

# Introdução aos Algoritmos e Estruturas de Dados

## (<https://fenix.tecnico.ulisboa.pt/disciplinas/IAED7645111326/2016-2017/2-semester>)

- 1) Considere o seguinte vector  $v = \langle 6, 3, 4, 2, 5, 1 \rangle$ . Indique o conteúdo de  $v$  no final de cada passo dos algoritmos **insertion sort** e **selection sort**.
- 2) Considere o seguinte vector  $v = \langle 12, 2, 18, 15, 16, -1, 35, 30, 15 \rangle$ . Indique as razões porque é que o vector  $\langle -1, 2, 12, 15, 18, 15, 16, 30, 35 \rangle$  **não** pode corresponder ao conteúdo de  $v$  num passo intermédio da aplicação dos algoritmos insertion sort, selection sort ou merge sort?
- 3) Considere a aplicação do algoritmo **bubblesort** ao vector  $\langle 20, 11, 16, 9, 12, 14, 17, 19, 13, 15 \rangle$ . Supondo que o algoritmo percorre o vector da esquerda para a direita em cada iteração, qual o conteúdo do vector após as duas primeiras iterações do algoritmo bubblesort ?
- 4) Diga quais dos seguintes vectores corresponde a um **amontoado** (heap)?
- a.  $\langle 50, 25, 30, 27, 24, 21, 28 \rangle$
  - b.  $\langle 50, 30, 25, 27, 24, 28, 21 \rangle$
  - c.  $\langle 60, 50, 9, 40, 41, 10, 8 \rangle$
  - d.  $\langle 40, 15, 18, 13, 11, 14, 16 \rangle$
  - e.  $\langle 60, 30, 80, 10, 35, 70, 40 \rangle$
- 5) A primeira operação do algoritmo **heapsort** é transformar o vector num amontoado. Considere que o vector de entrada do algoritmo é  $\langle 20, 11, 16, 9, 12, 14, 17, 19, 13, 15 \rangle$ .
- 5.1. Indique o conteúdo do vector após o passo de transformação num amontoado.
  - 5.2. Indique ainda o conteúdo do vector após os dois maiores elementos terem sido ordenados (colocados na sua posição final), durante a operação de ordenação (heapsort).
- 6) Qual o conteúdo do seguinte vector  $\langle 25, 19, 23, 15, 18, 16, 21, 12 \rangle$  depois de os dois primeiros elementos (i.e. os dois maiores) terem sido ordenados, utilizando o algoritmo de ordenação **heapsort**?
- 7) Considere a implementação clássica da função `int partition (Item v[], int l, int r)` usada no algoritmo **quicksort** tal como apresentada nas aulas teóricas. Esta função recebe o vector  $v$  e as posições  $l$  e  $r$  que definem, respectivamente, os índices limite esquerdo e direito do vector a considerar na função. Suponha que o procedimento `partition` é invocado com os seguintes argumentos:  $v = \langle 13, 6, 5, 14, 12, 4, 16, 18, 7, 9, 10 \rangle$ ,  $l = 0$ ,  $r = 10$ . Considerando a posição `a[r]` como pivot, indique qual o conteúdo do vector  $v$  após a execução da função `partition`.
- 8) Considere o exercício anterior, mas onde os argumentos da função **partition** são os seguintes:  $v = \langle 20, 11, 16, 9, 12, 14, 17, 19, 13, 15 \rangle$ ,  $l = 0$ ,  $r = 9$ . Qual o conteúdo do vector  $v$  após a execução do procedimento `partition`?

- 9) (**Radix LSD**) Considere a aplicação do algoritmo radix sort LSD, em que cada passo os elementos são ordenados considerando um dígito, ao seguinte vector:

$\langle 48372, 62309, 83861, 91874, 18913, 33829, 47812, 95954, 52377, 22394, 56108, 60991 \rangle$

Qual é o terceiro número da sequência, após o algoritmo ter considerado três dígitos?

- 10) (**Radix MSD**) Considere o seguinte vector de números inteiros sem sinal de 6 bits:

$\langle 32, 2, 34, 9, 6, 1, 20, 18, 10 \rangle$ . Qual o conteúdo do vector após os primeiros dois passos do algoritmo de

ordenação radix sort MSD, em que em cada passo os elementos são ordenados considerando 2 bits (ou seja, byte = 2 bits) ?

Nota: considere que o algoritmo é baseado numa versão estável do algoritmo counting sort. O algoritmo deve apenas processar os 6 bits menos significativos de cada número, independentemente dos números poderem ser guardados em palavras com maior número de bits.

## Sugestão de resolução

- 1) R: Ver aulas teóricas.

2)

- 3) R:  $\langle 11, 9, 12, 14, 16, 17, 13, 15, 19, 20 \rangle$

- 4) R: Ver aulas teóricas.

5)

5.1. :  $\langle 20, 19, 17, 13, 15, 14, 16, 9, 11, 12 \rangle$

5.2. :  $\langle 17, 15, 16, 13, 12, 14, 11, 9, 19, 20 \rangle$

- 6) R:  $\langle 21, 19, 16, 15, 18, 12, 23, 25 \rangle$

- 7) R:  $\langle 9, 6, 5, 7, 4, 10, 16, 18, 14, 13, 12 \rangle$

- 8) R:  $\langle 13, 11, 14, 9, 12, 15, 17, 19, 20, 16 \rangle$

- 9) R: ver exercícios das aulas teóricas

- 10) R: ver "Binary MSD" discutido na aula teórica. Adapte o exemplo dado para 2 bits.