

Reflexión Act 1.3 - Actividad Integral de Conceptos Básicos y Algoritmos Fundamentales

Didier Aarón Ricardo Hernández Ferreira
A01663817

Se implementaron dos algoritmos: el algoritmo de burbuja y el algoritmo quicksort. La elección entre estos dos se basó en la comprensión de sus complejidades temporales y su impacto en el rendimiento.

El algoritmo de burbuja revela una complejidad cuadrática $O(n^2)$. Esto significa que su eficiencia disminuye cuadráticamente a medida que aumenta el tamaño del conjunto de datos. La justificación detrás de su implementación radica en situaciones donde la simplicidad y claridad del código superan las preocupaciones de rendimiento, y el tamaño del conjunto de datos es relativamente pequeño.

Por otro lado, el algoritmo mergesort fue la elección preferida para conjuntos de datos más extensos. Su complejidad promedio $O(n \log n)$ proporciona un rendimiento superior en comparación con el algoritmo de burbuja. Mergesort se destaca por su capacidad para ordenar grandes conjuntos de datos de manera eficiente, gracias a su enfoque de divide y vencerás.

En cuanto a la búsqueda de registros dentro de un rango de fechas, se emplearon algoritmos de búsqueda binaria. Estos algoritmos se eligieron por su complejidad temporal reducida de $O(\log n)$ en listas ordenadas.

La justificación detrás de la elección de la búsqueda binaria radica en su eficiencia para grandes conjuntos de datos. La capacidad de dividir repetidamente el conjunto de datos en mitades, descartando segmentos enteros en cada paso, garantiza un rendimiento logarítmico incluso en escenarios de datos masivos.

Esta selección informada de algoritmos es esencial para optimizar el rendimiento y responder de manera efectiva a los desafíos de procesar grandes cantidades de datos temporales en aplicaciones como la bitácora.

Referencias:

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms

(3rd ed.). Disponible en línea: <https://mitpress.mit.edu/books/introduction-algorithms>

Bubble Sort Data Structure and Algorithm Tutorials. (2014, February 2). GeeksforGeeks;

GeeksforGeeks. <https://www.geeksforgeeks.org/bubble-sort/>

Merge sort en Java. (2020). Platzi.

<https://platzi.com/tutoriales/1469-algoritmos-practico/4260-merge-sort-en-java/>

Khan Academy. (2023). Khanacademy.org.

<https://es.khanacademy.org/computing/computer-science/algorithms/binary-search/a/binary-search#:~:text=La%20b%C3%BAsqueda%20binaria%20es%20un,ubicaciones%20posibles%20a%20solo%20una.>

Los algoritmos de ordenamiento y búsqueda elegidos en los códigos presentados se han seleccionado con base en la naturaleza de las operaciones a realizar y en la eficiencia computacional para diferentes situaciones.

Primero son los algoritmos de búsqueda.

La búsqueda secuencial es una elección adecuada para listas pequeñas o no ordenadas.

La implementación `busquedaSecuencial` recorre la lista de manera lineal hasta encontrar el elemento deseado.

La búsqueda binaria es altamente eficiente para listas ordenadas. Al dividir repetidamente el espacio de búsqueda por la mitad, se logra una reducción logarítmica en el tiempo de búsqueda.

Esto es diferente para los algoritmos de ordenamiento.

Aunque el Bubble Sort es un algoritmo simple y fácil de entender, se ha elegido más con fines educativos que prácticos. Su complejidad cuadrática lo hace menos eficiente para conjuntos de datos grandes, pero su simplicidad lo hace útil para explicar conceptos básicos de ordenación.

También utilizamos el algoritmo mergesort, conocido por ser el algoritmo de mejor eficiencia por su técnica de “divide y vencerás”.

Referencias:

Andrés Mise Olivera. (2020, May 11). Algoritmos de Búsqueda y Ordenamiento - Andrés Mise Olivera - Medium. Medium; Medium.

<https://medium.com/@mise/algoritmos-de-b%C3%BAsqueda-y-ordenamiento-7116bcea03d0>

Team, S. (2023, June 18). Algoritmos de ordenación con ejemplos en C++. Swhosting.com; SW Hosting.

<https://www.swhosting.com/es/comunidad/manual/algoritmos-de-ordenacion-con-ejemplos-en-c#:~:text=Los%20algoritmos%20de%20ordenaci%C3%B3n%20son,eficiencia%20y%20optimizaci%C3%B3n%20de%20procesos>.

Calvo, J. (2022, December 7). La complejidad de los algoritmos – Blog Europeanvalley. Europeanvalley.es.

<https://www.europeanvalley.es/noticias/la-complejidad-de-los-algoritmos/#:~:text=La%20complejidad%20de%20un%20algoritmo,el%20contexto%20de%20los%20algoritmos>.

Khan Academy. (2023). Khanacademy.org.

<https://es.khanacademy.org/computing/computer-science/algorithms/binary-search/a/binary-search>