



Programación de estructura de datos y algoritmos fundamentales

Actividad 1.3 Actividad Integral, Conceptos básicos y algoritmos fundamentales

Sebastian Antonio Almanza A01749694

Tecnológico de Monterrey

16 de Enero del 2024

Grupo 570

Profesor: Eduardo Arturo Rodriguez Tello

Gracias a la tecnología podemos realizar cosas increíblemente rápido y eficientemente; hoy por hoy si tenemos alguna duda la buscamos en Google y obtenemos la información que requerimos de manera ordenada. Pero para lograr todo esto es necesario tener conocimientos de algoritmos de búsqueda y de ordenamiento. Si bien es cierto existen muchos algoritmos para ordenar datos, siempre se busca el más eficiente. Por ello, decidimos implementar el Merge Sort y el Bubble Sort.

Binary Search

La búsqueda binaria es un algoritmo eficiente para encontrar un dato en un arreglo ordenado. Este algoritmo divide el arreglo buscando en qué mitad está el elemento a buscar y repite ese proceso de manera recursiva hasta que halla el elemento buscado. La complejidad del algoritmo es : $O(\log_2 n)$.

Merge Sort

Merge Sort es uno de los algoritmos de ordenamiento más eficientes. Está basado en el principio de divide y vencerás, el cual toma la estructura de datos a ordenar y la divide en sub listas para poder ordenarlas para tener como resultado el ordenamiento de los datos. El merge sort tiene la misma complejidad la cual es $O(n \log n)$.

Bubble Sort

El ordenamiento de BubbleSort es un método comúnmente usado para la organización de elementos en un arreglo, vector, etc. Este ordenamiento hace intercambios entre los datos, es decir los datos más pequeños los deja en el lado izquierdo intercambiando los datos más grandes. El mejor de los casos en este algoritmo es $O(n)$ y el caso promedio es $O(n^2)$.

Comparación de Swaps y comparaciones de los algoritmos

Comparaciones y swaps de bubbleSort:

```
Num de comparaciones de bubbleSort: 141346891
Num de swaps de bubbleSort: 70267928
```

Comparaciones de MergeSort:

```
Num de comparaciones: 214671
Ingrese la fecha de inicio en formato (mes, dia, hh:mm:ss)
```

Al ordenar mediante dos algoritmos de ordenamiento diferentes (en este caso mergeSort y bubbleSort) se puede observar que el número de comparaciones efectuadas en el merge es mucho menor al del algoritmo bubbleSort, en términos de swaps no podemos compararlos ya que el algoritmo de merge no realiza swaps entre los elementos para realizar el ordenamiento. Durante las ejecuciones del código también notamos que el incluir el algoritmo de bubbleSort al mismo tiempo que el de mergeSort (o incluso por sí mismo en algunas ocasiones) hacía que se produjera el error de Segmentation fault ya que se consume más memoria de la permitida. En conclusión el algoritmo más eficiente es el de mergeSort ya que ordena de

forma rápida los elementos del archivo ya que no realiza muchas comparaciones y es eficiente a nivel de memoria en la computadora.

Link a repositorio de github: <https://github.com/A01749694/TC1031-portafolio-final>

Referencias

Binary Search. (s. f.). Khaan Academy.

<https://www.khanacademy.org/computing/computer-science/algorithms/binary-search/a/binary-search#:~:text=Binary%20search%20is%20an%20efficient,possible%20locations%20to%20just%20one>.

Pankaj. (2022, 3 agosto). *Merge Sort Algorithm - Java, C, and Python implementation*.

DigitalOcean.

<https://www.digitalocean.com/community/tutorials/merge-sort-algorithm-java-c-python>

Rivel_co. (2020, 4 enero). Bubble sort. *Include Poetry*.

<https://www.include-poetry.com/Code/C++/Metodos/Ordenamientos/Bubble-sort/>

Rodriguez, E. (2024, 9 enero). *Algoritmos de busqueda* [Diapositivas].

https://tecmx-my.sharepoint.com/personal/ertello_tec_mx/_layouts/15/onedrive.aspx?ga=1&id=%2Fpersonal%2Fertello%5Ftec%5Fmx%2FDocuments%2FVideos%20unidades%20de%20formacion%2FVideos%2DTC1031%2DGpo570%2DInvierno2024%2FMateriales%2Fsesion05a%2DAlgoritmosDeBusqueda%2Epdf&parent=%2Fpersonal%2Fertello%5Ftec%5Fmx%2FDocuments%2FVideos%20unidades%20de%20formacion%2FVideos%2DTC1031%2DGpo570%2DInvierno2024%2FMateriales

Rodríguez, E. (2024, 11 enero). *Algoritmos de ordenamiento* [Diapositivas].

https://tecmx-my.sharepoint.com/personal/ertello_tec_mx/_layouts/15/onedrive.aspx?ga=1&id=%2Fpersonal%2Fertello%5Ftec%5Fmx%2FDocuments%2FVideos%20unidades%20de%20formacion%2FVideos%2DTC1031%2DGpo570%2DInvierno2024%2FMateriales

dades%20de%20formacion%2FVideos%2DTC1031%2DGpo570%2DInvierno2024%
2F Materiales%2F sesion07a%2D Algoritmos Ordenamiento%2D Parte03%2E pdf&pare
nt=%2F personal%2F tello%5F tec%5F mx%2F Documents%2F Videos%20 unidades
%20 de%20 formacion%2F Videos%2D TC1031%2D Gpo570%2D Invierno2024%2F M
ateriales