

## Prova de AEDs I - valor 25 ptos

Nome:

Data:

## Questão 1: (8 pontos)

Faça um programa em C que: a) Crie uma estrutura de dados chamada “Ponto”, que armazena um ponto do plano cartesiano. Um ponto é composto pelo par  $(x, y)$ , ambas do tipo *double*. b) Crie uma função chamada “distancia”, que receba como parâmetros duas variáveis do tipo “Ponto” e retorne o valor da distância euclidiana entre eles computada por:

$$distancia(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}, \text{ sendo } P_1(x_1, y_1) \text{ e } P_2(x_2, y_2)$$

Utilize a função *sqrt* da biblioteca “math.h” para calcular a raiz quadrada. c) Crie uma função chamada “maisProximoOrigem”, que leia, via *scanf*, *n* pontos e retorne o “Ponto” mais próximo à origem (0.0,0.0) do plano cartesiano. Esta função deve fazer uso da função “distancia”. d) No programa principal, o ponto mais próximo da origem deve ser gravado em arquivo.

## Questão 2: (6 pontos)

Crie um programa em C que faça uso de uma função chamada “InverteVetorAchaMaior”, que recebe como parâmetro dois vetores  $v_1$  e  $v_2$ , ambos de tamanho *n*. A função deve copiar os elementos de  $v_1$  para  $v_2$  na ordem inversa. Por exemplo, se  $v_1 = \{1, 2, 3, 4, 5\}$  a função registra  $v_2 = \{5, 4, 3, 2, 1\}$ . Além disso, a função também deve retornar o maior valor encontrado em  $v_1$ . A função deve obedecer ao seguinte protótipo:

```
int InverteVetorAchaMaior(int *v1, int *v2, int n);
```

## Questão 3: (5 pontos)

Considere equivalentes as sintaxes 1 e 2:

Sintaxe 1:

```
return (vetor[i]<XResto) ? vetor[i] : XResto;
```

Sintaxe 2:

```
if (vetor[i]<XResto)
    return vetor[i];
else
    return XResto;
```

Qual a finalidade do programa *main*? Apresente o resultado utilizando as entradas da tabela abaixo.

4	-5	12	-7	1
---	----	----	----	---

```
#include <stdio.h>
```

```
int encontrarX(int vetor[], int i, int n) {
    if (i == n - 1) {
        return vetor[i];
    }
    int XResto = encontrarX(vetor, i+1, n);
    return (vetor[i]<XResto) ? vetor[i] : XResto;
}
```

```
int main() {
    int n;
    scanf("%d", &n);
    int vetor[n];
    for (int i = 0; i < n; i++) {
        scanf("%d", &vetor[i]);
    }
    int X = encontrarX(vetor, 0, n);
    printf("O X eh: %d\n", X);
    return 0;
}
```

## Questão 4: (6 pontos)

Faça um programa em C que calcule o desvio padrão  $\sigma$  de um vetor  $v$  contendo *n* números *float* lidos do teclado. Desenvolva a função média aritmética “mediaV” e compute o  $\sigma$  pela seguinte expressão:

$$\sigma = \sqrt{\frac{1}{n-1} \sum (v[i] - mediaV(v, n))^2}$$

O programa deve imprimir a média e o desvio padrão de  $v$ .