



Vectores e Matrizes

? EXERCÍCIO 1: LEITURA E ESCRITA DE VECTOR COM N ELEMENTOS

```
using System;
namespace VM1
{class LeituraEEscritaNElementos
{static void Main(string[] args)
{Console.Write("Quantos elementos tem o vector? ");
int N=Convert.ToInt16(Console.ReadLine());
int[] A=new int[N];
for (int I=0; I <= N - 1; I++)
{Console.Write("Elemento {0}=", I + 1);
A[I]=Convert.ToInt16(Console.ReadLine());}

foreach (int Elemento in A)
Console.WriteLine("{0, 4}", Elemento);
}}}
```

? EXERCÍCIO 2: LEITURA DE VECTOR COM VÁRIOS ELEMENTOS

```
using System;
namespace VM2
{class LeituraEEscritaVariosElementos
{static void Main(string[] args)
{int I=0, N;
int[] A=new int[100];
Console.Write("Digite um numero (999 para terminar) ");
int Numero=Convert.ToInt32(Console.ReadLine());
while (Numero != 999)
{A[I]=Numero;
I++;
Console.Write("Digite um numero (999 para terminar) ");
Numero=Convert.ToInt32(Console.ReadLine());}
N=I-1;
for (I=0; I<=N ; I++)
Console.Write("{0,3}", A[I]);
Console.WriteLine();
}}}
```

? EXERCÍCIO 3: VECTOR DE CARACTERES

```
using System;
namespace VM3
{class VectorDeLetras
{static void Main(string[] args)
{int N=16;
char Letra='A';
char[] Letras =new char[N];
for (int I=0; I <= N-1; I++)
{Letras[I]=Letra;
Letra++;}
foreach (char Elemento in Letras)
Console.Write("{0, 3}",Elemento);
Console.WriteLine();
}}}
```

? EXERCÍCIO 4: SOMATÓRIO DOS ELEMENTOS

```
using System;
namespace VM4
{class Somatorio
{static void Main(string[] args)
{Console.Write("Quantos elementos tem o vector? ");
int N= Convert.ToInt16(Console.ReadLine());
double [] A=new double[N];
for (int I=0; I <=N-1; I++)
{Console.Write("Elemento {0}=", I+1);
A[I]=Convert.ToDouble(Console.ReadLine());}
double Total=0;
for (int I=0; I<=N-1; I++)
Total+=A[I];
Console.WriteLine("Somatório dos elementos={0,4:F2}", Total);
}}}
```

? EXERCÍCIO 5: MÉDIA ARITMÉTICA

```
using System;
namespace VM5
{class MediaAritmetica
{static void Main(string[] args)
{Console.Write("Quantos elementos tem o vector? ");
int N= Convert.ToInt16(Console.ReadLine());
```

```
double [] A=new double[N];
for (int I=0; I <= N - 1; I++)
    {Console.Write("Elemento {0}=", I+1);
    A[I]=Convert.ToDouble(Console.ReadLine());}
double Med=0;
for (int I=0; I<=N-1; I++)
    Med+=A[I];
Med=Med / N;
Console.WriteLine("Média aritmética={0,4:F2}", Med);
}}}
```

? EXERCÍCIO 6: CONTAGEM DE ELEMENTOS

```
using System;
namespace VM6
{class ContaOcorrencias
    {static void Main(string[] args)
    {int [] A={4,2,5,4,3,5,2, 2,4};
    int N= A.Length-1;
    int [] Conta= new int[N+1];
    for (int I=0; I<=N-1; I++)
        Conta[I]=1;
    for (int I=0; I<N-1;I++)
        {if (Conta[I]!=0)
            {for (int J=I+1; J<=N; J++)
                {if (A[I]==A[J])
                    {Conta[I]++;
                    Conta[J]=0;}}}}
    for (int I=0; I<=N; I++)
        if (Conta[I]!=0)
            Console.WriteLine(" {0, 4} ocorre {1, 2} vez(es) em A",
                               A[I],Conta[I]);
    }}}}
```

? EXERCÍCIO 7: MAIOR E SEGUNDO MAIOR

```
using System;
namespace VM7
{class MaiorESegundoMaior
    {static void Main(string[] args)
    {Console.Write("Quantos elementos tem o vector? ");
    int N=Convert.ToInt16(Console.ReadLine());
    double[] A=new double[N];
    for (int I=0; I <= N - 1; I++)
        {Console.Write("Elemento {0}=", I );
```

```
A[I]=Convert.ToDouble(Console.ReadLine());}
int Indmax1=0 , Indmax2=0;
for (int I=1; I <= N - 1; I++)
{if (A[I] >= A[Indmax1])
    {Indmax2=Indmax1;
    Indmax1=I;}
else
    if (A[I] >= A[Indmax2])
        Indmax2=I;}
Console.WriteLine("O maior elemento {0} tem o índice {1}",
    A[Indmax1], Indmax1);
Console.WriteLine("O segundo maior elemento {0} tem o índice"+
    " {1} ", A[Indmax2], Indmax2);
}}}
```

