

## BLOCO 2 – Estruturas de Dados Indexadas – PL8

*ASSUNTO – Estruturas de Dados Indexadas*

### Objetivos Específicos:

- Mediante a apresentação dum problema, os alunos deverão ser capazes de o analisar, conceber e descrever o algoritmo estruturado em módulos e utilizar *arrays* mono e bidimensionais.
- Desenvolver métodos de manipulação de *arrays*.

**TAREFAS DA SEMANA:**

### Exercício 1 (\*\*)

Considerando o seguinte programa:

```
public class Enigma {

    public static void main( String[] args ) {
        int x;
        int m[][] = {{1,4,2,1},{9,7,2,2},{1,7,3,5},{2,5,0,3},{4,7,2,1}};

        for (int i=0; i < m.length; i++) {
            x = m[i][0];
            for (int j=1; j < m[i].length; j++){
                if (m[i][j] > x){
                    x= m[i][j];
                }
            }
            System.out.println(x);
        }
    }
}
```

- Descreva a sua funcionalidade;
- Construa um método para mostrar a matriz m;
- Crie um método para apresentar a média de cada coluna e a média global da matriz m;
- Crie um método para mostrar a matriz transposta da matriz m.

### Exercício 2 (\*\*\*)

Um quadrado mágico é uma matriz quadrada de números inteiros onde a soma dos números de qualquer linha, de qualquer coluna e de cada uma das diagonais dá sempre o mesmo valor.

Exemplo de um quadrado mágico:

8	1	6
3	5	7
4	9	2

Elabore uma aplicação modular que permita verificar se uma dada matriz é um quadrado mágico.

Deve iniciar pedindo ao utilizador a dimensão da matriz e, de seguida, cada um dos seus elementos.

## BLOCO 2 – Estruturas de Dados Indexadas – PL8

A solução deve ser desenvolvida numa única classe e exclusivamente em modo texto.

A mensagem a apresentar deverá ser apenas: **true** ou **false**.

### Exercício 3 (\*\*\*)

Elabore um programa modular que permita ler uma matriz de números inteiros e determinar a frequência de ocorrência dos números na matriz.

Deve começar por solicitar ao utilizador o número de linhas e de colunas da matriz.

Deverá ainda mostrar a matriz original bem como todos os números diferentes e respetiva frequência, ordenados de forma decrescente desta. No caso de haver frequências iguais, devem aparecer pela mesma ordem em que se encontram na matriz original.

Exemplo:

Dada a matriz

8	-5	3	8
6	3	10	1
10	8	-5	2

Resultado:

8	-5	3	10	6	1	2
3	2	2	2	1	1	1

Formate a saída com: `System.out.printf("%4d", valorInteiro)`, de modo a que tenha o seguinte formato:

#### Matriz

```
8 -5 3 8
6 3 10 1
10 8 -5 2
```

#### Frequencia

```
8 -5 3 10 6 1 2
3 2 2 2 1 1 1
```

### Exercício 4 (\*\*)

Elabore uma aplicação que leia uma sequência de N números inteiros obrigatoriamente todos diferentes e mostre os M números maiores, sendo N e M definidos pelo utilizador.

### Exercício 5 (\*\*\*)

Elabore um programa modular que leia uma matriz de números inteiros e mostre os índices do elemento E, cuja soma dos elementos da matriz que não pertencem à mesma linha e mesma coluna, desse elemento E, é a maior. Se existirem múltiplos elementos nesta condição, o programa deve apresentar os índices de todos esses elementos.

A título de exemplo, a matriz a seguir apresentada ilustra, numa cor escura, os elementos que não pertencem à mesma linha e mesma coluna do elemento E.

## BLOCO 2 – Estruturas de Dados Indexadas – PL8

	E		

### Exercícios Complementares

#### Exercício 1 (\*\*\*)

Pretende-se fazer a gestão de um sistema de armazenamento automático de produtos. Neste sistema, as prateleiras e os produtos são identificados por números e códigos alfanuméricos, respetivamente. Não existem produtos com códigos iguais e em cada prateleira só pode ficar armazenado um produto. Cada produto fica registado com o número da prateleira onde se encontra armazenado.

Elabore uma aplicação com as seguintes funcionalidades:

- Inserir produtos no sistema, indicando os respetivos códigos de produto e números de prateleiras que ocupam;
- Modificar o número da prateleira ocupada por um produto;
- Remover produtos do sistema;
- Listar os produtos ordenados pelos números das prateleiras.

#### Exercício 2 (\*\*\*)

Uma empresa imobiliária pretende uma aplicação para obter algumas estatísticas sobre as suas vendas realizadas ao longo de um ano. O programa deve ser orientado por menu e tem de satisfazer os seguintes requisitos:

- Guardar os nomes e as vendas mensais (€) de cada um dos N vendedores da empresa;
- Mostrar o montante anual das vendas da empresa;
- Mostrar o(s) nome(s) do(s) vendedor(es) com o melhor montante mensal de vendas;
- Mostrar o(s) nome(s) do(s) vendedor(es) com o melhor montante anual de vendas;
- Apresentar o(s) mês(es) com o pior montante global de vendas;
- Listar a percentagem de vendas anuais de cada um dos N vendedores. A listagem deve ser ordenada pelo nome do vendedor e apresentada com o seguinte formato:

Nome    Total\_Anual    %.

- Apresentar os montantes trimestrais de vendas da empresa.