O Algoritmo Bubble Sort

Bubble Sort é um algoritmo de ordenação simples. Ele compara elementos adjacentes. Troca-os se estiverem em ordem errada.

Made with Gamma

Entendendo o funcionamento do Bubble Sort

1

Comparação

Compara dois elementos adjacentes.

2

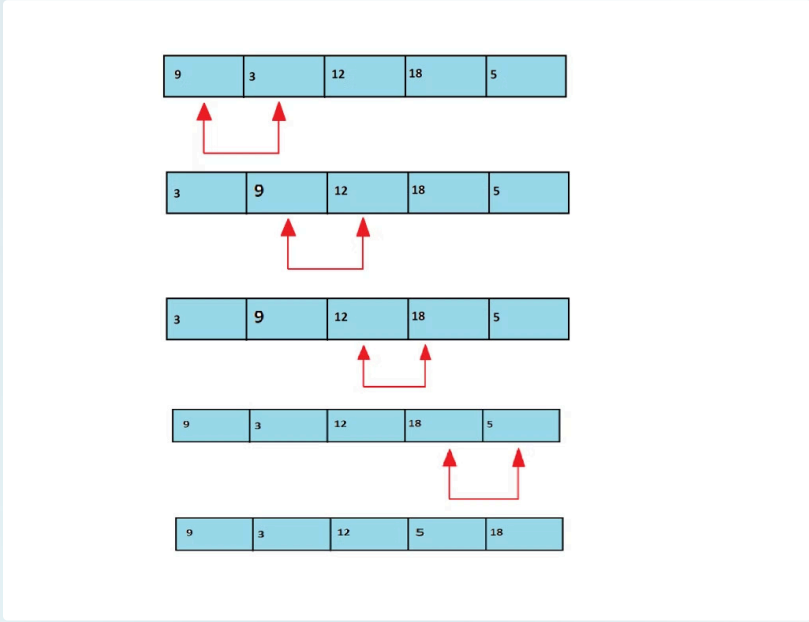
Troca

Se estiverem fora de ordem, troca.

3

Repetição

Repete até que esteja ordenado.



Made with Gamma

Vantagens e Desvantagens do Bubble Sort

O Bubble Sort, apesar de simples e direto, apresenta vantagens e desvantagens que o tornam adequado para cenários específicos, mas não para outros. Sua simplicidade o torna fácil de entender e implementar, especialmente para iniciantes. A lógica do algoritmo é intuitiva e fácil de visualizar, tornando-o uma ótima ferramenta para fins educacionais.

Outra vantagem é a baixa necessidade de espaço adicional na memória. O Bubble Sort opera diretamente no array de entrada, sem exigir estruturas de dados auxiliares complexas. Isso o torna eficiente em ambientes com recursos de memória limitados.

No entanto, para grandes conjuntos de dados, o Bubble Sort é notoriamente ineficiente. Sua complexidade de tempo é $O(n^2)$, o que significa que o tempo necessário para ordenar um conjunto de dados aumenta quadraticamente com o número de elementos. Essa ineficiência pode tornar o Bubble Sort impraticável em cenários com muitos dados.

Além disso, o Bubble Sort não é um algoritmo estável. Ele pode mudar a ordem relativa de elementos com o mesmo valor, o que pode ser um problema em algumas aplicações. Isso ocorre porque o algoritmo compara e troca elementos adjacentes, e a ordem relativa de elementos com o mesmo valor pode ser alterada durante a ordenação.

Made with Gamma

Bubble Sort using Python

Implementação do Bubble Sort

Pseudocódigo

Para cada elemento, compare com o próximo. Troque se necessário. Repita.

Python

Implementação concisa e legível em Python.

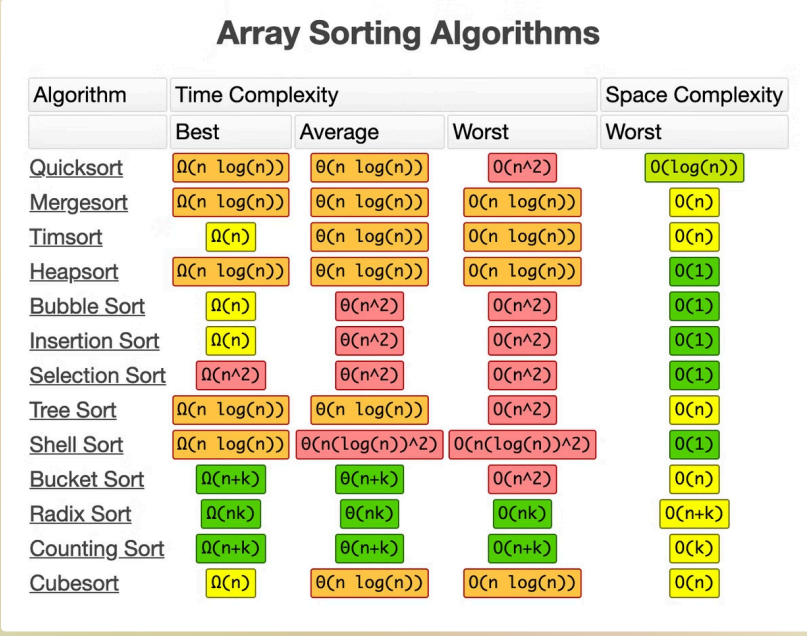
JavaScript

Versão equivalente em JavaScript para aplicações web.

Made with Gamma

Comparando o Bubble Sort com outros métodos de ordenação

Algoritmo	Melhor Caso	Pior Caso
Bubble Sort	O(n)	$O(n^2)$
Merge Sort	O(n log n)	O(n log n)
Quick Sort	O(n log n)	$O(n^2)$



Made with Gamma