? EXERCÍCIO 8: MAIOR E MENOR ELEMENTOS

```
using System;
namespace VM8
{class MaiorEMenor
{static void Main(string[] args)
{Console.Write("Quantos elementos tem o vector? ");
int N=Convert.ToInt16(Console.ReadLine());
double[] A=new double[N];
for (int I=0; I <= N - 1; I++)
{Console.Write("Elemento {0}=", I);
A[I]=Convert.ToDouble(Console.ReadLine());}
int Indmax=0, Indmin=0;
for (int I=1; I <= N - 1; I++)
{if (A[I] < A[Indmin])
    Indmin=I;
else if (A[I] > A[Indmax])
    Indmax=I;}
Console.WriteLine("Maior elemento: " + A[Indmax]);
Console.WriteLine("Menor elemento: "+ A[Indmin]);
}}}
```

? EXERCÍCIO 9: ADIÇÃO E PRODUTO DE DOIS VECTORES

```
using System;
namespace VM9
{class AdicaoEProduto
{static void Main(string[] args)
{Console.Write("Quantos elementos tem cada vector? ");
int N=Convert.ToInt16(Console.ReadLine());
```

```
double [] A= new double[N];
double [] B= new double[N];
double [] Total= new double[N];
double [] Prod= new double[N];
for (int I=0; I <= N - 1; I++)
    {Console.Write("Elemento a[{0}]=",I);
    A[I]=Convert.ToDouble(Console.ReadLine());}
for (int I=0; I <= N - 1; I++)
    {Console.Write("Elemento B[{0}]=", I);
    B[I]=Convert.ToDouble(Console.ReadLine());}
for (int I=0; I <= N - 1; I++)
    {Total[I]=A[I]+B[I];
    Prod[I]=A[I]*B[I];}
Console.WriteLine("Adição e produto dos elementos dos dois"+
    "vectores:");
for (int I=0; I<= N-1; I++)
    {Console.WriteLine("{0,3} + {1,3}={2,4}",A[I],B[I],Total[I]);
    Console.WriteLine("{0,3} * {1,3}={2,4}",A[I],B[I],Prod[I]);}
}}}
```

? EXERCÍCIO 10: RELATÓRIO DE VENDAS

```
using System;
namespace VM10
{class RelatorioVendas
{static void Main(string[] args)
{const int Mesano=12;
double[] Vendas = new double[12];
for (int M = 0; M <Mesano; M++)
    Vendas[M] = 0;
Console.Write("Mês da venda ou 0 para terminar ");
int Mes = Convert.ToInt16(Console.ReadLine());
Console.Write("Valor da venda ");
double Vvendas = Convert.ToDouble(Console.ReadLine());
while (Mes != 0)
    {Vendas[Mes-1] += Vvendas;
    Console.Write("Mês da venda ou 0 para terminar ");
    Mes = Convert.ToInt16(Console.ReadLine());
    Console.Write("Valor da venda ");
    Vvendas = Convert.ToDouble(Console.ReadLine());}
Console.WriteLine("{0,5} {1, 15}", "Mês", "Total de Vendas");
for (int M = 0; M < Mesano; M++)
    if (Vendas[M]!=0)
        Console.WriteLine("{0,4}{1, 10}", M + 1, Vendas[M]);
}}}
```

? EXERCÍCIO 11: PESQUISA LINEAR

```
using System;
namespace VM11
{
    class PesquisaLinear
    {
        static void Main(string[] args)
        {
            int X, N=10, I;
            int [] A= new int[N+1];
            for (I = 0; I <= N - 1; I++)
                A[I]=200 - 100 *I+ 150;
            Console.Write("Digite o valor a procurar ");
            X = Convert.ToInt16(Console.ReadLine());
            A[N]=X;
            I=0;
            while (A[I]!=X)
                I++;
            if (I!=N)
                Console.WriteLine("{0} é o elemento índice {1} do vector",
                                   X,I);
            else
                Console.WriteLine("{0} não existe no vector!!", X);
        }
    }
}
```

? EXERCÍCIO 12: PESQUISA BINÁRIA

```
using System;
namespace VM12
{
    class PesquisaBinaria
    {
        static void Main(string[] args)
        {
            double[] A={100,110,120,140,160,200,220};
            int N=7;
            Console.Write("Digite o valor a procurar ");
            double X = Convert.ToDouble(Console.ReadLine());
            int Inf=0, Sup=N-1, M=0;
            Boolean Exito=false;
            while (Exito==false && Inf<=Sup)
            {
                M=(Sup+Inf)/2;
                if (X>A[M])
                    Inf=M+1;
                else if (X<A[M])
                    Sup=M-1;
                else
                    Exito=true;
            }
            if (Exito==true)
                Console.WriteLine("O valor {0} está no array.", X);
            else
                Console.WriteLine("O valor {0} não está no array.", X);
        }
    }
}
```

```

Console.WriteLine ("{0} é o elemento índice {1} do vector",
                  X, M);
else
Console.WriteLine ("{0} não existe no vector!!", X);
}}}

