

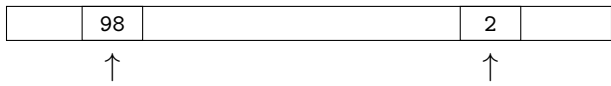
Porque os algoritmos porcaria de ordenação são uma porcaria ?

Bom, é fácil ver porque a ordenação por seleção é uma porcaria: ela faz n vezes uma operação que leva tempo $O(n)$ — (i.e., encontrar o maior elemento em uma faixa da lista)

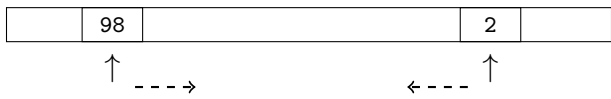
Logo, o seu tempo de execução é $O(n^2)$.

Mas e o algoritmo da bolha, porque ele é uma porcaria?

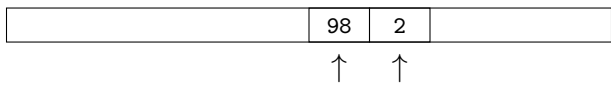
Bom, imagine que tem um elemento grandão no lado esquerdo da lista, e um elemento pequenininho no lado direito



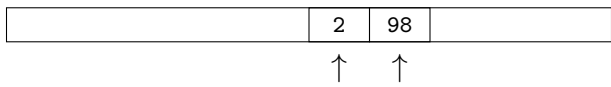
Então, a medida que as varreduras vão acontecendo, o elemento grandão vai andando para a direita, e o elemento pequenininho vai andando para a esquerda



Daí, chega uma hora em que os dois elementos estão lado a lado

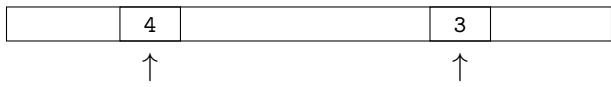


E nesse ponto, eles podem ser comparados e trocados entre si



Certo.

A seguir, a gente nota que isso acontece com toda inversão — (i.e., quando o elemento da esquerda é maior do que o elemento da direita)



Porque os elementos só se movem trocando de lugar com o vizinho — (logo o 4 não pode pular por cima do 3 ...)

Por outro lado, toda troca desfaz uma inversão.

Logo, o número de trocas realizadas pelo algoritmo da bolha é exatamente igual ao número de inversões na lista.

Agora nós podemos perguntar: quantas inversões existem em uma lista desordenada?

Bom, isso depende da lista, claro.

Mas, em uma lista aleatória a probabilidade de que dois elementos estejam invertidos é igual a $1/2$.

Logo, metade dos pares de elementos vai formar uma inversão.

E a quantidade total de pares de elementos é

$$\binom{n}{2} = \frac{n(n-1)}{2} = O(n^2)$$

Pronto, aqui a gente já pode ver porque o algoritmo da bolha é uma porcaria.

Mas não é só isso.

Esse argumento vale para todo algoritmo que troca apenas elementos consecutivos entre si.

E agora a gente já vê porque a ordenação por seleção é uma porcaria.