



E1. Actividad Integradora 1

Integrantes

Juan Diego Susunaga Velásquez - A01764884

Manuel Alejandro Cruz Valladares - A00836441

Adrián Salazar Rodríguez - A00835314

Alan Sanmiguel Garay - A00837404

Campus Monterrey

Análisis y diseño de algoritmos avanzados (Gpo 602)

Profesor

Felipe Castillo Rendón

Fecha de entrega

6 de abril, 2025

1. Búsqueda de Subcadenas (String Matching)

Inicialmente, decidimos utilizar una búsqueda básica de subcadenas, ya que era una forma directa de abordar el problema. Sin embargo, a medida que las transmisiones crecían en tamaño, nos dimos cuenta de que este enfoque no era lo suficientemente eficiente. Por eso, optamos por implementar el **algoritmo KMP (Knuth-Morris-Pratt)**. Este algoritmo es más rápido porque evita comparaciones innecesarias al aprovechar información sobre coincidencias previas, lo que mejora el rendimiento al analizar transmisiones grandes.

2. Detección de Palíndromos

Para encontrar palíndromos, comenzamos utilizando un enfoque simple basado en la expansión desde el centro, ya que era fácil de implementar. Sin embargo, a medida que el tamaño de las cadenas aumentaba, nos dimos cuenta de que este método no era lo más eficiente. Por eso, decidimos implementar el **algoritmo de Manacher**, que permite encontrar palíndromos en tiempo lineal ($O(n)$), lo cual fue clave para mejorar el rendimiento en cadenas más largas.

3. Subsecuencia Común Más Larga (LCS)

Al comparar las transmisiones, nos pareció adecuado utilizar el **algoritmo de Programación Dinámica para LCS (Longest Common Subsequence)**. Este algoritmo es ideal para encontrar las coincidencias más largas entre dos cadenas y detectar patrones comunes entre las transmisiones. Aunque su complejidad es $O(n * m)$, creímos que era una opción razonable dado el tamaño manejable de las transmisiones en este caso.

4. Reflexión Final sobre las Decisiones

Las decisiones sobre qué algoritmos implementar se tomaron con el objetivo de balancear eficiencia y precisión. Elegimos el KMP y Manacher para mejorar el rendimiento, especialmente con transmisiones grandes, mientras que LCS nos dio una forma confiable de comparar patrones comunes entre las cadenas. Al final, cada algoritmo se seleccionó para asegurarnos de que el sistema pudiera procesar las transmisiones de forma eficiente y precisa, sin importar el tamaño de los datos.