

**ACH2001 – Introdução Ciência da Computação I**  
**EACH – PRIMEIRO SEMESTRE DE 2008**  
**Segunda Prova – 24 de junho de 2008**

Nome: \_\_\_\_\_

Nº USP: \_\_\_\_\_

- Duração: 1 hora e 45 minutos.
  - A prova deve ser feita a caneta.
  - Capriche na indentação e na elegância de suas soluções.
1. [1,5 pontos] Implemente o método `double media(int N)` que lê do teclado  $N$  ( $N \geq 1$ ) números reais e retorna a média dos números lidos.
2. [2,5 pontos] Uma loja possui um cadastro de clientes (classes `Cadastro` e `Cliente` abaixo). Esse cadastro (armazenado em `Cliente[] clientes`) está ordenado pelo nome do cliente. Suponha que a loja deseja fazer promoções diferenciadas para três categorias de clientes: (i) clientes que moram em São Paulo/Capital, (ii) clientes que moram no estado de São Paulo mas não em sua capital e (iii) clientes dos demais estados.

Implemente, pelo menos, os três métodos da classe `Cadastro` abaixo:

- O método `Cliente[] capitalSP()` que devolve os clientes que moram em São Paulo/Capital
- O método `Cliente[] estadoSP()` que devolve os clientes que moram no estado de São Paulo mas não em sua capital
- O método `Cliente[] outrosEstados()` que devolve os clientes que não moram no estado de São Paulo

Observações:

(1) Os três métodos acima não devem alterar a ordenação. Isto é, os clientes continuam ordenados pelo nome do cliente.

(2) Vocês podem (e devem) implementar outros métodos (nas classes `Cliente` e `Cadastro`) que considerem necessários. Isto é, a resposta correta para essa questão envolve a implementação correta das duas classes com os métodos e atributos que vocês achem necessário ao bom funcionamento dessa aplicação.

```
class Cliente {
    private String nome;
    private String cidade;
    private String estado;
}

class Cadastro {
    private Cliente[] clientes;
    public void Cliente[] capitalSP();
    public void Cliente[] estadoSP();
    public void Cliente[] outrosEstados();
}
```

3. [3,0 pontos] Implemente o método `boolean éLatino(int[] [] A)` que verifica se uma dada matriz  $A_{n \times n}$  é um quadrado latino de ordem  $n$ .

Dizemos que uma matriz  $A_{n \times n}$  é um *quadrado latino* de ordem  $n$  se em cada linha e em cada coluna aparecem todos os inteiros  $1, 2, 3 \dots n$  (ou seja, cada linha e coluna é permutação dos inteiros  $1, 2, 3 \dots n$ ).

Exemplos de matrizes (quadrados latinos) de ordem 1, 2 e 3:

$$\begin{array}{c} \left| \begin{array}{c} 1 \end{array} \right| \qquad \left| \begin{array}{cc} 1 & 2 \\ 2 & 1 \end{array} \right| \qquad \left| \begin{array}{ccc} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{array} \right| \end{array}$$

4. [3,0 pontos] Implemente o método `int nroSubSequencias(int[] v, int k)` que dado um array de inteiros  $v$ , calcula (e retorna) o número de subsequências de tamanho  $k$  cujos elementos possuem os mesmos valores. Isto é, achar o número de subsequências  $v_i \dots v_{i+k-1}$ , tal que  $v[i] = v[i+1] = \dots = v[i+k-1]$ .

Ex: Supondo que  $v = [4, 2, 15, 15, 15, 3, 7, 7, 7, 2]$  e  $k = 3$ , o método deve retornar 3, pois o array dado contém três subsequências de tamanho 3: uma subsequência com valores 15 ( $v[2] \dots v[4]$ ) e duas (sobrepostas) com valores 7 ( $v[6] \dots v[8]$  e  $v[7] \dots v[9]$ ).