

Desafio

Ordenação e busca

Chegamos, agora, ao fim do módulo II e, como forma de solidificar todo o conhecimento visto até aqui, um desafio de ordenação e busca será proposto. Vocês viram, na prática, a função `sort()` funcionando, esta função possibilita que o usuário do programa ordene uma variável do tipo array em ordem crescente ou decrescente. Neste desafio, iremos implementar uma função `sort`, além disso, vamos utilizar o resultado obtido para encontrar uma chave dentro do array já ordenado e, verificar se a posição (índice) dessa chave no array, é a mediana do vetor passado.

Para realizar essa atividade, vocês **não** poderão usar a função `sort()`, como dito, vocês deverão implementar a sua própria função, fazendo uso das estruturas relacionais e de repetição vistas anteriormente.

Sendo assim, vocês irão precisar implementar um script e três funções, da seguinte maneira:

- O script deve declarar um vetor de tamanho 15, digamos:

$Vetor = [9, 13, -2, 4, 5, 3, 0, 1, -5, 11, 13, 8, 22, 42, 73]$

e realizar a chamada da primeira função, a qual deve ordenar esse vetor em **ordem crescente**. Após ordenado, essa função deve chamar a segunda função passando como entrada o vetor ordenado, essa função irá percorrer o vetor (*de modo inteligente*) e retornar o valor da posição do número 22. Feito isso, **a primeira função** irá chamar a terceira e última função, que irá receber o valor da posição do número 22, e verificar se essa posição é ou não a mediana do vetor, caso não seja, a função irá calcular essa mediana e retornar o valor correto.

O vetor ordenado, o índice correspondente ao número 22 e a mediana do vetor, devem ser impressos no Command Window **pelo script**, portanto, vocês devem fazer uso de funções com variáveis de saída, assim, conseguirão recolher e armazenar os resultados de cada função, retornando esses valores para o script.

***Obs.:** Note que a terceira função deverá receber não só a posição do número 22 no vetor ordenado, mas também, o próprio vetor, afim de calcular a mediana corretamente caso o resultado não seja válido.*