

## Lista de Exercícios

### Matrizes Bidimensionais

**Disciplina:** Algoritmos e Programação II

**Semestre** 2014/2

**Professora:** Daniela Scherer dos Santos

**Letivo:**

**Data:** 20/08/14

1. Faça um programa para preencher uma matriz de 6 linhas e 5 colunas com números inteiros digitados pelo usuário. No final o programa deverá mostrar a matriz preenchida.

Exemplo de formatação da saída do programa:

Matrix(6x5)

3	5	16	4	12
8	6	7	9	23
9	13	23	24	45
17	12	87	45	12
1	9	8	3	111
22	33	44	18	12

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```
namespace ConsoleApplication3
```

```
{
    class Program
    {
        static void Main(string[] args)
        {
            const int L = 3, C = 2;
            int x, y;
            int[,] matriz = new int[L, C];

            //preenche COM DADOS DIGITADOS PELO USUÁRIO
            Console.WriteLine("Informe os {0} valores para a matriz: ", L * C);
            for (x = 0; x < L; x++)
            {
                for (y = 0; y < C; y++)
                {
                    matriz[x, y] = Convert.ToInt16(Console.ReadLine());
                }
            }

            //mostra matriz preenchida
            Console.WriteLine("Matriz preenchida");
            for (x = 0; x < L; x++)
            {
                for (y = 0; y < C; y++)
                {
                    Console.Write(matriz[x, y] + "\t");
                }
                Console.WriteLine();
            }
            Console.ReadKey();
        }
    }
}
```

2. Faça um programa para ler uma matriz 5x5 e em seguida ler também **dois** pares de valores X e Y quaisquer correspondentes a duas posições na matriz. Ao final seu programa deverá escrever a soma dos valores encontrados nas respectivas posições X e Y.

```
static void Main(string[] args)
```

```

{
    const int LINHAS = 2;
    const int COLUNAS = 2;
    int[,] matriz = new int[LINHAS, COLUNAS];
    int soma, linha1, linha2, coluna1, coluna2, l, c;

    //faz a leitura dos dados da matriz
    Console.WriteLine("Digite os " + LINHAS * COLUNAS + " valores da matriz:");
    for ( l = 0; l < LINHAS; l++)
    {
        for ( c = 0; c < COLUNAS; c++)
        {
            Console.SetCursorPosition(c * 3, l + 1); //posiciona o cursor na linha e coluna determinada
            matriz[l, c] = Convert.ToInt16(Console.ReadLine());
        }
    }
    Console.WriteLine("Informe a linha do primeiro valor: ");
    linha1 = int.Parse(Console.ReadLine());
    while (linha1 < 0 || linha1 >= LINHAS)
    {
        Console.WriteLine("Valor fora dos limites, digite novamente!");
        Console.Write("Linha: ");
        linha1 = int.Parse(Console.ReadLine());
    }

    Console.WriteLine("Informe a coluna do primeiro valor: ");
    coluna1 = int.Parse(Console.ReadLine());
    while (coluna1 < 0 || coluna1 >= COLUNAS)
    {
        Console.WriteLine("Valor fora dos limites, digite novamente!");
        Console.Write("Coluna: ");
        coluna1 = int.Parse(Console.ReadLine());
    }

    Console.WriteLine("Informe a linha do segundo valor: ");
    linha2 = int.Parse(Console.ReadLine());
    while (linha2 < 0 || linha2 >= LINHAS)
    {
        Console.WriteLine("Valor fora dos limites, digite novamente!");
        Console.Write("Linha: ");
        linha2 = int.Parse(Console.ReadLine());
    }

    Console.WriteLine("Informe a coluna do segundo valor: ");
    coluna2 = int.Parse(Console.ReadLine());
    while (coluna2 < 0 || coluna2 >= COLUNAS)
    {
        Console.WriteLine("Valor fora dos limites, digite novamente!");
        Console.Write("Coluna: ");
        coluna2 = int.Parse(Console.ReadLine());
    }

    soma = matriz[linha1, coluna1] + matriz[linha2, coluna2];
    Console.WriteLine("A soma dos valores nas posições solicitadas é: " + soma);
    Console.ReadKey();
}

```

3. Faça um programa para ler uma matriz 6X7 e no final imprimir a soma de todos os valores desta matriz.

```

static void Main(string[] args)
{
    const int LINHAS = 3;
    const int COLUNAS = 2;
    int[,] matriz = new int[LINHAS, COLUNAS];
    int soma, l, c;

```

```

//preenche a matriz com valores digitados pelo usuário
Console.WriteLine("Digite os " + LINHAS * COLUNAS + " da matriz:");
for (l = 0; l < LINHAS; l++)
{
    for (c = 0; c < COLUNAS; c++)
    {
        matriz[l, c] = int.Parse(Console.ReadLine());
    }
}

//soma todos os elementos da matriz
soma = 0;
for (l = 0; l < LINHAS; l++)
{
    for (c = 0; c < COLUNAS; c++)
    {
        soma += matriz[l, c];
    }
}
Console.Clear();
Console.WriteLine("A soma de todos os elementos da matriz é: " + soma);
Console.ReadKey();
}

```

4. Faça um programa para preencher uma matriz 5x5 com 1 na diagonal principal e com 0 os demais elementos. Escreva ao final a matriz obtida.