```

? EXERCÍCIO 13: ORDENAÇÃO ALFABÉTICA

```

using System;
namespace VM13
{class OrdenacaoAlfabetica
{static void Main(string[] args)
{Console.Write("Quantos nomes tem a lista? ");
int K = Convert.ToInt16(Console.ReadLine());
string[] Nomes = new string[K];
string Temporaria;
int N = K - 1, Indicemenor;
for (int I=0; I<=N; I++)
{Console.Write("{0} nome? ", I+1);
Nomes[I] = Console.ReadLine();}
for (int Pas=0; Pas<=N-1; Pas++)
{Indicemenor=Pas;
for (int I=Pas+1; I<=N; I++)
if (Nomes[Indicemenor].CompareTo(Nomes[I])>=0)
Indicemenor=I;
if (Indicemenor!=Pas)
{Temporaria=Nomes[Pas];
Nomes[Pas]=Nomes[Indicemenor];
Nomes[Indicemenor]=Temporaria;}}
Console.WriteLine("\nLista ordenada:");
for (int I=0; I<=N; I++)
Console.WriteLine(Nomes[I]);
}}}}

```

? EXERCÍCIO 14: ORDENAÇÃO POR ORDEM DECRESCENTE

```

using System;
namespace VM14
{class OrdenacaoDecrescente
{static void Main(string[] args)
{Console.Write("Quantos números tem a lista? ");
int K = Convert.ToInt16(Console.ReadLine());
double[] A = new double[K];
double Temporaria;
int N = K - 1, Indicemaior;

```

```
for (int I = 0; I <= N; I++)
{
    Console.Write("{0} número? ", I + 1);
    A[I] = Convert.ToDouble(Console.ReadLine());
}
for (int Pas = 0; Pas <= N - 1; Pas++)
{
    Indicemaior = Pas;
    for (int I = Pas + 1; I <= N; I++)
        if (A[I] > A[Indicemaior])
            Indicemaior = I;
    if (Indicemaior != Pas)
    {
        Temporaria = A[Pas];
        A[Pas] = A[Indicemaior];
        A[Indicemaior] = Temporaria;
    }
}
Console.WriteLine("\nLista ordenada:");
for (int I = 0; I <= N; I++)
    Console.WriteLine(A[I]);
}}}
```

? EXERCÍCIO 15: INTERCALAÇÃO ORDENADA DE DOIS VECTORES

```
using System;
namespace VM15
{
    class IntercalacaoSimples
    {
        static void Main(string[] args)
        {
            int [] A={13,21,45,63};
            int N=A.Length;
            int[] B = { 1, 10, 25, 35, 55, 80, 90 };
            int M=B.Length;
            int[] C = new int[N+ M];
            int I, J, K;
            I = J = K = 0;
            while (I <= N - 1 && J <= M - 1)
            {
                if (A[I] <= B[J])
                {
                    C[K] = A[I];
                    I++;
                }
                else
                {
                    C[K] = B[J];
                    J++;
                }
                K++;
            }
            if (I>N-1)
            {
                for (int T=J; T<=M-1; T++)
                {
                    C[K] = B[T];
                    K++;
                }
            }
            else
            {
                for (int T=I; T<=N-1; T++)
                {
                    C[K] = A[T];
                    K++;
                }
            }
            for (K=0; K<=N+M-1; K++)
            {
            }
        }
    }
}
```



```
        Console.Write("{0, 3}", C[K]);
        Console.WriteLine();
    }
}
```

? EXERCÍCIO 16: LEITURA E ESCRITA DE UMA MATRIZ

```
using System;
namespace VM16
{
    class LeituraEscritaMatriz
    {
        static void Main(string[] args)
        {
            int N, M, Total;
            Console.Write("Quantas linhas tem a matriz? ");
            N=Convert.ToInt16(Console.ReadLine());
            Console.Write("Quantas colunas tem a matriz? ");
            M=Convert.ToInt16(Console.ReadLine());
            int [,] A=new int[N, M];
            for (int I=0; I<=N-1; I++)
                for (int J=0; J<=M-1; J++)
                {
                    Console.Write("Elemento a [{0},{1}]? ", I, J );
                    A[I,J]=Convert.ToInt16(Console.ReadLine());
                    Console.WriteLine();
                }
            for (int I=0; I<=N-1; I++)
                for (int J=0; J<=M-1; J++)
                {
                    Console.Write (" {0,4}",A[I,J]);
                    Console.WriteLine();
                }
        }
    }
}
```

