

# Lab. 10

## Matriz

ECT2303 - T02 - 19.1

Neste laboratório, será enfatizado o uso de matrizes com duas dimensões, conhecidas também por matrizes bidimensionais ou arrays bidimensionais.

Uma matriz com duas dimensões costuma ser utilizada para representar tabelas de valores consistindo em informações organizadas em **linhas** e **colunas**. Em geral, um array de  $m$  linhas e  $n$  colunas é chamado de array  $m$  por  $n$ .

### 10.1 Declaração de Matriz

Uma matriz de duas dimensões pode ser declarada da seguinte forma:

```
tipo nome[dimensao1][dimensao2];
```

Cada posição da matriz `a[i][j]` armazena um valor na posição `i` e `j`.

```
#include <iostream>
using namespace std;

int main() {
    int a[2][2];

    a[0][0] = 7;
    a[0][1] = 1;
    a[1][0] = 2;
    a[1][1] = 5;

    for ( int i = 0; i < 2; i++){
        for ( int j = 0; j < 2; j++){
            cout << a[i][j] << " ";
            cout << endl;
        }
    }
    return 0;
}
```

A atribuição de valores a uma matriz pode ser feita na declaração, por exemplo:

```
int a[2][2] = { 7, 1, 2, 5};
int a[2][2] = { { 7, 1 } , { 2, 5 } };
```

Muitas manipulações de matrizes utilizam as instruções de repetição **for**, por exemplo:

- Atribuindo zero a todos os elementos da linha 2:

```
for ( int j = 0; j < 2; j++ )
    a[1][j] = 0;
```

- Realizando a soma de todos os elementos da matriz:

```
soma = 0;
for ( i = 0; i < 2; i++ )
    for ( j = 0; j < 2; j++ )
        soma += a[ i ][ j ];
```

## 10.2 Função com Matrizes

Quando matrizes são passadas como argumentos para funções, todas as dimensões, exceto a primeira (a primeira dimensão é opcional), devem ser fornecidas, como no exemplo a seguir.

```
int funcA ( int a[][2] );
```

Nas chamadas às funções, matrizes são passadas como parâmetros utilizando apenas o seu nome (sem colchetes), o número de linhas e o número de colunas.

```
funcA ( a, nl, nc );
```

Toda matriz passada para funções como parâmetro é passada por referência. Isto significa dizer que as alterações realizadas nas matrizes são visíveis fora do corpo da função;

**Exemplo 10.2.1.** Crie duas funções: uma função que **lê do usuário** os elementos de uma matriz de números inteiros com tamanho  $nl \times nc$  termos, e outra função para **imprimir na tela** os elementos da matriz de números inteiros com tamanho  $nl \times nc$  termos. Crie um programa (**main**) para testar as duas funções.

```
1  #include<iostream>
2  using namespace std ;
3
4  const int MAXDIM = 50;
5
6  void leMatriz ( int M [ ][ MAXDIM ], int nl, int nc )
7  {
8      for ( int i = 0; i < nl ; i++)
9          for ( int j = 0; j < nc ; j++)
10             cin >> M[ i ][ j ];
11 }
12
```

```

13 void printMatriz ( int M [ ][ MAXDIM ], int nl, int nc )
14 {
15     for ( int i = 0; i < nl ; i++){
16         for ( int j = 0; j < nc ; j++)
17             cout << M[ i ][ j ] << " " ;
18         cout << endl ;
19     }
20 }
21 int main ()
22 {
23     int M[ MAXDIM ][ MAXDIM ];
24     int nl, nc ;
25
26     cout << " Digite numero de linhas da matriz : " << endl ;
27     cin >> nl ;
28     cout << " Digite numero de colunas da matriz : " << endl ;
29     cin >> nc ;
30
31     leMatriz(M, nl, nc ) ;
32     printMatriz(M, nl, nc ) ;
33
34     return 0;
35 }

```

## 10.3 Exercícios de Fixação

1. Dada uma matriz de inteiros  $5 \times 5$ ,

$$A = \begin{bmatrix} 11 & 12 & 13 & 14 & 15 \\ 21 & 22 & 23 & 24 & 25 \\ 31 & 32 & 33 & 34 & 35 \\ 41 & 42 & 43 & 44 & 45 \\ 51 & 52 & 53 & 54 & 55 \end{bmatrix}$$

faça um programa que:

