

## ALGORÍTMO BUBBLESORT:

Ele faz a comparação 1 a 1, ou seja pega dois elementos próximos do vetor e compara o maior que no caso seria:

Seja  $n = \text{fim}$

se  $V[i] > V[i+1]$

Caso seja joga pra frente, ou seja  $V[i]$  fica no lugar de  $V[i+1]$ , até terminar o primeiro loop (até  $\text{fim} - 1$  do for)

1ª Iteração

i=0	23	4	67	-8	90	54	21	Trocar
i=1	4	23	67	-8	90	54	21	OK
i=2	4	23	67	-8	90	54	21	Trocar
i=3	4	23	-8	67	90	54	21	OK
i=4	4	23	-8	67	90	54	21	Trocar
i=5	4	23	-8	67	54	90	21	Trocar
Final	4	23	-8	67	54	21	90	

Quando terminar o loop do for, decrementa o fim, ou seja diminuimos o tamanho do vetor para comparar novamente dentro do laço for até que semafaro(continua) seja igual a 0, ou seja não entre no laço do if, o que quer dizer que foi realizada todas as trocas possíveis e tudo ok.

Sem Ordenar

23	4	67	-8	90	54	21
----	---	----	----	----	----	----

1ª Iteração

i=0	23	4	67	-8	90	54	21	Trocar
i=1	4	23	67	-8	90	54	21	OK
i=2	4	23	67	-8	90	54	21	Trocar
i=3	4	23	-8	67	90	54	21	OK
i=4	4	23	-8	67	90	54	21	Trocar
i=5	4	23	-8	67	54	90	21	Trocar
Final	4	23	-8	67	54	21	90	

2ª Iteração

i=0	4	23	-8	67	54	21	90	OK
i=1	4	23	-8	67	54	21	90	Trocar
i=2	4	-8	23	67	54	21	90	OK
i=3	4	-8	23	67	54	21	90	Trocar
i=4	4	-8	23	54	67	21	90	Trocar
Final	4	-8	23	54	21	67	90	

3ª Iteração

i=0	4	-8	23	54	21	67	90	Trocar
i=1	-8	4	23	54	21	67	90	OK
i=2	-8	4	23	54	21	67	90	OK
i=3	-8	4	23	54	21	67	90	Trocar
Final	-8	4	23	21	54	67	90	

4ª Iteração

i=0	-8	4	23	21	54	67	90	OK
i=1	-8	4	23	21	54	67	90	OK
i=2	-8	4	23	21	54	67	90	Trocar
Final	-8	4	21	23	54	67	90	

5ª Iteração

i=0	-8	4	21	23	54	67	90	OK
i=1	-8	4	21	23	54	67	90	OK

Não houve mudanças: ordenação concluída

Melhor caso:  $O(N)$

Pior Caso:  $O(N^2)$

Não recomendado para grandes conjuntos de dados