? EXERCÍCIO 17: LEITURA E ESCRITA DE UMA MATRIZ COM VÁRIAS LINHAS

```
using System;
namespace VM17
{
    class LeituraEscritaMatriz2
    {
        static void Main(string[] args)
        {
            int N=0, M, Linha=0;
            Console.Write("Quantas colunas tem a matriz? ");
            M=Convert.ToInt16(Console.ReadLine());
            string [,] A=new string[100, M];
            Console.Write("Elemento a [{0},0] ou ZZZ? ", Linha);
            A[Linha,0]=Console.ReadLine();
            while (A[Linha, 0].ToUpper() != "ZZZ")
            {
                for (int Coluna = 1; Coluna <= M - 1; Coluna++)
                {
                    Console.Write("Elemento a [{0},{1}]? ", Linha, Coluna);
                    A[Linha,Coluna] = (Console.ReadLine());
                }
                Linha++;
                Console.Write("Elemento a [{0},0] ou ZZZ? ", Linha);
            }
        }
    }
}
```

```
A[Linha, 0] = Console.ReadLine();}
N = Linha;
Console.WriteLine();
for (Linha=0; Linha<=N-1; Linha++)
    {for (int Coluna=0; Coluna<=M-1; Coluna++)
        Console.Write (" {0,10}", A[Linha, Coluna].PadRight(10));
        Console.WriteLine();}
    }
}
```

? EXERCÍCIO 18: LEITURA E ESCRITA DE UMA MATRIZ TRIDIMENSIONAL

```
using System;
namespace VM18
{class LeituraMatriz3D
    {static void Main(string[] args)
        {Console.Write("Tamanho 1. dimensão ");
          int T = Convert.ToInt16(Console.ReadLine());
          Console.Write("Tamanho 2. dimensão ");
          int N = Convert.ToInt16(Console.ReadLine());
          Console.Write("Tamanho 3. dimensão ");
          int M = Convert.ToInt16(Console.ReadLine());
          string[, ,] A = new string[T,N, M];
          for (int P = 0; P <= T - 1; P++)
              for (int I = 0; I <= N - 1; I++)
                  for (int J = 0; J <= M - 1; J++)
                      {Console.Write("a[{0},{1},{2}]=", P, I, J);
                        A[P, I, J] = Console.ReadLine();}
          for (int P = 0; P <=T - 1; P++)
              {Console.WriteLine("Escola {0}", P);
                for (int I = 0; I <= N - 1; I++)
                    {for (int J = 0; J <= M - 1; J++)
                        Console.Write( A[P, I, J].PadRight(12));
                      Console.WriteLine();}}
```

? EXERCÍCIO 19: LEITURA E ESCRITA DE UM VECTOR DE VECTORES

```
using System;
namespace VM19
{class LeituraVectorDeVectores
    {static void Main(string[] args)
        {const int N = 3;
          String[,] Alunos = new string[N] [];
          Console.Write("Número de alunos inscritos na disciplina 1? ");
          int N1 = Convert.ToInt16(Console.ReadLine());
```

```

Console.WriteLine("Número de alunos inscritos na disciplina 2? ");
int N2 = Convert.ToInt16(Console.ReadLine());
Console.WriteLine("Número de alunos inscritos na disciplina 3? ");
int N3 = Convert.ToInt16(Console.ReadLine());
Alunos[0] = new string[N1];
Alunos[1] = new string[N2];
Alunos[2] = new string[N3];
for (int I = 0; I <=N-1; I++)
{
    Console.WriteLine("Disciplina {0}", I+1);
    for (int K = 0; K <= Alunos[I].GetLength(0) - 1; K++)
    {
        Console.WriteLine("Nome do {0} aluno ", K + 1);
        Alunos[I][K] = Console.ReadLine();
    }
    Console.WriteLine();
}
for (int I = 0; I <=N-1; I++)
{
    Console.WriteLine("Disciplina {0}", I+1);
    for (int K = 0; K <=Alunos[I].GetLength(0)-1; K++)
        Console.WriteLine(Alunos[I][K]);
}
}
}
}
}

