

Trabalho de estrutura de dados.

Valor 4.0 ponto.

Nome: _____

- O trabalho é individual;
- Todos os programas devem ser modularizados;
- Entregar até o dia da prova-A1, via Gitlab.

1. Faça um programa que modularize o código a seguir:

```
#include <iostream>

using namespace std;

const int tamanho =10;

int* vetor[tamanho];

int main()
{
    cout << "Inicializando o vetor " << endl;
    for (int i=0; i < tamanho; i++){
        vetor[i]=new int;
        *(vetor[i])=0;
    }
    cout << "Leia os valores do vetor" << endl;
    for (int i=0; i < tamanho; i++){
        cout << "Posicao "<<(i+1)<<": ";
        cin >> *(vetor[i]);
    }
    cout << "Imprima os valores do vetor "<< endl;
    for (int i=0; i < tamanho; i++){
        cout << "Posicao "<<(i+1)<<": "<<*(vetor[i])<<endl;
    }
    cout << "Fim do Programa!!"<< endl;
    return 0;
}
```

2. Faça um programa que leia uma matriz 3x3 e apresente na tela o menor valor da matriz.

3. Fazer um programa que leia duas matrizes 3x3 e faça as seguintes operações:

- a) Soma
- b) Multiplicação

4. Desenvolva um programa que leia dez valores de um vetor de inteiro e apresente:

- a) O maior valor;
- b) A média dos valores;
- c) O menor valor;

5. Fazer um sistema que preencha um vetor com cinco registros. A estrutura do registro é apresentada a seguir:

```
struct registro {
    int codigo;
    char nome[60];
    char endereco[60];
};
```

Após o preenchimento dos cinco registros, fazer uma rotina (função) que pesquise os registros por código e apresente as informações do registro na tela.

6. Faça um programa que receba um vetor com 15 posições e faça a escrita dos valores ao contrário das posições iniciais. Porém, **o programa deve percorrer o vetor e fazer a inversão usando ponteiros**
7. Faça um programa que crie um vetor em tempo de execução, pedindo o tamanho do vetor e preenchendo os valores. Após o preenchimento, apresentar os valores na tela do sistema.
8. Implemente o algoritmo de ordenação *selection sort* e aplique a um vetor com vinte posições. Os valores do vetor devem ser lidos pelo usuário e apresentados, ordenados, na tela do programa.
9. Desenvolva um programa que ordene um vetor de inteiro com dez posições. Os valores devem ser lidos no terminal. Usar o bubble sort.
10. Faça um programa que preencha dez posições de um vetor da estrutura a seguir:

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

E realize as seguintes operações:

- a) Apresente o maior valor de x entre os "pontos"(vetor);
 - b) Apresente o menor valor de y entre os "pontos"(vetor);
 - c) Dado um valor, imprimir em qual posição se encontra este valor. Se o valor estiver em várias posições, apresentar todas as posições em que encontra determinado valor.
11. Faça um programa que receba um vetor com 10 posições e peça para que o usuário escolha as seguintes opções de ordenamento:
- a) selection sort
 - b) insertion sort
 - c) bubble sort

Após a escolha do algoritmo, os valores devem ser ordenados pelo algoritmo selecionado e apresentados na tela.

12. Faça um programa que leia 10 posições de caracteres e os imprima em ordem inversa. Este programa deve usar uma **pilha**
13. Faça um programa que leia um vetor de ponteiros de inteiros e faça a sua ordenação usando o algoritmo bubble sort.

14. Faça um programa que leia um vetor de 10 posições e realize a ordenação pelo algoritmo quicksort.
15. Faça um programa que tenha um menu com as seguintes opções (as opções devem estar funcionais):
- Criar um vetor de ponteiros de structs com as seguintes informações: nome (char), idade (int), endereço (char). Determinar o tamanho do vetor quando esta opção for escolhida.
 - Apagar uma posição do vetor de estruturas. O usuário digita um valor que será aquele a ser apagado.
 - Localizar a estrutura que contenha a pessoa com a maior idade cadastrada.
 - Inserir uma struct em uma determinada posição do vetor. Verificar se já não existe uma struct na mesma posição.
 - Ordenar a struct usando o algoritmo selection sort. Ordenar pela idade.
 - Imprimir as informações das structs armazenadas no vetor.

Observação: usar um vetor de ponteiros de struct.