

Actividad Integradora 1: Reflexión individual

Estudiante: Diego Palma Rodríguez

Matrícula: A01759772

La situación problema consistió en analizar transmisiones de información e identificar si existe código malicioso en ella. Toda la información con respecto a las transmisiones se encuentra en tres archivos de texto y el código malicioso se encuentra en un archivo de texto. En base a ello se nos pidió realizar las siguientes actividades:

1. Identificar si los códigos maliciosos se encuentran en los archivos de transmisión. En caso de que sí, determinar la cantidad de incidencias y las posiciones donde se encuentran.
2. Encontrar el palíndromo más largo de los archivos de transmisión.
3. Analizar la similaridad entre los tres archivos de texto, dos a dos, determinando el substring más largo de cada una de las tres combinaciones.

Para resolver la primera actividad se aplicó el algoritmo Z, dado que es un algoritmo que se utiliza en la búsqueda de patrones en texto. Además de ello, es un algoritmo eficiente, pues tiene una complejidad de tiempo lineal. En líneas generales, el algoritmo Z funciona manteniendo un arreglo auxiliar llamado función Z. Esta función almacena la longitud de la subcadena más larga, a partir del índice actual, que también es su prefijo. Esto significa que cada índice almacena el número de caracteres que coinciden con los caracteres iniciales, a partir de este índice. Esto implica que si la función Z tiene un valor ' k ' para cualquier índice, significa que ' k ' caracteres después de este índice coinciden con los primeros ' k ' caracteres de la cadena. De este modo, al almacenar el número de coincidencias en la función Z se determina la cantidad de incidencias y las posiciones donde se encuentran los códigos maliciosos.

Para resolver la segunda actividad se implementó el algoritmo de Manacher, debido a que es utilizado para encontrar el palíndromo más largo dentro de cualquier texto. Asimismo, es uno de los algoritmos más eficientes para dicha tarea, pues tiene una complejidad temporal lineal, a comparación de otros métodos con complejidades cuadráticas e incluso cúbicas, como el método por fuerza bruta. El motivo de que este algoritmo sea más rápido que otros enfoques es que explota la idea de que un palíndromo ocurre dentro de otro palíndromo.

Finalmente, para la tercera actividad, se utilizó el algoritmo del *substring* común más largo con un enfoque de programación dinámica. El motivo de utilizar el enfoque de programación dinámica es que nos permite reducir la complejidad temporal de tal modo que sea lineal. La idea principal es almacenar la longitud de los substrings comunes en una matriz. De este modo encontraremos el substring más largo en la casilla con el mayor valor. Asimismo, almacenamos la posición de la casilla y recorremos la cantidad de casillas hacia atrás para obtener el *substring* más largo.

En conclusión, se aplicaron tres algoritmos eficientes relacionados con el análisis de texto (*strings*): algoritmo Z, algoritmo de manacher y el enfoque de programación dinámica, con el objetivo de solucionar problemas que se encuentran constantemente en aplicaciones reales: coincidencia de patrones, el palíndromo más largo y el *substring* común más largo. El desarrollo de esta actividad me permitió comprender la importancia de dichos algoritmos, así como realizar su implementación en una aplicación real.