

## Algoritmia e Programação

Exame Época Recurso – Parte Prática  
13 de Fevereiro de 2014

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

Descreva um algoritmo, em pseudo-código, que transforma uma sequência de números inteiros positivos numa outra sequência também de números inteiros positivos. A transformação consiste na substituição, em cada número, dos algarismos pares pelo ímpar seguinte e os ímpares pelo par anterior. Os números da sequência inicial pertencem a um intervalo fechado delimitado por dois números inteiros positivos que deverão ser lidos antes da sequência. Esta sequência termina quando for introduzido um número não pertencente ao intervalo.

### Grupo II

Cotações: 1- 2 val 2- 4 val 3- 2 val

Considere a seguinte classe com todas as funcionalidades descritas devidamente implementadas:

```
public class ParesImpares {  
  
    public static int[][] lerMatriz() {  
        /* Cria, preenche e retorna uma matriz de números inteiros */  
    }  
  
    public static char parOuImpar() {  
        /* Pergunta ao utilizador se pretende pesquisar pares ou ímpares, devolvendo um carater ('p' ou  
        'i', respetivamente). */  
    }  
  
    public static void mostrar(int[][] m, String s, int n) {  
        /* Lista para o ecrã os elementos existentes na matriz m, apresentando uma mensagem (String  
        s) relativa à matriz a mostrar. Recebe um terceiro parâmetro n relativo ao número de linhas  
        preenchidas da matriz. Caso esse número seja desconhecido, o parâmetro n deve conter o número  
        total de linhas da matriz */  
    }  
}
```

Utilizando, quando necessário, os métodos apresentados anteriormente, **implemente em JAVA** os seguintes métodos:

1. **indicesMatriz**

Recebe como parâmetros duas matrizes, **mat** e **ind**, de números inteiros e um carater ('p' ou 'i') o qual indica se se pretendem encontrar números pares ou ímpares, respetivamente.

O método deve percorrer a matriz **mat** e deve preencher a matriz **ind** (com duas colunas) com os índices (linha e coluna) relativos aos elementos da matriz **mat** que satisfaçam a condição associada ao terceiro parâmetro. Deve ainda retornar o número de linhas da matriz **ind**

2. **mudaMatriz**

Recebe como parâmetros a matriz **mat** de números inteiros, a matriz **ind** preenchida no método **indicesMatriz** e o número de linhas preenchidas nesse método.

Com base na informação contida na matriz **ind**, deve alterar a matriz **mat** somando uma unidade aos elementos cujos índices não constam na matriz **ind**.

3. **main**

Considerando todos os métodos da classe *ParesImpares* corretamente implementados, incluindo os das alíneas anteriores, codifique este método de forma a:

- Criar, preencher e mostrar a matriz **mat** de números inteiros e mostrar todos os seus elementos.
- Perguntar ao utilizador se pretende números pares ou ímpares.
- Criar e mostrar a matriz **ind** relativa aos índices dos elementos da matriz **mat** que satisfaçam a escolha do utilizador do ponto anterior.
- Alterar todos os elementos da matriz **mat** que não satisfaçam a escolha do utilizador em termos de par ou ímpar, somando-lhe uma unidade.
- Mostrar a matriz alterada.

**Grupo III**

Cotações: 4 val

Elabore um programa em linguagem Java que leia um ficheiro de texto ("Original.txt") que tenha no máximo 100 linhas e construa um outro ficheiro ("Ordenado.txt") onde as frases apareçam por ordem crescente da quantidade de palavras da frase. As frases com a mesma quantidade de palavras devem ser ordenadas alfabeticamente.

Exemplo:

Original.txt	Ordenado.txt
As armas e os barões assinalados Que da ocidental praia Lusitana Por mares nunca de antes navegados Passaram ainda além da Taprobana	Passaram ainda além da Taprobana Que da ocidental praia Lusitana As armas e os barões assinalados Por mares nunca de antes navegados