

## Programação de Computadores II - BCC702

### Lista de Exercícios

#### Ponteiros e Alocação Dinâmica de Memória

##### **Exercício 01**

Faça um programa que leia o tamanho de um vetor e seus elementos. Em seguida guarde em variáveis: o endereço da variável tamanho; o endereço do vetor; o endereço do primeiro elemento do vetor; o valor do primeiro elemento do vetor; o endereço do último elemento do vetor. Por fim, escreva o valor de cada uma dessas variáveis (os endereços serão diferentes para cada execução).

```
Informe o tamanho do vetor: 3
Informe os elementos do vetor: 5 1 8

Endereço de tamanho: 0x7fff18f7a41c
Endereço do vetor: 0x7fff18f7a3f0
Endereço do primeiro elemento: 0x7fff18f7a3f0
Valor do primeiro elemento: 5
Endereço do último elemento: 0x7fff18f7a3f8
```

##### **Exercício 02**

Para cada opção abaixo, qual é o valor variável *r*?

- a)        `int x = 5, y = 2;`  
          `int *z = &x;`  
          `int *w = new int;`  
          `*w = *z;`  
          `int r = x + y + *w + *z;`
- b)        `bool b1 = true, b2 = false, b3 = true;`  
          `bool *c = &b2;`  
          `bool d = b3;`  
          `bool r = b1 and *c and d;`
- c)        `double a = 2.3, *b = &a, c = 4.5;`  
          `double v[3] = { 1.1, 2.2, 3.3};`  
          `double *x = new double[3];`  
          `double *y = v;`  
          `x[0] = a;`  
          `x[1] = *b;`  
          `x[2] = y[2] + c;`  
          `double r = x[0] + x[1] + x[2];`

### **Exercício 03**

Faça um programa que preenche um vetor de N números inteiros. O programa deve ter uma função que recebe o vetor e o seu tamanho N como parâmetros e retorna a moda do conjunto (o valor que mais se repete). Se não houver valor repetido, a função deve retornar -1.

A função deve ser chamada na main, onde deve ser exibido o valor da moda.

Informe o número de elementos do vetor: 7

Informe os elementos do vetor: 8 3 3 2 8 7 3

A moda é: 3

### **Exercício 04**

Faça um programa com um registro *Ponto*, que contém os campos inteiros x e y. Faça uma função que recebe por parâmetro dois pontos, calcula e retorna a distância euclidiana entre eles (função  $\text{sqrt}(x)$  da biblioteca *math* = retorna a raiz quadrada de x).

Na função *main*, crie um vetor de pontos alocado dinamicamente de tamanho N (N informado pelo usuário) e leia as coordenadas de cada ponto. Use a função definida para calcular o tamanho do caminho definido pelos pontos, na ordem em que foram lidos.

Informe o número de pontos: 4

Informe a coordenada x: 3

Informe a coordenada y: 3

Informe a coordenada x: 1

Informe a coordenada y: 1

Informe a coordenada x: -1

Informe a coordenada y: 2

Informe a coordenada x: 0

Informe a coordenada y: 3

Tamanho do caminho: 6.47871