

## Algoritmia e Programação

### Exame Época Normal – Parte Prática 21 de Janeiro de 2013

Duração: 2 horas

- Responda a cada Grupo em folhas separadas identificadas com o seu nome e número.
- A interpretação do enunciado faz parte da resolução da prova. Se encontrar ambiguidades ou incoerências, resolva-as da melhor maneira e explique as decisões tomadas.

#### Grupo I

Cotação: 3 valores

Elabore um algoritmo que leia um número inteiro positivo e apresente esse número sem algarismos consecutivos repetidos. Por exemplo, se for lido o número 11133866, o algoritmo deve apresentar o número 1386. Caso o número lido não possua algarismos consecutivos repetidos, o algoritmo deve apresentar uma mensagem apropriada.

#### Grupo II

Cotações: 1- 2 val 2- 4 val 3- 2 val

Considere a seguinte classe em que os métodos *criarMatriz* e *mostrarMatriz* estão devidamente implementados:

```
public class Exame {  
  
    public static int[ ][ ] criarMatriz (int n, int m) {  
        /* Cria uma matriz de números inteiros com n linhas e m  
        colunas, preenche-a a partir de valores lidos no teclado e no  
        final retorna essa matriz */  
    }  
  
    public static void mostrarMatriz (int[ ][ ] m) {  
        /* Apresenta no ecrã a matriz m passada por parâmetro */  
    }  
  
    ...  
  
    public static void main(String[] args){  
        /* ... A implementar ... */  
    }  
}
```

Implemente em JAVA os seguintes métodos da classe *Exame*:

### 1. **rotacaoEsquerda**

Recebe por parâmetro uma matriz de números inteiros e faz uma rotação para a esquerda das colunas dessa matriz, ou seja, move os elementos de cada coluna para a coluna à sua esquerda e os elementos da primeira coluna para a última coluna.

Exemplo:

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 1 | 2 | 3 |

 → 

|   |   |   |
|---|---|---|
| 2 | 3 | 1 |
| 2 | 3 | 1 |

### 2. **verificarMatrizContida**

Recebe por parâmetro duas matrizes de números inteiros e verifica se a matriz passada através do primeiro parâmetro contém a outra matriz passada pelo segundo parâmetro. Em caso afirmativo o método retorna *true*, senão retorna *false*.

### 3. **main**

Considerando todos os métodos da classe *Exame*, incluindo os implementados anteriormente, complete este método de forma a:

- Criar e preencher uma matriz de números inteiros, chamada **mat1**. O número de linhas e colunas da matriz devem ser pedidos ao utilizador. O preenchimento da matriz deve ser feito a partir de valores lidos no teclado.
- Mostrar a matriz lida anteriormente com as colunas rodadas para a esquerda, usando o método implementado no ponto 1.
- Criar e preencher uma matriz de números inteiros de dimensão 4x4, chamada **mat2**. O preenchimento da matriz deve ser feito a partir de valores lidos no teclado.
- Informar o utilizador de que a matriz criada na alínea anterior está ou não contida em **mat1**.

## Grupo III

Cotação: 4 valores

Elabore um programa em linguagem Java para determinar estatísticas sobre consultas realizadas por médicos de uma clínica. Os dados encontram-se num ficheiro de texto (de nome *consultas.txt*) que contém os nomes dos médicos e as horas de início e fim de cada uma das consultas realizadas. Cada linha deste ficheiro possui o seguinte formato:

|  |
|--|
| <nome médico>;<hora início consulta 1>;<hora fim consulta 1>;... |
|--|

Exemplo:

Jorge Almeida;14:20;14:50;15:0;15:15

António Luís Cardoso;8:0;8:20;8:30;9:15;9:45;10:5

O programa deve criar um novo ficheiro de texto (de nome *estatisticas.txt*) em que cada linha contém o nome do médico (último e primeiro nome) e as durações máxima, mínima e média das suas consultas (em minutos). Considere que nenhuma consulta dura mais de 60 minutos.

|   |
|---|
| <nome médico>;<duração máxima>;<duração mínima>;<duração média> |
|---|

Exemplo:

Almeida, Jorge;30;15;22

Cardoso, António;45;20;28