe qué cambios se hicieron primero.

La idea es usar los cambios al revés, pero en el orden justo para poder hallar la imagen de antes. Para resolver el problema voy a seguir estos pasos:

1. Cargar imagen ID: Es lo que queda al final, después de todo. Voy a arrancar con esta porque es lo único que se ve del cambio. Tiene las alteraciones, así que es por donde voy a empezar a hacer el proceso inverso.

2. Cargar imagen IM: Como el XOR es reversible, contar con IM permite deshacer ese paso

3. Tomar la máscara M: El enmascaramiento usa valores de M para alterar los bits de la imagen

4. Leer los archivos .txt que tienen las huellas de los cambios usados. Tienen datos clave sobre el orden y el tipo de cambios usados. A pesar de que no siempre están claros, ayudan a rehacer el flujo del proceso.

5. Debo implementar funciones para cada transformación: XOR, rotación, desplazamiento, enmascaramiento.

6. Verificar la forma del enmascaramiento y revertirlo

Estudiar cómo se hizo enmascaramiento y qué zona de la imagen esta más afectada. Esto me puede ayudar a saber si hay que revertir primero ese cambio.

7. Deducir el orden de transformación: Como no sé el orden real de las operaciones aplicadas a la imagen, debo hacer pruebas con diferentes combinaciones de operaciones inversas. Hay que ver si la imagen se ve mejor, lo que significa que casi voy por buen camino.

8. Volver a la imagen original

Esta resolución de este desafío no solo implica programar las transformaciones bit a bit, sino entender cómo revertir los procesos complejos sin conocer el orden en el que fueron aplicados. Permite fortalecer habilidades de análisis, manipulación de datos a bajo nivel y razonamiento lógico. Además, se trabajan conceptos clave como el manejo de imágenes.

Diagrama del proceso

