

IFRN – Câmpus Currais Novos
Fundamentos de Programação – Prof. Bruno Gomes
Exercícios de revisão
Funções, vetores e matrizes

1. Funções sobre números:

- a) **float** pot (**float** x, **float** y) – potencia de x por y (x^y).
- b) **float** max (**float** x, **float** y) – maior valor entre dois números.
- c) **float** min (**float** x, **float** y) – menor valor entre dois números.
- d) **float** abs (**float** x) – retorna o valor absoluto de um número. Se o argumento recebido for não-negativo, ele é retornado, caso contrário a sua negação é retornada.
- e) **int** sucessor (**int** x) – retorna o próximo valor inteiro ($x+1$).
- f) **int** predecessor (**int** x) – retorna o número inteiro anterior ao fornecido ($x - 1$).

2. Funções sobre caracteres e cadeias de caracteres:

- a) **bool** eh_branco (**char** ch) – retorna verdadeiro (*true*) se o caractere for espaço em branco e falso (*false*) caso contrário.
- b) **int** contar_palavras (**char**[] str) – conta quantas palavras existem em um texto. Considere espaços em branco como separador de palavras.
- c) **int** comparar (**char**[] str1, **char**[] str2) – compara duas cadeias de caracteres. Retorna 0 se elas forem iguais, 1 se a primeira for maior que a segunda e -1 se a primeira for menor que a segunda.

3. Funções sobre matrizes (tamanho até 10x10):

OBS.: Para indicar a passagem de uma matriz a uma função, deve-se colocar o índice do tamanho máximo da coluna (no caso deste exercício, 10). Não é necessário usar colocar o valor para a linha.

Exemplo (resolvido na próxima página):

/ m[][10] – matriz a ser recebida (até 10x10).*

** tam – tamanho do vetor. */*

float maior_elemento (**float** m[][10], **const short** tam) – retorna o maior elemento armazenado na matriz.

- a) **bool** eh_identidade (**float** m[][10], **const short** tam) – verifica se a matriz passada como argumento é ou não identidade. A matriz identidade possui todos os elementos da diagonal principal iguais a zero.
- b) **void** gera_transposta (**float** m[][10], **const short** tam) – gera a matriz transposta da matriz fornecida como argumento.
- c) **bool** eh_simetrica (**float** m[][10], **const short** tam) – retorna se a matriz é ou não simétrica. Uma matriz é simétrica se ela é igual a sua transposta.

```

#include <iostream>
using namespace std;

//Assinatura (protótipo) da função "maior_elemento"
float maior_elemento (float m[][10], int tam);

int main () {
    float mat[10][10]; //declaração da matriz
    int ordem = 0;    //tamanho da matriz

    do {
        cout << "Ordem matriz (até 10): ";
        cin >> ordem;
    } while (ordem < 1 || ordem > 10);

    for (int i = 0; i < ordem; i++) {
        for (int j = 0; j < ordem; j++) {
            cout << "mat[" << i+1 << "]" << j+1 << "] = ";
            cin >> mat[i][j];
        }
    }

    //Chamada à função 'maior elemento' e impressão do resultado
    cout << "O maior elemento é: " << maior_elemento(mat, ordem);

    return 0;
}

//Declaração da função maior elemento
float maior_elemento (float m[][10], int tam) {
    float maior = m[0][0];

    for (int i = 0; i < tam; i++) {
        for (int j = 0; j < tam; j++) {
            if (maior < m[i][j]) {
                maior = m[i][j];
            }
        }
    }

    return maior;
}

```