```

? EXERCÍCIO 20: LEITURA E ESCRITA DE UM VECTOR DE MATRIZES

```

using System;
namespace VM20
{
    class LeituraVectorDeMatrizes
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Quantos produtos há? ");
            int N=Convert.ToInt16(Console.ReadLine());
            double [,] Produtos=new double [N][,];
            int [,] Nmat=new int[N,2];
            int I;
            for(I=0; I<=N-1; I++)
            {
                Console.WriteLine("Quantas vendas para o produto {0}? ",I+1);
                Nmat[I,0]=Convert.ToInt16(Console.ReadLine());
                Console.WriteLine("Quantos dados para cada venda do produto "+
                    "{0}? ", I + 1);
                Nmat[I,1]=Convert.ToInt16(Console.ReadLine());
            }
            I=0;
            for (int P=0; P<=N-1; P++)
            {
                Produtos[P]= new double [Nmat[I,0],Nmat[I,1]];
                I++;
            }
            for (int P = 0; P <= N - 1; P++)
            {
                Console.WriteLine("Produto {0}", P + 1);
                for (I = 0; I <= Produtos[P].GetLength(0) - 1; I++)
                {
                    for (int J = 0; J <= Produtos[P].GetLength(1) - 1; J++)
                    {
                        Console.WriteLine("Dados para a linha {0} coluna {1} = ",
                            I+1, J+1);
                        Produtos[P][I,J]=Convert.ToDouble(Console.ReadLine());
                    }
                }
            }
        }
    }
}

```

```
Console.WriteLine();
for (int P = 0; P <= N - 1; P++)
{
    Console.WriteLine("Produto {0}", P + 1);
    for (I = 0; I <= Produtos[P].GetLength(0) - 1; I++)
    {
        for (int J = 0; J <= Produtos[P].GetLength(1) - 1; J++)
        {
            Console.Write("{0, 6}", Produtos[P][I, J]);
            Console.WriteLine();
        }
    }
}
```

? EXERCÍCIO 21: SOMATÓRIO EM LINHA

```
using System;
namespace VM21
{
    class SomatorioEmLinha
    {
        static void Main(string[] args)
        {
            int N, M, Total;
            Console.Write("Quantas linhas tem a matriz? ");
            N = Convert.ToInt16(Console.ReadLine());
            Console.Write("Quantas colunas tem a matriz? ");
            M = Convert.ToInt16(Console.ReadLine());
            int [,] A = new int[N, M+1];
            for (int I = 0; I <= N-1; I++)
            {
                for (int J = 0; J <= M-1; J++)
                {
                    Console.Write("Elemento a [{0},{1}]? ", I, J);
                    A[I, J] = Convert.ToInt16(Console.ReadLine());
                }
            }
            for (int I = 0; I <= N-1; I++)
            {
                Total = 0;
                for (int J = 0; J <= M-1; J++)
                {
                    Total += A[I, J];
                }
                A[I, M] = Total;
            }
            for (int I = 0; I <= N-1; I++)
            {
                for (int J = 0; J <= M-1; J++)
                {
                    Console.Write("{0, 6}", A[I, J]);
                }
                Console.WriteLine("| {0, 6}", A[I, M]);
            }
        }
    }
}
```

? EXERCÍCIO 22: SOMATÓRIO EM COLUNA

```
using System;
namespace VM22
{
    class SomatorioEmColuna
    {
        static void Main(string[] args)
        {
            int N, M;
            double Total;
            Console.Write("Quantas linhas tem a matriz? ");
```

```

N=Convert.ToInt16(Console.ReadLine());
Console.Write("Quantas colunas tem a matriz? ");
M=Convert.ToInt16(Console.ReadLine());
double [,] A=new double[N+1, M];
for (int I=0; I<=N-1; I++)
    for (int J=0; J<=M-1; J++)
        {Console.Write("Elemento a[{0},{1}]? ", I, J );
          A[I,J]=Convert.ToDouble(Console.ReadLine());}
for (int J=0; J<=M-1; J++)
    {Total=0;
      for (int I=0; I<=N-1; I++)
          Total+=A[I,J];
      A[N,J]=Total;}
Console.WriteLine();
for (int I=0; I<=N-1; I++)
    {for (int J = 0; J <= M - 1; J++)
      Console.Write("{0, 6}", A[I, J]);
      Console.WriteLine();}
for (int J = 0; J <= M - 1; J++)
    Console.Write("{0, 6}", "-");
Console.WriteLine();
for (int J = 0; J <= M - 1; J++)
    Console.Write("{0, 6}", A[N, J]);
Console.WriteLine();}
}}}

```

? EXERCÍCIO 23: MAIOR ELEMENTO DA MATRIZ

```

using System;
namespace VM23
{class MaiorElementoMatriz
{static void Main(string[] args)
{Console.Write("Quantas linhas tem a matriz? ");
int N = Convert.ToInt16(Console.ReadLine());
Console.Write("Quantas colunas tem a matriz? ");
int M = Convert.ToInt16(Console.ReadLine());
double[,] A = new double[N, M];
for (int I = 0; I <= N - 1; I++)
    for (int J = 0; J <= M - 1; J++)
        {Console.Write("A[{0},{1}]=", I, J);
          A[I,J] = Convert.ToDouble(Console.ReadLine());}
int Linmax = 0;
int Colmax = 0;
for (int I = 0; I <= N - 1; I++)
    for (int J = 0; J <= M - 1; J++)
        if (A[I, J] > A[Linmax, Colmax])
            {Linmax = I;

