

Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos Fundamentales (Evidencia Competencia)

El canal de Suez es un canal navegable que conecta el mar Mediterráneo con el mar Rojo a través de alrededor de 190 km, uniendo por un el lado del mar Mediterráneo desde el puerto Said hasta el golfo de Suez en el mar Rojo. Este canal es navegado entre 49 y 97 barcos diariamente. Se tiene un registro de los barcos que navegan por el canal con el siguiente formato:

<fecha> <hora> <punto-entrada> <UBI-Identificador único del buque>

En este primer entregable, se realizó un código capaz de leer un archivo que contenga los registros de los barcos que navegan por el canal, para después ordenarlos y pedirle al usuario una serie a buscar, es este caso pedirá los primeros tres caracteres del UBI para al final, desplegar todos los registros que coincidan de manera ordenada.

Algoritmos de búsqueda y ordenamiento utilizados

Merge sort: este algoritmo es para ordenar los elementos que se le pasen, consiste en dividir en dos los datos para proceder a acomodarlos en cada mitad y al final, junta las dos mitades de manera ordenada.

Su complejidad temporal es de $O(n \log n)$ ya que consiste en una función auxiliar, en este caso *unir*, al multiplicar la complejidad de cada una de las partes, nos da de resultante $O(n \log n)$

Búsqueda binaria: Este algoritmo es utilizado para buscar un valor dentro de un array o vector, este utiliza los datos ordenados para después, *partir* el vector, una vez partido ve si el valor que se busca está en la primera o la segunda mitad, corroborando esto, hace que la parte en la que el dato se encuentra sea el nuevo vector, para repetir este proceso hasta encontrar el número.

Su complejidad temporal es de $O(\log n)$ en el peor de los casos ya que va particionando la entrada en dos, terminando con un valor menor a n , en el mejor de los casos este algoritmo presentará una complejidad de $O(1)$, que es cuando el valor que está en el centro es el valor que se busca.

Reflexión

Es importante el uso de distintos métodos de ordenamiento y búsqueda ya que hay algunos que se adaptan mejor al caso, por ejemplo, en esta actividad, se utilizó la

búsqueda binaria ya que los datos eran bastantes, y de esta forma el tiempo utilizado por el algoritmo sería menor que con una secuencial.

Por ejemplo, si tuviéramos un array de 100 números del 1 al 100 y pedimos que nos devuelva el index del número 100, en búsqueda secuencial tardaría mucho mas que en binaria.

Al igual que con los algoritmos de búsqueda, tenemos que encontrar un algoritmo de ordenamiento que satisfaga nuestras necesidades, en este caso se utilizó el merge sort, pero dependiendo de lo que tengamos que hacer y como tengamos que manipular los datos, podríamos utilizar distintos métodos.