

# Análise da Complexidade do Merge Sort

## Complexidade de Tempo

O Merge Sort possui uma estrutura recursiva com divisão e conquista, e a complexidade de tempo pode ser analisada em relação às suas fases:

### 1. Divisão do problema:

- Em cada chamada recursiva, o vetor é dividido em duas partes iguais. Isso acontece até que cada subvetor tenha apenas um elemento.
- O número de divisões é  $\log_2(n)$ , onde  $n$  é o tamanho do vetor.

### 2. Conquista (Mesclagem dos subvetores):

- Na fase de mesclagem, todos os elementos são comparados uma vez para ordenar e combinar os subvetores.
- O tempo necessário para mesclar dois subvetores é proporcional ao número total de elementos envolvidos ( $O(n)$ ).

Combinar estas duas fases resulta em uma complexidade total de tempo de:

$$T(n) = 2T(n/2) + O(n)$$

---

Por meio de substituição e solução da recorrência, obtemos:

$$T(n) = O(n \log n)$$

---

## Complexidade de Espaço

A complexidade de espaço depende do uso de memória adicional para armazenar subvetores temporários e da profundidade da pilha de recursão:

### 1. Espaço adicional para vetor auxiliar:

- Um vetor auxiliar `vaux[]` de tamanho  $O(n)$  é alocado em cada mesclagem.

### 2. Pilha de recursão:

- A profundidade máxima da pilha de recursão é  $\log_2(n)$ , uma vez que o vetor é dividido ao meio em cada etapa.

A complexidade total de espaço é, portanto:

$$S(n) = O(n) \text{ (vetor auxiliar)} + O(\log n) \text{ (recursão)}$$

---