```

```
        Colmax = J; }
    Console.WriteLine("Maior elemento:{0}", A[Linmax, Colmax]);
    Console.WriteLine("Linha {0}", Linmax);
    Console.WriteLine("Coluna {0}", Colmax);
    }}}
```

? EXERCÍCIO 24: MAIOR DE CADA LINHA

```
using System;
namespace VM24
{
    class MaiorDeCadaLinha
    {
        static void Main(string[] args)
        {
            Console.Write("Quantas linhas tem a matriz? ");
            int N = Convert.ToInt16(Console.ReadLine());
            Console.Write("Quantas colunas tem a matriz? ");
            int M = Convert.ToInt16(Console.ReadLine());
            double[,] A = new double[N, M + 1];
            int Colmax;
            for (int I = 0; I <= N - 1; I++)
                for (int J = 0; J <= M - 1; J++)
                    {
                        Console.Write("A[{0},{1}]= ", I, J);
                        A[I, J] = Convert.ToDouble(Console.ReadLine());
                    }
            for (int I = 0; I <= N - 1; I++)
            {
                Colmax = 0;
                for (int J = 1; J <= M - 1; J++)
                    if (A[I, J] > A[I, Colmax])
                        Colmax = J;
                A[I, M] = A[I, Colmax];
            }
            for (int I = 0; I <= N - 1; I++)
            {
                for (int J = 0; J <= M - 1; J++)
                    Console.Write("{0, 5}", A[I, J]);
                Console.WriteLine("|{0, 5}", A[I, M]);
            }
        }
    }
}
```

? EXERCÍCIO 25: MENOR DE CADA COLUNA

```
using System;
namespace VM25
{
    class MenorDeCadaColuna
    {
        static void Main(string[] args)
        {
            Console.Write("Quantas linhas tem a matriz? ");
            int N = Convert.ToInt16(Console.ReadLine());
            Console.Write("Quantas colunas tem a matriz? ");
            int M = Convert.ToInt16(Console.ReadLine());
            int[,] A = new int[N+1, M];
        }
    }
}
```

```

for (int I = 0; I <= N - 1; I++)
    for (int J = 0; J <= M - 1; J++)
        {Console.Write("A[{0},{1}]=", I, J);
         A[I, J] = Convert.ToInt16(Console.ReadLine());}
int Linmin;
for (int J = 0; J <= M - 1; J++)
    {Linmin = 0;
     for (int I = 1; I <= N - 1; I++)
         if (A[I, J] < A[Linmin, J])
             Linmin = I;
     A[N, J] = A[Linmin, J];}
for (int I=0; I<=N-1; I++)
    {for (int J = 0; J <= M - 1; J++)
        Console.Write("{0, 6}", A[I, J]);
      Console.WriteLine();}
for (int J = 0; J <= M - 1; J++)
    Console.Write("{0, 6}", "-");
Console.WriteLine();
for (int J = 0; J <= M - 1; J++)
    Console.Write("{0, 6}", A[N, J]);
Console.WriteLine();
}}}
```

? EXERCÍCIO 26: TESTE ESCOLHA MÚLTIPLA

```

using System;
namespace VM26
{class ClassificacaoFinal
{static void Main(string[] args)
{char[] Chave={'a', 'b', 'c', 'c', 'd', 'a'};
 int[] Cotacao={1,2, 3, 3, 2, 2};
 int M=Chave.Length;
 Console.Write("Quantos alunos há? ");
 int N=Convert.ToInt16(Console.ReadLine());
 char[,] Resp=new char[N,M];
 int[] Final=new int[N];
 for (int I=0; I<N; I++)
     for (int J=0; J<M; J++)
         {Console.Write("Resposta do aluno {0} à pergunta {1} ",
                          I+1, J+1);
          Resp[I,J]=Convert.ToChar(Console.ReadLine());}
 for (int I=0; I<N; I++)
     {Final[I]=0;
      for (int J=0; J<M; J++)
          if (Resp[I,J]==Chave[J])
              Final[I]+=Cotacao[J];}
 Console.WriteLine("\n {0,4} {1, 12}", "Aluno",
```

```
        "Classificação");
    for (int I=0; I<N; I++)
        Console.WriteLine("{0, 4} {1,6}", I+1, Final[I]);
    }}}
```

