



Act 1.3 – Actividad Integral de conceptos básicos y Algoritmos Fundamentales

Lydia Delgado Uriarte

A01740532@itesm.mx

12 Septiembre 2021

Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Guadalajara

TC1031.13

Luis Ricardo Peña Llamas

Reflexión Individual

En diferentes situaciones para analizar datos necesitamos la recolección de estos datos para ayudar con la comprensión lectora y manejarlos con mayor facilidad. Los algoritmos de búsqueda y ordenamiento son ideales para este tipo de casos.

Posibles situaciones que se presenten estas problemáticas puede ser cuando tengamos las fechas, códigos postales, horarios, nombres, etc, de manera desordenada en la cual utilizaríamos un algoritmo de ordenamiento ya sea de la forma ascendente o descendente y por orden alfabético todo con la finalidad de que a la hora de consultar la nueva base de datos sea más entendible para el usuario. Cuando queremos saber datos en específico utilizamos los algoritmos de búsquedas, estos nos localizan el punto en el que se encuentra lo que buscamos devolviendo la información requerida.

En esta situación problema utilizamos los métodos de ordenamiento Bubble Sort y Merge Sort para hacer una comparación entre ambos, de esta observamos cual es el más conveniente para utilizar.

El método Merge Sort consiste en dividir en tres partes la lista, primero lo divide a la mitad y luego divide estas mitades, después de esto junta las listas ya ordenadas. El tiempo de complejidad de este método es $O(n\log(n))$. A continuación se ve una foto de la ejecución del código utilizando el método merge sort.

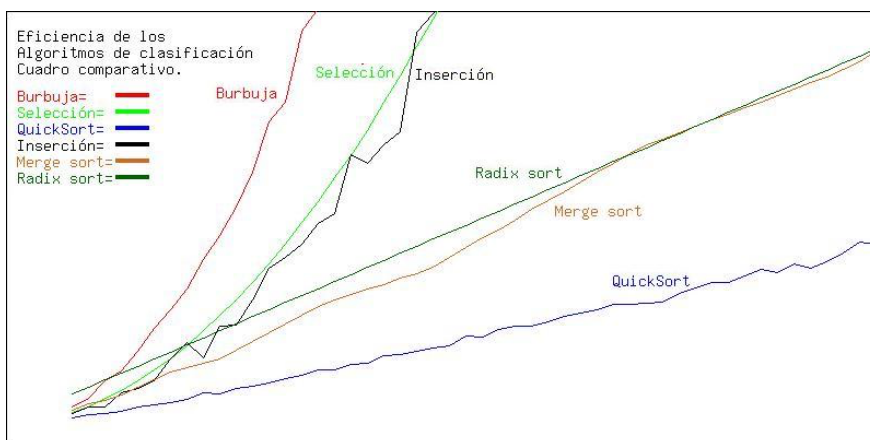
```
Loading entries...
Entries loaded (16807).

Sorting entries...
Entries sorted
Time difference (sec) = 0.184
Saving all entries...
Sorted entries saved successfully, check your folder
```

El método Bubble Sort lee los elementos de la lista según como estén ordenados, si el primero es el mayor va a intercambiando hasta terminar en la última posición, básicamente intercambia posición con los que sean menores de lado derecho y si es mayor que los presentados del lado izquierda sin incluir su siguiente ya que por eso no se movió de la lista va a esperar otra vez la leída de los datos y el ordenamiento. Este método es $O(n^2)$ por lo cual es muy tardado. En la imagen debajo se ve la ejecución del código con bubble sort.

```
Sorting entries...
Entries sorted
Time difference (sec) = 93.7404
Saving sorted entries...
Sorted entries saved.
```

Comparando ambos métodos de ordenamiento observamos que el merge sort es el más conveniente de utilizar a diferencia del burbuja y hay una gráfica a continuación que demuestra la eficiencia de los ordenamientos. Aunque pudimos utilizar el quick sort optamos por usar el merge sort en esta ocasión para ver la ejecución ya que era un número mayor de datos y de todas formas fue muy eficaz pues nos regreso un total de 16807 datos en 0.184 segundos.



Para encontrar cada elemento usamos un método que se llama búsqueda secuencial, lo que hace es leer elemento por elemento, también conocido como búsqueda lineal ya que va uno en uno. La complejidad de este método es $O(n)$ ya que pasa por todo el arreglo o vector, fue utilizado para seleccionar el primer y último elemento de los datos que quiere consultar el usuario.

Queda concluir que el uso de estos métodos sirven para ahorrar tiempo, mejorar la comprensión del análisis de datos y poder tener un mayor conocimiento de cual es la mejor manera para organizar los datos sin importar si el número de estos es muy grande.

Referencias

Díaz, N. (2006). BÚSQUEDA DE VECTORES. *Unicauca*. Recuperado de:
<http://artemisa.unicauca.edu.co/~nediaz/EDDI/cap02.htm>

Anónimo (s.f). Algoritmos de ordenación y búsqueda. Recuperado de:
<https://sites.google.com/a/espe.edu.ec/programacion-ii/home/a1-arreglos/algoritmos-de-ordenacion-y-busqueda>

Cormen, T. Balkcom, D. (2021). Análisis del ordenamiento por mezcla. *Khan Academy*.
Recuperado de:
<https://es.khanacademy.org/computing/computer-science/algorithms/merge-sort/a/analysis-of-merge-sort>