

### ***Exercícios Complementares 3***

1. Escreva um programa em C que preencha uma matriz 4X4 com "0" e a sua diagonal principal com "1". Em seguida, exiba a matriz na tela (uma linha da matriz por vez).

Resolução:

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
const int n=4;
```

```
int i, j;
```

```
void preenche(int m[n][n]);
```

```
void mostra_mat(int m[n][n]);
```

```
main()
```

```
{
```

```
    int mat[n][n];
```

```
    //Entrada de dados automática
```

```
    preenche( mat );
```

```
    //Saída de dados
```

```
    printf(" \n\n\tMatriz Identidade %d x %d\n\n", n, n);
```

```
    mostra_mat(mat);
```

```
    system("pause");
```

```
}
```

```
void preenche(int m[n][n])
```

```
{
```

```
    for(i=0;i<n; i++)
```

```
        for(j=0;j<n;j++)
```

```
            if(i==j)
```

```
                m[i][j]=1;
```

```
            else
```

```
                m[i][j]=0;
```

```
}
```

```
void mostra_mat(int m[n][n])
```

```
{
```

```
    for(i=0;i<n; i++)
```

```
    {
```

```
        for(j=0;j<n;j++)
```

```
            printf("%5d", m[i][j]);
```

```
        printf("\n\n");
```

```
    }
```

```
}
```

2. Fazer um programa C modularizado que leia uma matriz 15 x 15 de reais e calcule a soma dos elementos da diagonal secundária. Mostrar a soma ao final.
3. Fazer um programa C modularizado que calcule a média dos elementos da diagonal principal de uma matriz 10 x 10 de inteiros.
4. Fazer um programa C modularizado que encontre (e mostre) o maior valor de uma matriz 10 x 5 de inteiros positivos.

Resolução:

```
int encontra_maior( int m[lin][col]) //const int lin=10, col=5;
{
    int maior=m[0][0];
    for(i=0;i<lin;i++)
        for(j=0;j<col;j++)
            if ( m[i][j] > maior )
                maior = m[i][j];
    return(maior);
}
```

5. Refaça o programa do exercício anterior de forma que a posição do maior elemento seja retornada.
6. Para um professor que tenha 5 turmas de 30 alunos cada, escreva um programa C modularizado que leia as médias dos alunos de cada turma e calcule a média de cada turma. Em seguida, exiba o resultado.
7. Faça um programa C modularizado que leia os nomes de 40 alunos de uma turma de Programação de Computadores II e imprima os nomes dos alunos que tenham iniciais 'A'.
8. Faça um programa C modularizado que calcule a soma de dois vetores unidimensionais de 40 posições. Guarde a resposta em um terceiro vetor. (Simule os vetores com uma matriz 3X40)
9. Faça um programa C modularizado para calcular a soma de duas matrizes de ordens 5 x 3. Guarde a resposta numa terceira matriz e mostre-a na tela.
10. Faça um programa C modularizado que calcule e mostre a multiplicação de duas matrizes de ordens **m** x **n** e **n** x **n**, respectivamente. O programa deve guardar o produto numa terceira matriz **m** x **n**. Para fins de exemplo suponha m = 3 e n = 2.

11. Faça um programa C modularizado que receba as vendas semanais ( de um mês ) de 5 vendedores de uma loja e armazene essas vendas em uma matriz.

Calcule e imprima:

- . total de vendas do mês ( 4 semanas) de cada vendedor;
- . total de vendas de cada semana (todos os vendedores juntos);
- . total de vendas do mês.

Resolução:

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
const int lin = 5, col= 4;
```

```
int i,j;
```

```
void le_mat(float mv[lin][col])
```

```
{
```

```
    printf( "Digite o valor das vendas dos vendedores por semana:\n");
```

```
    for (i=0;i<lin;i++)
```

```
    {
```

```
        for (j=0;j<col;j++)
```

```
        {
```

```
            printf( "\n\t%d do vendedor na semana %d: ",i+1,j+1);
```

```
            scanf("%f", &mv[i][j]);
```

```
        }
```

```
        printf("\n\n");
```

```
    }
```

```
}
```

```
void calc_tot_vend(float mv[lin][col], float *vv)
```

```
{
```

```
    for (i=0;i<lin;i++)
```

```
    {
```

```
        vv[i]=0;
```

```
        for (j=0;j<col;j++)
```

```
            vv[i] += mv[i][j];
```

```
    }  
}
```

```
void calc_tot_sem(float m[lin][col], float *vs)  
{  
    for (j=0;j<col;j++)  
    {  
        vs[j]=0;  
        for (i=0;i<lin;i++)  
            vs[j] += mv[i][j];  
    }  
}
```

```
float calc_tg( float *vs )  
{  
    float total=0;  
    for (j=0;j<col;j++)  
        total += vs[j];  
    return(total);  
}
```

```
void mostra_tot_v(float *vv)  
{  
    printf("\n\nTotal vendas do por vendedor: \n\n");  
    for(i=0;i<lin;i++)  
        printf("\n\t%do Vendedor: R$ %.2f", vv[i]);  
    printf("\n\n");  
}
```

```
void mostra_tot_s_tg( float*vs, float tg )  
{  
    printf("\n\nTotal vendas do por semana: \n\n");  
    for(j=0;j<col;j++)  
        printf("\n\t%da Semana: R$ %.2f", vs[j]);  
    printf("\n\n");  
}
```

```

        printf("\n\nTotal geral de vendas e: %8.2f\n\n",tg);
    }

main()
{
    float matriz_vendas[lin][col], vet_vend[lin], vet_sem[col], total_geral;

    le_mat(matriz_vendas);
    calc_tot_vend( matriz_vendas, vet_vend);
    calc_tot_sem(matriz_vend, vet_sem);
    total_geral = calc_tg(vet_sem);
    system("cls");//Limpa a tela para mostrar a saída de dados
    mostra_tot_v( vet_vend);
    mostra_tot_s_tg( vet_sem, total_geral );
    system("pause");
}

```

12. Faça um programa C modularizado que carregue uma matriz 2 X 2 com números inteiros, calcule e imprima a soma dos elementos da diagonal principal.

13. Dadas duas matrizes numéricas A e B de dimensão 4x3, fazer um programa C modularizado que gere uma matriz de zeros e uns denominada C, tal que o elemento C[i,j] seja verdadeiro (1 na linguagem C) se os elementos nas posições respectivas das matrizes A e B forem iguais e falso (0 na linguagem C) caso contrário. Exibir na tela as matrizes A, B e C:

EXEMPLO:

	2	4	6		2	5	8		1	0	0
A =	1	5	9	B =	1	9	7	C =	1	0	0
	3	7	2		3	7	1		1	1	0
	4	6	8		4	5	8		1	0	1

14. Dada uma matriz contendo 4 notas de 10 alunos, elabore um programa C modularizado que calcule e exiba um vetor (unidimensional) que deverá conter a média aritmética das 4 notas de cada aluno.

15. Elabore um programa C modularizado que receba o estoque de 4 produtos(colunas) que estão armazenados em 5 armazéns (linhas) e coloque estes dados em uma matriz 5 por 4. Calcule e imprima:

- a) a quantidade de itens armazenados em cada armazém;
- b) qual armazém possui a maior quantidade de itens em estoque;
- c) qual armazém possui a menor quantidade de itens em estoque.