

# Segundo Exercício Escolar de Programação 1 (IF968)

Fernando Castor  
Centro de Informática  
Universidade Federal de Pernambuco

9 de dezembro de 2015

**Instruções.** Durante a prova, é proibido usar telefone celular com qualquer fim, se comunicar com colegas, realizar consultas a qualquer tipo de material. A falha em observar essas restrições resultará em penalidades na nota da prova. Além disso, em caso de dúvida, indique na prova se identificou uma ambiguidade e como lidou com ela, para evitar de atrapalhar os outros alunos chamando o professor.

1. **(5,0 ptos.)** Construa uma função `realizaCalculos()` que funciona em duas etapas:

**Etapla 1:** Lê um número,  $N$ .  $N$  deve ser igual a um dos números existentes em uma lista contendo apenas números inteiros positivos fornecida como argumento para a função. Se  $N$  não foi um dos números fornecidos na lista de entrada, o programa deve emitir uma mensagem de erro.

**Etapla 2:** O programa imprime na tela as seguintes informações:

- A quantidade de números na lista de entrada **menores que  $N$**
- A quantidade de números lidos na lista de entrada **maiores que  $N$**
- O **maior número** lido na lista de entrada

Por exemplo, se a função `realizaCalculos()` é chamada com a lista `[10, 20, 9, 54, 35, 23, 11]` e depois lê o número 20 a partir do teclado, ela imprimirá 3, 3 e 54. Seu programa deve funcionar de maneira estritamente recursiva, ou seja, é proibido usar laços (`for`, `while`). Não é permitido usar a função `max()` para resolver esta questão.

2. **(5,0 ptos.)** Um arquivo representa um banco de dados de pessoas. Esse arquivo está estruturado da seguinte maneira:

```
123456789-00
Eddard Stark
Winterfell
Westeros
```

```
666666666-69
Cersei Lanister
The Red Keep
Westeros
```

```
777888999-10
Daenerys Targaryen
Meereen
Slaver's Bay
...
```

Crie duas funções:

- `lerRegistros()`: lê todo o conteúdo do arquivo, guardando os elementos em um **dicionário** na memória, usando o CPF como chave para cada registro. Cada um desses registros deve agrupar todas as informações sobre cada pessoa (nome, endereço e país). Seu programa deve ignorar as linhas em branco entre os registros.
- `atualizar()`: recebe um CPF, um nome, um endereço e um país e atualiza os dados de um registro pré-existente (para o CPF fornecido) fornecidos como argumento. Essa atualização deve se refletir no arquivo.