

Algoritmos (simples) de Ordenação

Técnicas de Programação - 2025/02

Última atualização: 27/08/2025 12:38

Em um primeiro momento, estudaremos dois algoritmos simples de ordenação: *Bubble Sort* e *Selection Sort*. Um dos nossos objetivos aqui é contar o número de **comparações** feitas por esses algoritmos, sempre considerando o **pior caso**.

Bubble Sort

Considere o algoritmo abaixo para as próximas perguntas.

Algorithm 1: BubbleSort

```
Input  : array  $A$ 
Output:
trocou  $\leftarrow$  true
 $C \leftarrow 0$ 

while trocou = true do
    trocou  $\leftarrow$  false
    for  $i \leftarrow 0$  to  $A.length - 1$  do
        if  $A[i] > A[i + 1]$  then
            trocar valores entre  $A[i]$  e  $A[i + 1]$ 
            trocou  $\leftarrow$  true
        end
    end
     $C \leftarrow C + 1$ 
end
```

Nosso objetivo será entender esse algoritmo somente **simulando seu funcionamento**. Sabemos que ele ordena o array A , mas o ponto aqui é entender **como** ele faz isso.

Exercício: Para o array $A = [3, 4, 2, 5, 77, 8, 12]$, simule a primeira iteração do loop *while*. Escreva abaixo o valor final de A , C e trocou.

Exercício: Faça o mesmo para o array $A = [8, 7, 6, 5, 4, 3, 2, 1]$.

Exercício: Faça o mesmo para o array $A = [10, 20, 30, 40, 50, 60, 70, 80]$. O que você notou de diferente em relação aos anteriores?

Exercício: Começando agora com o array $A = [3, 4, 2, 5, 77, 8, 12]$, preencha a tabela abaixo. Cada linha corresponde a uma vez que foi feita uma troca de duas posições vizinhas. As primeiras duas linhas já estão preenchidas.

C	i	A
0	1	[3, 2, 4, 5, 77, 8, 12]
0	4	[3, 2, 4, 5, 8, 77, 12]

Exercício: Começando agora com o array $A = [5, 4, 7, 2, 9, 7, 11, 32, 4]$, preencha a tabela abaixo. Cada linha corresponde a uma vez que foi feita uma troca de duas posições vizinhas. As primeiras duas linhas já estão preenchidas.

C	i	A
-----	-----	-----

Vamos agora tentar entender o número máximo de operações feitas por este algoritmo. Para isso, vamos usar o array ordenado em ordem **decrescente** como exemplo. Dado que ele está totalmente ao contrário, é um bom exemplo para testar algoritmos de ordenação.

Exercício: Começando agora com o array $A = [5, 4, 3, 2, 1]$, preencha a tabela abaixo. Cada linha corresponde a uma vez que foi feita uma troca de duas posições vizinhas.

C	i	A
-----	-----	-----

Exercício: O algoritmo fez quantas trocas no exemplo acima? Se a entrada tivesse tamanho $N = 6$, quantas você acha que ele faria? Use esse raciocínio para estimar o número de trocas feitas para entradas de tamanho $N > 0$ quaisquer.

Agora que entendemos melhor o algoritmo, vamos implementá-lo no PrairieLearn. Como bônus, assistam o vídeo abaixo e tentem relacioná-lo com a execução do algoritmo no início desta atividade.



Figure 1: qr bubble sort