

<b>Departamento de Matemática da Universidade de Coimbra</b>		
<b>2020/2021</b>	<b>Programação Orientada para os Objectos</b>	<b>Folha 6 (2022/06/06; v1256)</b>

## 9 Ordenação/Pesquisa

### 9.1 Ordenação

**49** Ordene os primeiros  $n$  elementos de um vector usando o seguinte algoritmo denominado de **Borbulagem**:

1. percorra o vector trocando pares de elementos que estejam fora de ordem.
2. repita o processo até que, numa das passagens anteriores, não se proceda a nenhuma troca.

**50** Ordene os primeiros  $n$  elementos de um vector usando o seguinte algoritmo denominado de **Seleccção Linear**:

1. encontre o maior dos  $n$  elementos;
2. coloque esse elemento na sua posição final trocando-o com o  $n$ ésimo elemento;
3. recursivamente ordene  $n-1$  primeiras posições do vector.

Note-se que a recursão pode ser facilmente eliminada.

**51** Ordene os primeiros  $n$  elementos de um vector usando o seguinte algoritmo denominado de **Inserção Linear**:

1. Consideramos o vector como a concatenação de duas sequências: uma ordenada, e a segunda não ordenada.

No início a primeira sequência contém apenas um elemento.

2. O primeiro elemento da sequência não ordenada é inserido na posição correcta da sequência ordenada (movendo os elementos maiores uma posição para a esquerda).

Esta operação aumentou a sequência ordenada de um elemento e retira esse elemento à sequência não ordenada.

3. Repetimos até todos os elementos estarem na sequência ordenada.

**52** Ordene os primeiros  $n$  elementos de um vector usando o seguinte algoritmo denominado de **Quick Sort**:

Dado um vector de  $n$  inteiros e um inteiro  $l$  existente no vector pretende-se particionar esse vector em duas partes: os elementos inferiores a  $l$  e os elementos superiores a  $l$ .

Uma maneira de fazer isto é a seguinte:

1. Começando do lado esquerdo do vector procurar um elemento que seja maior ou igual a  $l$ .
2. Começando do lado direito do vector procurar um elemento que seja menor ou igual a  $l$ .
3. Trocar estes dois elementos.
4. Continuar até as duas procuras se cruzarem.

## 9.2 Pesquisa

**53** Dado um vector e um seu possível elemento construa programas para verificar se o mesmo está, ou não, contido no vector. Implemente os métodos a seguir descritos.

1. Método de procura exaustiva. Caso o vector não esteja ordenado o único algoritmo possível é dado pela procura exaustiva desde o início do vector até ao seu fim.
2. Caso o vector esteja ordenado a pesquisa pode ser feita de forma muito eficiente. O algoritmo é designado por **Pesquisa binária**:
  - (a) verificar se o elemento é igual à posição média do vector;
  - (b) caso seja menor do que a posição média, repetir o processo na primeira metade do vector;
  - (c) caso seja maior do que a posição média, repetir o processo na segunda metade do vector.

O algoritmo para assim que encontrar o elemento, ou quando atinge um vector de dimensão nula.