- (a) Escreva os elementos da diagonal principal;
  - (b) Escreva a soma dos elementos acima da diagonal principal;
  - (c) Escreva o produto dos elementos abaixo da diagonal principal;
  - (d) Escreva todos os elementos exceto os elementos da diagonal secundária.
2. Faça um programa que preencha uma matriz  $m \times n$  com números inteiros e some cada uma das linhas, armazenando o resultado das somas em um vetor. A seguir, o programa deverá multiplicar cada elemento da matriz pela soma da linha correspondente e mostrar a matriz resultante.
  3. Crie uma função que obtém o maior elemento de uma matriz de números inteiros com tamanho  $nl \times nc$  termos. Crie um programa (main) para testar as duas funções.

4. Considerando uma matriz esparsa quando pelo menos 70% dos seus elementos forem nulos, escreva uma função que recebe uma matriz como parâmetro e retorna verdadeiro caso ela seja esparsa e falso caso contrário. Escreva também um programa (main) para testar a função implementada.
5. Crie uma função que recebe duas matrizes A e B de mesmas dimensões como parâmetros de entrada e uma matriz R como parâmetro de saída. Na matriz R deve ser calculada a subtração matricial  $A - B$ . Matematicamente, a subtração matricial é definida para cada termo  $r_{ij}$  da matriz resultante como sendo:

$$r_{ij} = a_{ij} - b_{ij}$$

Crie também um programa main para testar a função implementada. Nela, devem ser definidas as matrizes de entrada pelo usuário, calcular a subtração e, ao final, imprimir a matriz resultante na tela.

6. Na teoria dos sistemas, define-se o elemento MINMAX de uma matriz como o maior elemento da linha em que se encontra o menor elemento da matriz. Crie uma função que receba uma matriz e o seu tamanho (número de linhas e colunas), calcule e mostre seu MINMAX e sua posição (linha, coluna).
7. Crie uma função que receba uma matriz e o seu tamanho (número de linhas e colunas) e **exibe** a matriz transposta da entrada. Dada uma matriz A, a matriz transposta  $A^t$  possui a seguinte lei de formação:

$$A_{ij}^t = A_{ji}$$

8. Faça um programa em C++ que recebe do usuário uma matriz de ponto flutuante A com dimensões  $nl \times nc$ . A seguir, o seu programa deve calcular e exibir as seguintes normas matriciais:

**OBS:** Crie uma função para o cálculo de cada uma das normas.

- (a) Norma infinita: A maior soma absoluta das linhas da matriz.

$$\|A\|_{\infty} = \max_{1 \leq i \leq nl} \left( \sum_{j=1}^{nc} |a_{ij}| \right)$$

- (b) Norma 2,1: Soma da norma euclidiana das colunas da matriz.

$$\|A\|_{2,1} = \sum_{j=1}^{nc} \left( \sum_{i=1}^{nl} a_{ij}^2 \right)^{1/2}$$

- (c) Norma de Frobenius: Representação da norma Euclidiana para um espaço  $nl \times nc$ .

$$\|A\|_F = \sqrt{\sum_{i=1}^{nl} \sum_{j=1}^{nc} a_{ij}^2}$$

**OBSERVAÇÃO:** A notação utilizada assume a indexação matemática (de 1 até  $nl$  para as linhas, de 1 até  $nc$  para as colunas). Lembre-se de adaptar para a situação de C++.

Exemplo de aplicação:

```
\\Exemplo 01:
Digite o no de linhas da matriz:
3
Digite o no de colunas da matriz:
4
Digite os termo da matriz:
1 2 -1 6
3 -4 0 2
8 0 -3 1
Norma infinita: 12
Norma 2,1: 22.639863
Norma Frobenius: 12.041595
```

## 10.4 Referências Bibliográficas

1. MANZANO, J.A.; OLIVEIRA, J.F.; **Algoritmos - Lógica para Desenvolvimento de Programação**. Editora Erica.
2. ASCENCIO, A.F.G.; CAMPOS, E.A.V. **Fundamentos da Programação de Computadores - Algoritmos, Pascal e C/C++**. 3ed. Editora Pearson.
3. DEITEL, H. M.; DEITEL, P. J. **C++ Como Programar**. 3ed. Editora Bookman.