

SÉTIMA LISTA DE EXERCÍCIOS DE PROGRAMAÇÃO II

Matéria da Segunda Prova

- 1) Elabore uma função que receba duas strings como parâmetros e verifique se a segunda string ocorre dentro da primeira.
- 2) Crie um tabuleiro de jogo da velha, usando uma matriz de caracteres (char) 3x3, onde o usuário pede o número da linha (1 até 3) e o da coluna (1 até 3). A cada vez que o usuário entrar com esses dados, colocar um 'X' ou 'O' no local selecionado.
 - a. Não permita jogadas fora dos valores ou em lugares já preenchidos;
 - b. Não é necessário informar o vencedor, apenas indique que não há mais lugares possíveis para se jogar.
- 3) Dizemos que uma matriz quadrada inteira é um quadrado mágico se a soma dos elementos de cada linha, a soma dos elementos de cada coluna e a soma dos elementos das diagonais principal e secundária são todas iguais. Leia uma matriz 10x10 do usuário e imprima se ela é um quadro mágico ou não.
- 4) Escreva uma função chamado AUMENTO que receba dois valores reais X e Y como parâmetros e aumente o valor de X em Y%, retornando o novo valor de X.
- 5) Faça um programa que permita ao usuário entrar com uma matriz de 3 x 3 números inteiros. Em seguida, gere um vetor pela soma dos números de cada coluna da matriz e mostre esse vetor na tela. Por exemplo, a matriz:

$$\begin{bmatrix} 5 & -8 & 10 \\ 1 & 2 & 15 \\ 25 & 10 & 7 \end{bmatrix}$$

Gerará um vetor onde cada posição é a soma das colunas da matriz. Ficando da seguinte forma:

$$[31 \quad 4 \quad 3]$$

- 6) Faça um programa que leia uma matriz de 5 linhas e 4 colunas contendo as seguintes informações sobre alunos de uma disciplina, sendo todas as informações do tipo inteiro:
 - a. Primeira coluna: número de matrícula
 - b. Segunda coluna: média das provas
 - c. Terceira coluna: média dos trabalhos
 - d. Quarta coluna; nota final

Elabore um programa que:

- i) Leia as três primeiras informações de cada aluno (matrícula, média das provas e média dos trabalhos)

- ii) Calcula a nota final como sendo a soma da média das provas e da média dos trabalhos
- iii) Imprima a matrícula do aluno que obteve a maior nota final (assuma que só existe uma maior nota)
- iv) Imprima a média aritmética das notas finais

Ponteiros

- 7) Escreva um programa que declare um inteiro, um float e um char, e ponteiros para inteiro, float, e char. Leia os valores do inteiro, float e char do teclado, armazene nas variáveis e depois associe as variáveis aos ponteiros (use &). Modifique os valores de cada variável usando os ponteiros. Imprima os valores das variáveis antes e após a modificação.
- 8) Faça um programa que leia 2 valores inteiros e chame uma função que receba estas duas variáveis e troque o seu conteúdo, ou seja, esta função é chamada passando duas variáveis A e B e, após a execução da função, A conterá o valor de B e B terá o valor de A.
- 9) Implemente uma função que calcule a área da superfície e o volume de uma esfera de raio R. Essa função deve possuir a seguinte assinatura:

```
void calc_esfera(float R, float *area, float *volume);
```
- 10) Crie um programa que contenha um array de inteiros contendo 5 elementos. Utilizando apenas aritmética de ponteiros, leia os valores desse array do teclado e imprima o dobro de cada valor lido.
- 11) Escreva uma função que aceite como parâmetro um array de inteiros com N valores, e determina o maior elemento do array e o número de vezes que este elemento ocorreu no array. Por exemplo, para um array com os seguintes elementos: 5, 2, 15, 3, 7, 15, 8, 6, 15, a função deve retornar para o programa que a chamou o valor 15 e o número 3 (indicando que o número 15 ocorreu 3 vezes). A função deve ser do tipo **void**.
- 12) Implemente uma função que receba como parâmetro um array de números reais de tamanho N e retorne quantos números negativos há nesse array. Essa função deve obedecer ao protótipo:

```
int negativos(float *vet, int N);
```
- 13) Crie uma função para somar dois arrays. Esta função deve receber dois arrays e retornar a soma em um terceiro array. Caso o tamanho do primeiro e segundo array seja diferente então a função retornará ZERO (0). Caso a função seja concluída com sucesso a mesma deve retornar o valor UM (1). Utilize aritmética de ponteiros para manipulação do array.

- 14) Crie um programa para calcular a área e o perímetro de um hexágono. O seu programa deve implementar uma função chamada `calcula_hexagono` que calcule a área e o perímetro de um hexágono regular de lado L . A função deve obedecer o seguinte protótipo:

`void calcula_hexagono(float L, float *area, float *perimetro);`

Lembrando que a área e o perímetro de um hexágono regular são dados por:

$$A = \frac{3l^2\sqrt{3}}{2} \quad P = 6l$$

Para os cálculos, obrigatoriamente você deve utilizar as funções `sqrt` e `pow` da biblioteca `math.h`. Em seguida crie a função principal do programa e utilize a função `calcula_hexagono` para calcular a área e o perímetro de um hexágono de lado L informado pelo usuário.

- 15) Crie um programa para manipular vetores. O seu programa deve implementar uma função chamada `inverte_vetor`, que recebe como parâmetro dois vetores $V1$ e $V2$, ambos de tamanho N . A função deve copiar os elementos de $V1$ para $V2$ na ordem inversa. Ou seja, se a função receber $V1 = \{1,2,3,4,5\}$, a função deve copiar os elementos para $V2$ na seguinte ordem: $V2 = \{5,4,3,2,1\}$. Além disso, a função também deve retornar o maior valor encontrado em $V1$. A função deve obedecer ao seguinte protótipo:

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