? EXERCÍCIO 27: VALORES DAS VENDAS

```
using System;
namespace VM27
{class TotalVendasVectorDeMatrizes
{static void Main(string[] args)
{int N=3;
double[,] Prod=new double[N][,];
Prod[0]=new double[,] {{1,1000,10}, {12,500,15}};
Prod[1]=new double[,] {{1,200}, {2, 300}, {3, 400},
{4,500}};
Prod[2]=new double [, ] {{10, 1000, 5},{11, 2000,10},
{12,3000,15}};
double Totvendas=0, Tot=0;
for (int P=0; P <= N - 1; P++)
{Tot=0;
Console.Write("Subtotal produto {0}=",P + 1);
for (int I=0; I<=Prod[P].GetLength(0) - 1; I++)
{Tot+=Prod[P][I, 1];
for (int J=2; J<=Prod[P].GetLength(1)-1; J++)
Tot-=Prod[P][I, J];}
Console.WriteLine(Tot);
Totvendas+=Tot;}
Console.WriteLine("Total das vendas={0} ",Totvendas);
}}}
```

? EXERCÍCIO 28: MATRIZ TRANSPOSTA

```
using System;
namespace VM28
{class MatrizTransposta
{static void Main(string[] args)
{double[,] A = { { 13, 21, 45, 63 }, { 5, 0, 10, 80 } };
int N = A.GetLength(0);
int M = A.GetLength(1);
double[,] At = new double[M, N];
int Ti = 0, Tj;
for (int J = 0; J <= M - 1; J++)
{Tj = 0;
for (int I = 0; I <= N - 1; I++)
```



```

        {At[Ti, Tj] = A[I, J];
          Tj++;}
        Ti++;}
for (Ti = 0; Ti <= M - 1; Ti++)
    {for (Tj = 0; Tj <= N - 1; Tj++)
        Console.Write("{0, 4}", At[Ti, Tj]);
      Console.WriteLine();
    }
}
}
}

```

? EXERCÍCIO 29: MATRIZ SIMÉTRICA

```

using System;
namespace VM29
{class MatrizSimetrica
    {static void Main(string[] args)
        {Console.Write("Quantas linhas tem a matriz? ");
          int N = Convert.ToInt16(Console.ReadLine());
          int[,] A = new int[N, N];
          for (int I = 0; I <= N - 1; I++)
              for (int J = 0; J <= I; J++)
                  {Console.Write("A[{0},{1}]= ", I, J);
                    A[I, J] = Convert.ToInt16(Console.ReadLine());
                    if (I!=J)
                        A[J,I]=A[I,J];}
          for (int I = 0; I <= N - 1; I++)
              {for (int J = 0; J <= N - 1; J++)
                  Console.Write("{0, 5}", A[I, J]);
                Console.WriteLine();
              }
        }
    }
}

```

? EXERCÍCIO 30: MATRIZ TRIANGULAR

```

using System;
namespace VM30
{class MatrizTriangular
    {static void Main(string[] args)
        {Console.Write("Triangular superior (S) ou inferior (I)? ");
          Char Tipo=Convert.ToChar(Console.ReadLine());
          Console.Write("Número de linhas da matriz? ");
          int N = Convert.ToInt16(Console.ReadLine());
          int[,] A = new int[N,N];
          if (Tipo=='i' || Tipo=='I')
              {for (int I = 0; I <= N - 1; I++)
                  for (int J = 0; J <= I; J++)
                      {Console.Write("a[{0},{1}] = ", I, J);

```

```
        A[I, J] = Convert.ToInt16( Console.ReadLine());
        if (I!=J)
            A[J, I] = 0;}}
    else
    {for (int I = 0; I <= N - 1; I++)
        for (int J = N-1; J >= I; J--)
            {Console.Write("a[{0},{1}]=", I, J);
            A[ I,J] = Convert.ToInt16(Console.ReadLine());
            if (I!=J)
                A[J,I] = 0;}}
        for (int I = 0; I <= N - 1; I++)
            {for (int J = 0; J <= N - 1; J++)
                Console.Write( "{0, 4}",A[I, J]);
            Console.WriteLine();}
    }}}
```

? EXERCÍCIO 31: ADIÇÃO DE MATRIZES

```
using System;
namespace VM31
{class AdicaoDeMatrizes
    {static void Main(string[] args)
        {int[,] A = { { 0, 1, 0 }, { 4, 0, 2 }, { 6, 3, 3}};
        int[,] B = { { 1, 0, 1 }, { -1, 2, -2 }, { 1, -3, 3 } };
        int N = A.GetLength(0);
        int M = A.GetLength(1);
        int[,] C = new int[N,M];
        for (int I = 0; I <= N-1 ; I++)
            for (int J = 0; J <= M-1; J++)
                C[I, J] = A[I, J] + B[I, J];
        for (int I = 0; I <= N-1; I++)
            {for (int J = 0; J <= M-1; J++)
                Console.Write("{0, 4}", C[I, J]);
            Console.WriteLine();}
        }
    }
}
```

