



Programación de estructuras de datos y algoritmos fundamentales TC1031

REFLEXION 1.3

Nicolas Aguirre v.

A00832772

Profesor

Dr. Eduardo Arturo Rodríguez Tello

03 de Abril de 2022

Algoritmos Fundamentales 1.3 - Reflexión

Es importante saber el correcto uso e implementación de los diferentes algoritmos de ordenamiento y búsqueda ya que para cada problema o situación se debe usar el mas correcto. Algunas veces es mas eficiente usar uno que el otro ya que los tiempos y uso de recursos pueden variar. Los algoritmos de ordenamiento son de los mas usados ya que estos hacen la manera mas sencilla que es quitando el elemento más pequeño y poniéndolo de primero y luego el segundo mas pequeño y se pone después del primero y así sucesivamente. De igual manera los algoritmos de búsqueda son muy utilizados y útiles ya que casi siempre se necesita buscar elementos o datos en un programa y pues esta es la mejor manera de hacerlo.

Para nuestro desarrollo del proyecto, estos algoritmos nos van a ayudar mucho para ordenar cierta información durante la realización de este mismo. Gracias al conocer estos algoritmos podremos determinar el más acertado y eficiente para cada situación.

Array Sorting Algorithms

Algorithm	Time Complexity			Space Complexity
	Best	Average	Worst	Worst
Quicksort	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n^2)$	$O(\log(n))$
Mergesort	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$
Timsort	$\Omega(n)$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$
Heapsort	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(1)$
Bubble Sort	$\Omega(n)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
Insertion Sort	$\Omega(n)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
Selection Sort	$\Omega(n^2)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
Tree Sort	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n^2)$	$O(n)$
Shell Sort	$\Omega(n \log(n))$	$\Theta(n(\log(n))^2)$	$O(n(\log(n))^2)$	$O(1)$
Bucket Sort	$\Omega(n+k)$	$\Theta(n+k)$	$O(n^2)$	$O(n)$
Radix Sort	$\Omega(nk)$	$\Theta(nk)$	$O(nk)$	$O(n+k)$
Counting Sort	$\Omega(n+k)$	$\Theta(n+k)$	$O(n+k)$	$O(k)$
Cubesort	$\Omega(n)$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$

Bibliografia

GeeksForGeeks. (n.d.). *Sorting Algorithms*. GeeksforGeeks. Retrieved June 20, 2022, from <https://www.geeksforgeeks.org/sorting-algorithms/>

Programiz. (n.d.). *Sorting Algorithm*. Programiz. Retrieved June 20, 2022, from <https://www.programiz.com/dsa/sorting-algorithm>