

Análisis y Diseño de Algoritmos Avanzados

Alexia Saucedo A01283646
Diego Palma A01759772
Ivan Delfín A01024265



Actividad Integradora #1





Encontrar código malicioso dentro del proceso de Transmisión de Datos entre dispositivos.

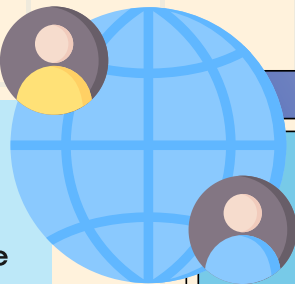
ALGORITMOS UTILIZADOS	Descripción del algoritmo	Complejidad Temporal
Z-function	<ul style="list-style-type: none">Algoritmo utilizado para búsqueda de patrones de texto.Al utilizar la función Z se determina la cantidad de incidencias y las posiciones donde se encuentran los códigos maliciosos.	$O(n)$
Algoritmo de Manacher	<ul style="list-style-type: none">Algoritmo que se emplea para encontrar el palíndromo más largo dentro de cualquier texto.Suele ser más rápido que otros algoritmos debido a que explota la idea de que un palíndromo ocurre dentro de otro palíndromo.	$O(n)$
Algoritmo del substring común más largo	<ul style="list-style-type: none">Utiliza el enfoque de programación dinámica para disminuir su complejidad.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.	$O(n)$

Reflexión

El desarrollo de esta actividad nos permitió comprender la importancia de dichos algoritmos, así como realizar su implementación en una aplicación real. Además de ello, nos brindó **nuevos enfoques para poder diseñar algoritmos**, como la creación de nuevas funciones para solucionar problemas específicos, el enfoque de programación dinámica o el hecho de entender el contexto del problema para usarlo a nuestro favor (Manacher).



Debido a la Pandemia aumento la necesidad del servicio de internet en nuestras casas. Los proveedores deben ser capaces de implementar mejoras en su servicio.



Actividad Integradora #2

ALGORITMOS UTILIZADOS	Descripción del algoritmo	Complejidad Temporal
Algoritmo Kruskal	<ul style="list-style-type: none">Se utiliza comúnmente para solucionar el problema del árbol de mínima expansión.Dado que se modela las colonias como un grafo, el algoritmo es utilizado para encontrar la forma más óptima de cablear todo las colonias.	$O(\log^2 E)$
TSP Branch and Bound	<ul style="list-style-type: none">Algoritmo que muestra la ruta óptima entre colonias no centrales, pasando solo 1 vez por cada colonia hasta regresar al lugar de origen (Ciclo Hamiltoniano).Esta técnica consiste básicamente en elaborar un árbol y recorrerlo en anchura, almacenando los costos en cada nodo. Finalmente, seleccionar el nodo con el menor costo real, donde la ruta óptima corresponde a los padres de dicho nodo.	$O(2^n)$
Floyd-Warshall	<ul style="list-style-type: none">Este algoritmo soluciona el problema de encontrar el camino más corto entre todas las colonias centrales, las cuales se modelan como los nodos de un grafo.Este algoritmo utiliza la técnica de programación dinámica, donde el valor de cada celda corresponde a la ruta menos costosa entre el nodo actual y el nodo adyacente de forma directa o utilizando un nodo intermediario.	$O(n^3)$
Función para punto más cercano	<ul style="list-style-type: none">Para mostrar cual colonia es conveniente conectar con los nuevos puntos.Se utilizaron las coordenadas para obtener las distancias entre todos los puntos y la nueva colonia, de ahí se escogió la mínima de todas esas distancias.	$O(n^2)$

Reflexión

- La actividad nos sirvió para aprender a cómo **seleccionar las estructuras de datos** adecuadas para solucionar los problemas de forma eficiente. Utilizando grafos para el modelo de conexiones y utilizar diferente estructura de datos para disminuir la complejidad en Kruskal.
- Nos permitió **analizar a profundidad algoritmos** que se utilizan frecuentemente en la solución de diversos problemas reales, como el problema del viajero, el problema del árbol de mínima expansión y el problema del camino más corto.
- Tener en cuenta **la complejidad temporal** de los algoritmos, pues los casos de prueba contenían miles de datos y esto hacía que la ejecución de los algoritmos sea retardada.