1. El desafío es un ejercicio de ingeniería inversa que requiere reconstruir un mensaje original que ha sido sometido a dos procesos secuenciales: compresión y encriptación. La única información que nos proporciona el cliente es un mensaje encriptado y comprimido, junto con un pequeño fragmento del texto original en texto plano.

El objetivo principal es identificar el método de compresión (RLE o LZ78), los parámetros de encriptación (rotación de bits n y clave K), y luego utilizar esta información para desencriptar y descomprimir el mensaje completo.

La clave para resolver este problema radica en el fragmento de texto original conocido. Al ser un fragmento en texto plano, podemos asumir que su versión encriptada se encuentra dentro del mensaje comprimido y encriptado. La estrategia propuesta se centrará en probar sistemáticamente diferentes combinaciones de los parámetros de encriptación hasta que la versión desencriptada de un segmento del mensaje encriptado coincida con el fragmento conocido

1. El desarrollo se ha dividido en las siguientes tareas principales, organizadas para abordar el problema de forma metódica:
2. **Módulo de Desencriptación:**

* Implementar la función de desencriptación que toma un byte, una clave K y un valor de rotación n.
* Esta función aplicará primero el XOR con la clave y luego la rotación de bits a la derecha.

1. **Módulo de Detección de Parámetros:**

* Iterar a través de todos los valores posibles de la clave K (0 a 255) y la rotación n (1 a 7).
* Para cada combinación (n,K), desencriptar el mensaje completo.
* En cada iteración, buscar el fragmento de texto original conocido dentro del mensaje desencriptado.
* Si se encuentra una coincidencia, se han identificado los parámetros correctos.

1. **Módulo de Descompresión:**

* Implementar el algoritmo de descompresión para **RLE**. Este algoritmo debe leer el número de repeticiones y luego el carácter, repitiendo el carácter la cantidad de veces indicada.
* Implementar el algoritmo de descompresión para **LZ78**. Este debe construir un diccionario dinámicamente a medida que lee los pares (índice, carácter) y reconstruye las cadenas correspondientes.

1. **Módulo de Integración y Recuperación Final:**

* Una vez que se hayan encontrado los parámetros (n,K), se desencripta el mensaje completo.
* Se probarán ambos algoritmos de descompresión (RLE y LZ78) en el mensaje desencriptado.
* El método que resulte en un mensaje de texto plano coherente será el correcto.
* El resultado final será el mensaje original, los parámetros de compresión y los de encriptación.

1. Se implementarán dos conjuntos de algoritmos, cada uno con su contraparte inversa:
2. **Algoritmos de Encriptación/Desencriptación:**

* **Función desencriptar\_byte(byte, n, K):** Aplica byte XOR K y luego rota el resultado n bits a la derecha. Se implementará a nivel de bits utilizando operaciones como >> y << para las rotaciones, asegurando que los bits que salen por un lado entren por el otro.
* **Función rotacion\_derecha(byte, n):** Esta función auxiliar implementará la rotación circular de bits.

1. **Algoritmos de Compresión/Descompresión:**

* **RLE:** La implementación de descompresión de RLE leerá una secuencia de bytes. Si un byte representa un número, lo usará para determinar cuántas veces debe repetir el siguiente byte (el carácter).
* **LZ78:** La implementación de descompresión de LZ78 construirá un diccionario en memoria dinámica (un arreglo de punteros a caracteres). Leerá pares (índice, carácter) del mensaje, reconstruirá la cadena correspondiente del diccionario y la añadirá a la salida y al diccionario como una nueva entrada.

1. Problemas a afrontar:

* **Manejo de memoria:** Implica que todo el manejo de cadenas y del diccionario LZ78 debe realizarse manualmente con punteros y arreglos de caracteres. Esto requiere una cuidadosa gestión de la memoria dinámica para evitar fugas de memoria y errores de acceso.

**Eficiencia en el código:** Un pilar fundamental en el desarrollo de este proyecto es la eficiencia, tanto en el rendimiento computacional como en el uso de la memoria. Dado que el problema implica un proceso de fuerza bruta, la eficiencia no será solo una buena práctica, sino un requisito crítico para que la solución sea viable en un tiempo razonable.