

Identificação das operações principais e a quantificação do número de ocorrências para todos os algoritmos.

1. Bubble Sort

1.1 Operações principais

- **Comparações:** comparação de pares adjacentes.
- **Trocas:** ocorre quando $\text{arr}[i] > \text{arr}[i+1]$.

1.2 Quantificação

Caso	Comparações	Trocas
Melhor caso (vetor já ordenado)	$\sim n(n-1)/2$	0
Médio caso	$\sim n(n-1)/2$	$\sim n(n-1)/4$
Pior caso (ordem inversa)	$\sim n(n-1)/2$	$\sim n(n-1)/2$

2. Insertion Sort

2.1 Operações principais

- **Comparações:** entre o elemento atual e os anteriores.
- **Movimentações:** deslocamento dos elementos maiores uma posição à direita.

2.2 Quantificação

Caso	Comparações	Movimentações
Melhor caso (já ordenado)	$n-1$	$n-1$
Médio caso	$\sim n^2/4$	$\sim n^2/4$
Pior caso (ordem inversa)	$\sim n(n-1)/2$	$\sim n(n+1)/2$

3. Selection Sort

3.1 Operações principais

- **Comparações:** busca do menor elemento na parte não ordenada.
- **Trocas:** 1 troca ao final de cada iteração (se necessário).

3.2 Quantificação

Caso	Comparações	Trocas
Todos os casos	$n(n-1)/2$	$n-1$ (máximo)

4. Merge Sort

4.1 Operações principais

- **Comparações:** realizadas na etapa de *merge*.

- **Movimentações:** cada elemento é copiado para um vetor auxiliar e depois devolvido.

4.2 Quantificação

Merge Sort tem complexidade garantida **$O(n \log n)$** .

Caso	Comparações	Movimentações
Melhor, médio e pior	entre $n \log_2(n)$ e $n \log_2(n) + n$	$\sim 2n \log_2(n)$

5. Quick Sort

5.1 Operações principais

- **Comparações:** durante particionamento.
- **Trocas:** sempre que um elemento \leq pivô vai para a esquerda.

A análise depende fortemente da escolha do pivô.

5.2 Quantificação

Caso	Comparações	Trocas
Melhor caso	$\sim n \log n$	$\sim n \log n$
Médio caso	$\sim 1.39 n \log n$	$\sim \sim 0.39 n \log n$
Pior caso (pivô ruim – ex.: vetor já ordenado)	$\sim n^2$	$\sim n^2/2$

6. Shell Sort

6.1 Operações principais

- **Comparações:** como no insertion sort, mas considerando gaps.
- **Movimentações:** deslocamentos durante as “insertion sorts internas”.

Shell Sort não tem fórmula fechada geral, pois depende da sequência de gaps.

Usando a sequência clássica: $n/2, n/4, n/8, \dots, 1$

Caso	Comparações	Movimentações
Melhor caso	$O(n \log n)$	$O(n \log n)$
Médio caso	entre $O(n^{1.25})$ e $O(n^{1.5})$	mesmo nível
Pior caso	$O(n^2)$	$O(n^2)$