```

static void Main(string[] args)
{
    const int LINHAS = 5;
    const int COLUNAS = 5;
    int[,] matriz = new int[LINHAS, COLUNAS];
    int l, c;

    //GERA A MATRIZ
    for (l = 0; l < LINHAS; l++)
    {
        for (c = 0; c < COLUNAS; c++)
        {
            if (l == c)//testa se POSIÇÃO corresponde a diagonal principal
                matriz[l, c] = 1;
            else//se não estiver na diagonal principal
                matriz[l, c] = 0;
        }
    }

    //imprime matriz na tela
    for (l = 0; l < LINHAS; l++)
    {
        for (c = 0; c < COLUNAS; c++)
        {
            Console.SetCursorPosition(c * 3, l + 1);//posiciona o cursor na linha e coluna determinada
            Console.Write(matriz[l, c]);
        }
    }

    Console.ReadKey();
}

```

5. Faça um programa para ler uma matriz 20 x 20 e ler também um valor X. O programa deverá fazer uma busca desse valor na matriz e, sempre que o encontrar, escrever a localização (linha e coluna) ou uma mensagem de “não encontrado” no final da execução do programa (CASO O VALOR NÃO ESTEJA PRESENTE NA MATRIZ).

```

static void Main(string[] args)
{
    const int LINHAS = 3;
    const int COLUNAS = 3;

```

```

int[,] matriz = new int[LINHAS, COLUNAS];
int valor, l, c;
bool achou = false;

//leitura da matriz
Console.WriteLine("Digite os " + LINHAS * COLUNAS + " da matriz: ");
for (l = 0; l < LINHAS; l++)
{
    for (c = 0; c < COLUNAS; c++)
        matriz[l, c] = int.Parse(Console.ReadLine());
}

Console.Write("Digite o valor que deseja encontrar na matriz: ");
valor = int.Parse(Console.ReadLine());

for (l = 0; l < LINHAS; l++)
{
    for (c = 0; c < COLUNAS; c++)
    {
        if (valor == matriz[l, c])
        {
            Console.WriteLine("Valor: " + matriz[l, c] + " encontrado na linha: " + l + " coluna: " +
c);
            achou = true;
        }
    }
}
if (achou == false)
    Console.WriteLine("Valor não encontrado na matriz");
Console.ReadKey();
}

```

6. Faça um programa que leia uma matriz 4X5 de números inteiros e determine o maior elemento desta matriz e a sua posição (linha e coluna). Exemplo de formatação da saída do programa:

Matrix (4x5)

17	12	87	45	12
3	5	16	4	12
8	6	7	9	23
9	13	23	24	45

Maior elemento da matriz= 87

Posição do maior elemento da matriz: linha=0 coluna=2

```

static void Main(string[] args)
{
    const int LINHAS = 4;
    const int COLUNAS = 5;
    int[,] matriz = new int[LINHAS, COLUNAS];
    int maior, posLin, posCol, l, c;
    //ler matriz
    Console.WriteLine("Digite os " + LINHAS * COLUNAS + " da matriz: ");
    for (l = 0; l < LINHAS; l++)
    {
        for (c = 0; c < COLUNAS; c++)
        {
            Console.SetCursorPosition(c * 3, l + 1);
            matriz[l, c] = int.Parse(Console.ReadLine());
        }
    }
    //encontra maior elemento
    maior = matriz[0, 0];
    posLin = 0;
    posCol = 0;
    for (l = 0; l < LINHAS; l++)
    {
        for (c = 0; c < COLUNAS; c++)

```

```

        {
            if (matriz[l, c] > maior)
            {
                maior = matriz[l, c];
                posLin = l;
                posCol = c;
            }
        }
    }
    Console.WriteLine("O maior elemento da matriz é: " + maior + " e está na linha " + posLin + " e na coluna " + posCol);
    Console.ReadKey();
}

```

7. Faça um programa que leia uma matriz 8x 8 e escreva o maior elemento da diagonal principal e a soma dos elementos da diagonal secundária.

```

static void Main(string[] args)
{
    const int LINHAS = 4;
    const int COLUNAS = 4;
    int[,] matriz = new int[LINHAS, COLUNAS];
    int maior, soma, l, c;

    //preenche a matriz
    Console.WriteLine("Digite os " + LINHAS * COLUNAS + " da matriz: ");
    for (l = 0; l < LINHAS; l++)
    {
        for (c = 0; c < COLUNAS; c++)
        {
            Console.SetCursorPosition(c * 3, l + 1);
            matriz[l, c] = int.Parse(Console.ReadLine());
        }
    }

    //encontra o maior valor da diagonal principal
    maior = matriz[0, 0]; //maior recebe o primeiro valor da diagonal principal
    for (l = 1; l < LINHAS; l++)
    {
        for (c = 1; c < COLUNAS; c++)
        {
            if (l == c) //se a posição pertence a diagonal principal
            {
                if (matriz[l, c] > maior)
                    maior = matriz[l, c];
            }
        }
    }

    //soma os elementos da diagonal secundária
    soma = 0;
    c = COLUNAS - 1;
    for (l = 0; l < LINHAS; l++)
    {
        soma += matriz[l, c];
        c--;
    }
    Console.WriteLine("O maior valor da diagonal principal é: " + maior);
    Console.WriteLine("A soma dos valores da diagonal secundária é: " + soma);

    Console.ReadKey();
}

```

8. Faça um programa que leia uma matriz B, determine a linha de B que possui o maior resultado da soma de seus elementos. Exemplo de formatação da saída do programa:  
Matriz(4x4)

23	13	5	4
13	8	23	35
5	23	66	54
4	35	54	20

Linha que possui maior resultado da soma dos elementos= 2

9. Uma matriz quadrada inteira é chamada de "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 abaixo representa um quadrado mágico:

8	0	7
4	5	6
3	10	2

Escreva um programa que verifica se uma matriz de n linhas e n colunas representa um quadrado mágico. Obs.: o programa deve funcionar para qualquer valor de "n".