

G

DEI DEPARTAMENTO DE ENGENHARIA INFORMÁTICA TÉCNICO LISBOA

Introdução aos algoritmos e estruturas de dados, 2021

ep05 (25min)

I (10 val.) Considere o procedimento de partition indicado abaixo utilizado no algoritmo quickSort.

**1 typedef int Item;**

**#define key (A) (A)**

**#define less (A, B) (key (A) < key (B) ) 4 #define exch (A, B) { Item t = A; A = B; B = t; }**

OO JOGA WNPOO JO

int partition (Item a [], int left, int right) {

**int i = left; int j = right+1; Item v = a[left]; while (i < j) { while (less (a [ ++i), v)) if (i == right)**

break; while (less (v, a [--j])); if (i < j)

exch (a[i], a[j]);

**exch (a[j], a [left]); return ji**

19 20 }

Considere o vector a indicado abaixo. Indique o conteúdo do vector a após a chamada à função partition (a, 0, 10). Na sua resposta, indique a sequência das trocas de elementos do vector, assim como o valor de retorno da chamada à função. a = { 15, 18, 20, 21, 25, 29, 3, 7, 8, 12, 14 }

II (10 val.) Escolha um dos vectores abaixo que não seja um amontoado e explique como se poderia transformar esse vector num amontoado usando a operação FixDown. Indique os passos

ação FixDown e o vector resultante.

1. { 50, 40, 35, 20, 25, 34, 33, 11, 19, 22, 24 } 2. { 50, 35, 40, 34, 22, 20, 24, 33, 25, 19, 11 } 3. { 50, 33, 40, 24, 25, 35, 34, 11, 20, 22, 19 } 4. { 50, 34, 40, 24, 22, 25, 33, 35, 11, 20, 19 } 5. { 50, 40, 35, 25, 20, 33, 34, 22, 24, 11, 19 }