

## Lista de exercícios

### – Ordenação: *Merge* e *Quick* sort –

1. Selecione a alternativa correta tal que os algoritmos de ordenação estejam associados corretamente às suas respectivas características indicadas a seguir:
  - (i) Sempre divide a lista em dois pedaços de tamanhos iguais ou com diferença a 1 e ordena as duas metades recursivamente;
  - (ii) Divide a lista com base em um pivô e ordena as duas partes recursivamente.

#### Alternativas:

- A - (i) Selection sort; (ii) Insertion sort
  - B - (i) Quick sort; (ii) Merge sort
  - C - (i) Merge sort; (ii) Quick sort
  - D - (i) Quick sort; (ii) Selection sort
2. Seja  $v = [30, 5, 12, 40, 70, 32, 16, 21]$  um vetor não ordenado. Identifique quais serão os vetores que, quando intercalados ao final do processo de ordenação *Merge sort*, resultarão no vetor  $v$  ordenado. Mostre cada um dos passos da ordenação que te levaram a obter tal resposta.
  3. Seja  $v = [11, 45, 56, 3, 12, 89]$  um vetor não ordenado. Identifique quais serão os vetores que, quando intercalados ao final do processo de ordenação *Merge sort*, resultarão no vetor  $v$  ordenado. Mostre cada um dos passos da ordenação que te levaram a obter tal resposta.
  4. No algoritmo de ordenação denominado *Quick sort*, escolhe-se um ponto de referência denominado pivô e separam-se os elementos em dois grupos: à esquerda ficam os elementos maiores que o pivô e, à direita, ficam os menores. Repete-se esse processo para os grupos de elementos formados (esquerda e direita) até que todos os elementos estejam ordenados. Verdadeiro ou falso? Justifique sua resposta.
  5. Seja  $v = [56, 65, 7, 10, 3, 20, 9]$  um vetor não ordenado. Aplique o algoritmo de ordenação *Quick sort* no vetor  $v$  utilizando como pivô  $p$  o elemento central do vetor (ou seja, sendo  $l$  e  $r$  os índices das extremidades esquerda e direita do vetor respectivamente, considera-se  $p = \lfloor \frac{l+r}{2} \rfloor$ ). Durante a ordenação, o subvetor mais à esquerda do pivô deve ser ordenado antes do subvetor mais à direita. Mostre cada um dos passos da ordenação que levaram a obter o vetor ordenado. Considere que um *passo de ordenação* está completo quando o pivô está em sua posição definitiva.

### Informações úteis:

- Para conferir o funcionamento dos algoritmos de ordenação, utilizar o simulador disponível em
  - [www.hackerearth.com/pt-br/practice/algorithms/sorting/merge-sort/visualize/](http://www.hackerearth.com/pt-br/practice/algorithms/sorting/merge-sort/visualize/) (merge sort)
  - [www.hackerearth.com/pt-br/practice/algorithms/sorting/quick-sort/visualize/](http://www.hackerearth.com/pt-br/practice/algorithms/sorting/quick-sort/visualize/) (quick sort)

### Respostas:

Exercício 1:  
Alternativa C

Exercício 2:

- Passo 1: [5, 30]
- Passo 2: [12, 40]
- Passo 3: [5, 30] + [12, 40]  
Resulta no vetor  $v_a$ : [5, 12, 30, 40]
- Passo 4: [32, 70]
- Passo 5: [16, 21]
- Passo 6: [32, 70] + [16, 21]  
Resulta no vetor  $v_b$ : [16, 21, 32, 70]

Exercício 3:

- Passo 1: [11, 45]
- Passo 2: [11, 45] + [56]  
Resulta no vetor  $v_a$ : [11, 45, 56]
- Passo 3: [3, 12]
- Passo 4: [3, 12] + [89]  
Resulta no vetor  $v_b$ : [3, 12, 89]

Exercício 4: Falso. No algoritmo de ordenação denominado *Quick sort*, escolhe-se um ponto de referência, denominado pivô, e separam-se os elementos em dois grupos: **à direita** ficam os elementos maiores que o pivô e, **à esquerda**, ficam os menores.

Exercício 5 (Os pivôs estão em **negrito**. Os elementos escritos em cinza indicam a parte do vetor que não é considerada em cada passo):

- Passo 1: [3, 9, 7] + [10] + [56, 20, 65]
- Passo 2: [3, 7] + [9] + [10, 56, 20, 65]
- Passo 3: [3] + [7] + [9, 10, 56, 20, 65]  $\rightsquigarrow$  neste ponto, a primeira metade do vetor (posições 0 a 3) está ordenada
- Passo 4: [3, 7, 9, 10] + [20] + [65, 56]  $\rightsquigarrow$  começando a ordenar a segunda metade do vetor (posições 4 a 6)
- Passo 5: Passo 6: [3, 7, 9, 10, 20] + [56] + [65]  $\rightsquigarrow$  neste ponto, o vetor está totalmente ordenado