? EXERCÍCIO 32: MULTIPLICAÇÃO DE MATRIZES

```
using System;
namespace VM32
{class MultiplicacaoDeMatrizes
    {static void Main(string[] args)
        {int[,] A = { { 1, 2, 3 }, { 4, 0, 2 } };
        int N1 = A.GetLength(0);
```

```
int M1 = A.GetLength(1);
int[,] B = { { 1, 0 }, { 2, 2 }, { 3, 3 } };
int N2 = B.GetLength(0);
int M2 = B.GetLength(1);
int[,] C = new int[N1,M2];
int Total = 0;
for (int I = 0; I <= N1-1 ; I++)
    {for (int K = 0; K <= M2-1; K++)
        {Total = 0;
         for (int J = 0; J <= M1-1; J++)
             Total = Total + A[I, J] * B[J, K];
         C[I, K] = Total;}}
for (int I = 0; I <= N1-1; I++)
    {for (int J = 0; J <= M2-1; J++)
        Console.Write("{0, 4}", C[I, J]);
        Console.WriteLine();
    }
}
```

? EXERCÍCIO 33: DETERMINANTE DA MATRIZ

```
using System;
namespace VM33
{class Determinante
{static void Main(string[] args)
{double[,] A = { { 1, 2, 3 }, { 3, 1, 4 }, { 0, 1, 2 } };
double[,] Alg = new double[A.GetLength(0), A.GetLength(1) + 2];
int K = 0;
double D=0, D1 = 0, D2 = 0, Prod;
for (int I = 0; I <= A.GetLength(0)-1; I++)
    {K = 0;
     for (int J = 0; J <= A.GetLength(1)+1; J++)
         {if (K==A.GetLength(1))
             K=0;
          Alg[I, J] = A[I, K];
          K++;}}
for (int Dg = 0; Dg <= Alg.GetLength(0)-1; Dg++)
    {K = Dg;
     Prod = 1;
     for (int I = 0; I <=Alg.GetLength(0)-1; I++)
         {Prod = Prod * Alg[I, K];
          K++;}
     D1 += Prod;}
for (int Dg = Alg.GetLength(1)-1; Dg>=Alg.GetLength(1)-3; Dg--)
    {K = Dg;
     Prod = 1;
     for (int I = 0; I <= Alg.GetLength(0)-1 ; I++)
```

```
        {Prod = Prod * Alg[I, K];  
        K--;}  
        D2 += Prod;}  
D = D1 - D2;  
Console.WriteLine("Determinante= {0, 2:F1}", D);  
}}}
```

? EXERCÍCIO 34: TRIÂNGULO DE PASCAL

```
using System;  
namespace VM34  
{class TrianguloPascal  
{static void Main(string[] args)  
{Console.Write("Altura do triângulo? ");  
int N = Convert.ToInt16(Console.ReadLine());  
long[,] T=new long [N*2+1,N*2+1];  
int Meio = (0 + 2*N) / 2;  
T[0, Meio] = 1;  
for (int I = 1; I <= Meio-1; I++)  
    for (int J=Meio-I; J<=Meio+I; J=J+2)  
        T[I, J] = T[I - 1, J - 1] + T[I - 1, J + 1];  
for (int I = 0; I <=Meio-1; I++)  
{Console.WriteLine();  
for (int J = 0; J <= N*2; J++)  
    if (T[I,J]!=0)  
        Console.Write("{0, 3}",T[I, J]);  
    else  
        Console.Write("{0, 3}", "");  
Console.WriteLine();}  
}}}
```

? EXERCÍCIO 35: QUADRADO MÁGICO

```
using System;  
namespace VM35  
{class QuadradoMagico  
{static void Main(string[] args)  
{Console.Write("Introduza a dimensão do quadrado: ");  
int N = Convert.ToInt16(Console.ReadLine());  
if (N % 2 == 0)  
    N++;  
int[,] Quad = new int[N, N];  
int Colcorr = (0 + N - 1) / 2, Colprox = 0;  
int Lincorr = 0, Linprox = 0;  
int Z = 1;
```

```

Quad[Lincorr, Colcorr] = Z;
for (Z = 2; Z <= N * N; Z++)
{
    Linprox = Lincorr - 1;
    if (Linprox < 0)
        Linprox = N - 1;
    Colprox = Colcorr - 1;
    if (Colprox < 0)
        Colprox = N - 1;
    if (Quad[Linprox, Colprox] != 0)
        Linprox++;
    Lincorr = Linprox;
    Colcorr = Colprox;
    Quad[Lincorr, Colcorr] = Z;}
int Total = 0;
for (Lincorr=0; Lincorr<=N- 1; Lincorr++)
{
    for (Colcorr=0; Colcorr<=N - 1; Colcorr++)
    {
        if (Lincorr==Colcorr)
            Total += Quad[Lincorr, Colcorr];
        Console.Write("{0,4}", Quad[Lincorr, Colcorr]);
        Console.WriteLine("");
    }
    Console.WriteLine("Total das linhas, colunas e "+
        "diagonais={0}", Total);
}
}
}
}

```