

Aula Prática Revisora 3

Desenvolver os seguintes programas em Linguagem “C”, **INDIVIDUALMENTE!**

1. Dada uma matriz $A_{m \times n}$, imprimir o número de linhas e o número de colunas nulas da matriz.

Exemplo: $m = 4$ e $n = 4$

$$\begin{pmatrix} 1 & 0 & 2 & 3 \\ 4 & 0 & 5 & 6 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ tem 2 linhas nulas e 1 coluna nula.}$$

2. 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.

Exemplo: A matriz

$$\begin{pmatrix} 8 & 0 & 7 \\ 4 & 5 & 6 \\ 3 & 10 & 2 \end{pmatrix} \text{ é um quadrado mágico de soma 15.}$$

Dado um número $n \leq 5$, e uma matriz quadrada $A_{n \times n}$, verificar se A é um quadrado mágico. Utilize as funções de ler matriz e mostrar matriz desenvolvidas na lista revisora 2.

3. Deseja-se atualizar as contas correntes dos clientes de uma agência bancária, contendo, para cada cliente, o número de sua conta e o seu saldo. Faça um programa que leia um número $n \leq 20$, e em seguida leia os códigos das contas e os saldos iniciais destes n clientes na variável “cadastro”. Em seguida, deve ser lido o número $m \leq 100$ de operações efetuadas no dia e, para cada operação, o número da conta, uma letra **C** ou **D** indicando se a operação é de **Crédito** ou **Débito**, e o valor da operação. Utilize as estruturas abaixo, e crie as funções indicadas:

```
struct tipo_conta
{
    int cod_conta;
    float saldo;
} cadastro[20];

struct tipo_operacao
{
    int cod_conta;
    char tipo;
    float valor;
}opers[100];
```

```
void ler_dados_cadastro(struct tipo_conta cadastro[20],int n);
void ler_operacoes(struct tipo_operacaoopers[100], int m);
void atualiza_saldo(struct tipo_conta cadastro[20],int n,struct tipo_operacaoopers[100], int m);
void mostra_saldos(struct tipo_conta cadastro[20],int n);
```

4. Altere o programa anterior para carregar os dados do cadastro de clientes de um arquivo de nome “cadastro.dat”, e ler as operações de um arquivo de nome “movimento.dat”, gravando os dados de cadastro contendo os saldos atualizados no arquivo “cadatual.dat”.