



## *Programação de Computadores I*

### **SEGUNDA AVALIAÇÃO OBSERVAÇÕES IMPORTANTES**

1. ESTA AVALIAÇÃO CONTÉM 3 QUESTÕES E 3 PÁGINAS:

Questão 1 - 5 pontos (página 1)

Questão 2 – 2 pontos (página 2)

Questão 3 – 3 pontos (página 2)

2. Todos os códigos **deverão** apresentar um cabeçalho contendo a informação sobre o autor do programa e o que o programa faz. Por exemplo:

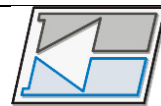
```
/******  
* Autor: Fulano de Tal *  
* Este programa responde se um ano fornecido pelo usuário é um *  
* ano bissexto *  
*****/
```

3. Programas que não compilam/interpretam não serão considerados. Portanto entregue apenas os itens que estiverem funcionando e respeitando o solicitado.

4. Além dos resultados pedidos em consonância com **o solicitado** serão avaliados:

- Uso apropriado do tipo de variável para cada situação
- Endentação
- Tamanho do código
- Declaração das funções (protótipo)

5. **Não podem ser utilizadas variáveis globais.**



**QUESTÃO 1** – O objetivo da regressão linear é determinar a “melhor” reta que se ajuste a um conjunto de dados. Pelo Método dos Mínimos Quadrados é possível determinar uma função aproximada que ajuste a tendência geral dos dados sem necessariamente passar pelos pontos individuais.

O exemplo mais simples é o **ajuste de uma reta** a um conjunto de pontos  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ :

$$y = a_0 + a_1 x$$

Os coeficientes  $a_0$  e  $a_1$  da reta de ajuste podem ser calculados por:

$$a_1 = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \cdot \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2}$$

e

$$a_0 = \bar{y} - a_1 \bar{x}$$

onde  $\bar{x}$  e  $\bar{y}$  são as médias dos valores de  $x$  e  $y$ , respectivamente.

Em um arquivo binário foram armazenadas alguns pontos (no máximo 10) segundo a seguinte estrutura:

```
struct pontos {  
    float x;  
    float y;  
};
```

**Pede-se, um programa em C que**

1. Leia os números gravados no arquivo **dados.dat** e armazene em um vetor do tipo **struct pontos**.

2. Imprima organizadamente na tela os pontos contidas no arquivo binário. Por exemplo:

Pontos lidos: (1.1,2.4), (2.0,4.5), (3.2,8.4), (5.5,12.8)

3. Contenha uma **função** que receba o **endereço inicial do vetor de estrutura** contendo os dados e o **número de pontos** lidos e **devolva o coeficientes  $a_1$**  da reta de ajuste para o programa principal.

4. Contenha uma **função** que receba o **endereço inicial do vetor de estrutura** contendo os dados, o **número de pontos lidos**, o **coeficiente  $a_1$**  e **devolva o coeficiente  $a_0$**  da reta de ajuste para o programa principal. Para calcular o coeficientes  $a_0$  essa função deve chamar duas vezes a função do item 5.

5. Contenha uma **função** que retorne a média de um conjunto de valores. Essa função deve receber o **endereço do campo  $x$  ou  $y$  do primeiro elemento do vetor de estrutura** contendo os dados e o **número de pontos lidos**.

6. Imprima no programa principal a equação da reta obtida, por exemplo:

Reta de ajuste:  $y = 2 - 3x$

**QUESTÃO 2** – Dados um vetor de números inteiros declarado no próprio código, elabore um programa em C que determine em que elemento a soma sequencial dos elementos do vetor se torna máxima e qual é o valor dessa soma. O Programa deve funcionar qualquer que seja a dimensão do vetor declarado.

A resposta deve ser gravada em um arquivo texto.

Por exemplo, para o vetor

```
int vetor[] = {5, 2, -2, -7, 3, 14, 10, -3, 10, -6, 4, 1 };
```

O arquivo deverá conter

A soma se torna máxima no nono elemento e vale 32.

**QUESTÃO 3** – Elabore um programa em C que leia um número inteiro **n** e um número inteiro de apenas um dígito **d**.

- Faça uma função que receba **d** e **n** e devolva quantas vezes **d** aparece em **n**.
- O programa principal imprime o resultado.
- Caso o usuário digite um número **d** com mais de um dígito ou qualquer outro caractere que não seja número, o programa deve solicitar que o número seja digitado novamente até o usuário digitar um número **d** de um dígito apenas.

Por exemplo,

Com **n = 543214** o dígito **d = 4** aparece 2 vezes.

***Boa Prova !***