



**Tecnológico  
de Monterrey**

## **Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos Fundamentales**

Programación de estructuras de datos y algoritmos  
fundamentales

Grupo 12

24.09.2021

### **Alumno**

Fernando Cerriteño Magaña - A01702790

### **Profesor**

Dr. Eduardo Arturo Rodríguez Tello

## Reflexión

La era digital de la cual formamos parte hoy en día es una era en la cual se puede tener información almacenada de forma digital, permitiendo tener toda la información sin ocupar espacio físico, permitiendo tener lo que se le conoce como una base de datos. No obstante, es fundamental que esa base de datos este organizada, de algún modo (Por orden alfabético, por fecha de creación, de menor a mayor, etc.), con la finalidad de poder generar una búsqueda de archivos mas eficientes al momento de querer obtener algún registro o archivo de la forma más efectiva posible. Es debido a esto que existen diferentes algoritmos de ordenamiento, para que en vez de que se tenga que realizar de forma manual, pueda ser automatizado por un algoritmo de forma automática, así mismo existen los algoritmos de búsqueda, los cuales, como su nombre indica, sirven para poder realizar una búsqueda en una lista para obtener el archivo, registro, etc., que se necesite encontrar.

En una situación como la que se nos presenta en la actividad 1.3, es de vital importancia tener un orden en los registros para poder darse una idea de que el periodo de tiempo con el que se está trabajando, así como del poder darse una idea del donde se encuentra algún registro en específico. Es por estas razones, por las que se nos pide lograr ordenar la bitácora presentada, para posteriormente, poder realizar una búsqueda binaria, y encontrar un rango de registros.

Para la primera etapa en la resolución de la problemática, se utilizó la librería “sstream” para poder leer el archivo “bitácora.txt”, posteriormente, se creo la clase “Entrada.h” para poder almacenar todos los datos de cada registro de la bitácora en un vector. Una vez que se obtuvo el vector, se realizo un algoritmo de ordenamiento Merge, esto es debido a que el algoritmo, es fácil de implementar, solo se tendría que realizar una sobrecarga de operadores en la clase Entrada, y finalmente, el algoritmo cuenta con una complejidad temporal de  $O(n \log n)$ , esto significa que se puede realizar el ordenamiento de una manera más rápida y eficaz. Después de obtener el algoritmo ordenado, se procede a crear un archivo de texto llamado “bitácora\_ordenada.txt”, el cual contiene la bitácora ordenada.

Para la segunda etapa en la actividad, se realizó una búsqueda binaria con los datos de la bitácora ordenada, esto con el fin de poder obtener los registros entre dos fechas dadas por el usuario. La razón por la que se utilizó el algoritmo de búsqueda binaria, fue debido a que era un requisito en la rúbrica, y debido a que cuenta con una complejidad temporal  $O(\log_2(n))$ , permitiéndolo ser un algoritmo bastante efectivo al momento de realizar una búsqueda en un arreglo ordenado. Finalmente, se imprime en la consola el algoritmo de forma ordenada.

La realización de esta actividad me ha ayudado para poder mejorar en; la creación e implementación de objetos en un archivo main, así como el poder leer y crear un archivo de texto en el código, y finalmente en la lectura de la complejidad temporal de algunos algoritmos.