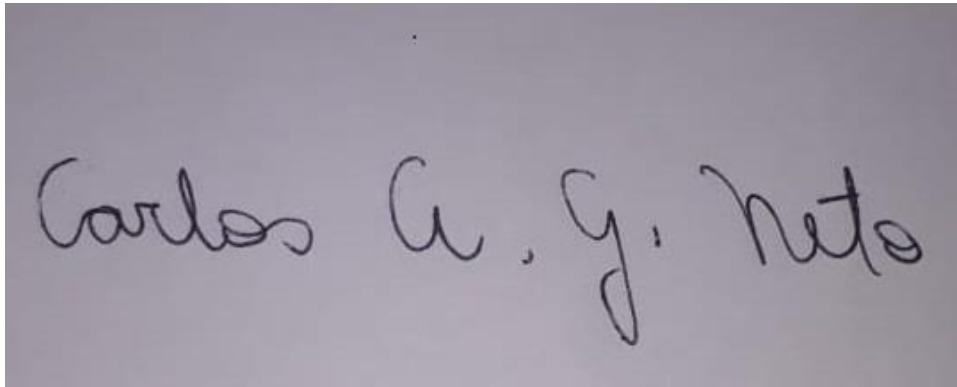


Nome: Carlos Augusto Gomes Neto

Matrícula: 11821BCC016

### Declaração de Idoneidade da Prova

“Eu afirmo que todas as respostas desta prova são de minha autoria. Durante todo o tempo em que esta prova esteve comigo, isto é, até antes de eu devolver esta prova no Teams®, eu não conversei por meio escrito ou oral, por qualquer meio de comunicação, com qualquer outra pessoa (física ou jurídica) sobre qualquer questão desta prova.”

A photograph of a handwritten signature in dark ink on a light-colored surface. The signature reads "Carlos A. G. Neto" in a cursive, slightly slanted script.

/\* Nome: Carlos Augusto Gomes Neto

Matrícula: 11821BCC016

Este programa apresenta uma matriz lendo suas informações em um arquivo.

O programa inicialmente irá abrir um arquivo que possui informações como a ordem da matriz e seus elementos.

O programa irá ler a ordem e os elementos da matriz, imprimir ao usuário a matriz lida

Através de funções, será mostrado a diagonal primária, secundária e a soma dos elementos de uma determinada linha da matriz

O programa deve ficar em um loop onde o usuário irá requisitar o que quiser o tanto de vezes que quiser

O programa encerra se ele escolhe a opção sair ( opção 4)

Criado ano de 2020.

\*/

#include <stdlib.h>

```

#include <stdio.h>
#include <string.h>
#include <ctype.h>
#define MAX1 10
#define MAX2 100

//-----Protótipo funções-----
void dp(int m, float a[MAX1][MAX1]);
void ds(int m, float a[MAX1][MAX1]);
int tl( int m, float a[MAX1][MAX1], int x);

int
main()
{
    FILE *abrir = NULL;                //variavel do tipo file
    float a[MAX1][MAX1];               //representa a matriz
    char arquivo[MAX2];                //string que guarda o nome
do arquivo digitado
    int opcao = 0;                      //representa a opcao do
menu
    int m = 0;                          //índice da matriz
quadrada
    int i = 0;                          //contador de laço 1
    int j = 0;                          //contador de laço 2
    int x = 0;                          //representa a linha que o
usuario quer fazer a soma
    int r1 = 0;                         //representa o retorno da
funcao tl

    printf("Digite o nome do arquivo: "); //usuário digita nome do
arquivo
    scanf("%s", arquivo);               //armazena em um vetor o
nome digitado

```

```

    abrir = fopen(arquivo,"r");                                //abertura do arquivo

    if(abrir == NULL)                                          //caso haja falha
ao abrir o arquivo
    {
        printf("Nao foi possivel abrir o arquivo");
        exit(1);
    }

    fscanf(abrir, "%d", &m);                                //captura da ordem da matriz

    for (i = 0, j = 0;!feof(abrir);i++, j = 0)                //laço para
mudança de linha e para manter o próximo laço funcionando até que o
arquivo seja completamente lido
    {
        for (;j < m;j++)                                      //para mudar de
coluna
            fscanf(abrir, "%f", &a[i][j]);                    //carregamento da
matriz
            fscanf(abrir, " ");                                //para evitar
que o programa leia o fim do arquivo e dê erros
    }

    fclose(abrir);                                            //fecha o arquivo
    abrir == NULL;                                            //libera memoria do
ponteiro

    printf("\nA matriz: \n");
    for ( i = 0; i < m; i++)                                  //impressão da matriz
    {
        for ( j = 0; j < m; j++)
            printf("%.0f ", a[i][j]);
        printf("\n");
    }

```

```

        while ( opcao != 4)                //enquanto opcao for diferente de 4
(sair), teremos o loop
    {
        printf("\n\nSelecione uma opcao (digite o numero):\n1 - DP\n2 -
DS\n3 - TotalLinha\n4 - Sair\n");
        scanf("%d", &opcao);                //armazena um numero na variavel opcao

        if ( opcao == 1)                    //caso opcao seja = 1
            dp(m,a);                        //chamada a função dp

        if ( opcao == 2)                    //caso opcao seja = 2
            ds(m,a);                        //chamada a funcao ds

        if ( opcao == 3)                    //caso opcao seja = 3
        {
            printf("Digite a linha que quer saber a soma dos
elementos:\n");
            scanf("%d", &x);                //usuario define a linha que
quer saber a soma

            r1 = tl(m,a,x);                  //chamada a funcao tl
            printf("\n\nA soma da linha %d: %d", x,r1);    //imprime o que
a funcao tl retornou
        }
    }

    return 0;
}

```

//dp

//-----

//Apresenta ao usuário os elementos da diagonal principal

//A funcao recebe a ordem da matriz quadrada e a matriz

```

void dp(int m, float a[MAX1][MAX1])
{
    int i = 0;
    int j = 0;
    printf("\n\nA Diagonal principal: \n");
    for ( i = 0; i < m; i++)          //impressão da diagonal
principal
    {
        for ( j = 0; j < m; j++)
        {
            if ( i == j)              //condicao para imprimir elementos
da diagonal principal
                printf("%.0f ", a[i][j]);
        }
    }
}

//ds
//-----
//A funcao recebe como parametro a ordem da matriz e a propria matriz
//Irá apresentar ao usuário a diagonal secundária da matriz
void ds(int m, float a[MAX1][MAX1])
{
    int i = 0;
    int j = 0;
    printf("\n\nA Diagonal secundaria: \n");
    for ( i = 0; i < m; i++)          //impressão da diagonal
secundaria
    {
        for ( j = 0; j < m; j++)
        {
            if ( j == m - 1 - i)      //condicao para imprimir
elementos da diagonal secundaria

```

```

        printf("%.0f ", a[i][j]);
    }
}

//t1
//-----
//A funcao recebe como parametro a ordem da matriz, a propria matriz e x
//que é a linha de interesse
//Retorna ao usuário o valor da soma dos elementos da linha requisitada
int t1( int m, float a[MAX1][MAX1], int x)
{
    int soma = 0;           //representa o que a funcao retornara
    int i = 0;              //contador 1
    int j = 0;              //contador 2
    i = x;                  //variavel auxiliar

    while ( j < m)          //i fixo, j variando para percorrer a linha
    {
        soma = soma + a[i-1][j];    //soma recebe cada elemento da
linha
        j++;
    }

    return soma;            //retorna o resultado
}

```