

# Análisis de algoritmos promedio: Merge sort

---

Alberto Benavides  
23 de abril de 2020

## 1. MERGE SORT

El algoritmo de *merge sort* parte de un arreglo  $A$  de  $n$  elementos que desean ordenarse a partir de un criterio dado. En el peor de los casos, el algoritmo

- divide el arreglo en dos arreglos de tamaño  $n/2$  hasta que todos los arreglos tengan tamaño de 1, lo que se logra en  $\mathcal{O}(n - 1)$ ,
- compara y combina en un arreglo todos los elementos entre sí por pares de arreglos, reordenándolos, en el peor de los casos, en tiempo  $\mathcal{O}(n - 1)$ ,

de donde se tiene  $\mathcal{O}((n - 1) * (n - 1)) = \mathcal{O}(n^2)$ . Sin embargo, esto sucedería en un arreglo en que se deba hacer intercambios en todos los pares, por ejemplo

$[4, 0, 6, 2, 5, 1, 7, 3]$ .

Aquí, primero se separa el arreglo por la mitad y así sucesivamente hasta llegar a arreglos de un elemento:

$[4, 0, 6, 2]; [5, 1, 7, 3]$

$[4, 0]; [6, 2]; [5, 1]; [7, 3]$

$[4]; [0]; [6]; [2]; [5]; [1]; [7]; [3]$ .

Luego se comparan y unen todos los elementos entre sí por pares de arreglos, ordenando, en este caso, en todos los niveles el máximo de veces:

$[4, 0]; [6, 2]; [5, 1]; [7, 3]$

$[0, 2, 4, 6]; [1, 3, 5, 7]$

$[0, 2, 4, 6]; [1, 3, 5, 7]$

$[0, 1, 2, 3, 4, 5, 6, 7]$ .

Esto sucede porque cada vez que se ordenan elementos, se deben cambiar de lado. Aún así, para cada tamaño de elementos de un arreglo, sólo existe un caso en el que se deban reordenar todos los elementos en todos los niveles. Este arreglo para el peor caso en cada tamaño de arreglo se pueden obtener mediante un algoritmo [1] que consiste en partir de los dos peores algoritmos para tamaños de arreglos de uno y dos elementos, a saber [1] y [2, 1]. A partir de ellos se puede calcular recursivamente el arreglo de  $x$  elementos para el peor caso del *merge sort*. Para ello,

- se toman los peores arreglos de tamaños  $a$  y  $b$  que cumplan  $a + b = x$ ,
- se duplican los elementos de ambos arreglos,
- se resta uno a cada elemento del arreglo con menos elementos, y
- se forma un arreglo que lleve primero los elementos del arreglo de menos elementos y luego los del restante.

Con todo, para el caso promedio se puede suponer que no se reemplazarán todos los elementos, por lo que bastaría con separar y luego unir los elementos de cada nivel en tiempo  $\mathcal{O}(2n)$  y hacerlo  $\log_2 n$  veces, pues ese es el número de niveles que se forman, por lo que la complejidad computacional esperada en promedio es  $\mathcal{O}(2n \log_2 n)$ .

## REFERENCIAS

- [1] Morgan Wilde. When will the worst case of Merge Sort occur? <https://stackoverflow.com/a/33409294>, 2020. [Accedido 23/abril/2020].
- [2] Elisa Schaeffer. Complejidad computacional de problemas y el análisis y diseño de algoritmos. [No publicado], 2019.