

REFLEXION ACTIVIDAD 1.3

Alejandra Cabrera Ruiz A01704463

Un algoritmo de ordenamiento nos permite organizar una serie de datos según un criterio definido ya sea número o alfabético o por fecha. Según Gómez (2022) estos algoritmos se pueden dividir en diferentes tipos de acuerdo a su funcionamiento y eficiencia:

· Ordenamiento por selección: basa su ejecución en que cada vez que se mueve un elemento, se lleva a su posición correcta.

- Ordenamiento por inserción: conocido como el método de la baraja por ser el método habitual de los jugadores de carta.
- Ordenamiento por intercambio directo (burbuja): comparar pares de elementos contiguos y se intercambian si deben ordenarse.
- Ordenamiento Shell1: es una versión mejorada del método de inserción directa.
- Ordenamiento por montículos: conocido por Heapsort, es el método mas eficiente de los métodos de ordenación que trabaja con árboles.
- Ordenamiento rápido: (Quicksort en inglés), basado en la técnica "divide y vencerás", que permite ordenar n elementos en un tiempo proporcional a $n \log n$.

(Gómez, 2022)

Referencias

Gómez, N. (2022). FAEDIS. Umng.edu.co. http://virtual.umng.edu.co/distancia/ecosistema/odin/odin_desktop.php?path=Li4vb3Zhecy9pbmdlbmlcmHx2luZm9ybWFOaWNhL2VzdHJ1Y3R1cmFfZGVfZGF0b3MvdW5pZGFkXzEv#slide_1

Reflexión:

A mi forma de ver el estudio de los algoritmos de ordenamiento nos permite identificar que algoritmo usar para desarrollar un programa, yo empecé usando bubble sort y terminé usando insertion sort porque era más eficiente y se corría más rápido en la consola debido a la complejidad del mismo. Esta actividad me sirvió para darme cuenta de la utilidad de usar Shell para probar el código y el uso de archivos txt al fin me quedo completamente claro. Aprendí sobre nuevas funciones como `getline` y `.substr()` que fueron esenciales para la implementación de mi código.