

TREINANDO MATRIZES EM LINGUAGEM C#

Desafio proposto 1

- Programa que lê número inteiro N e uma matriz quadrada de ordem N contendo números inteiros. Em seguida, mostrar a diagonal principal e a quantidade de valores negativos da matriz.

```
•
• using System;
•
• namespace Ex1Res
• {
•     class Program
•     {
•         static void Main(string[] args)
•         {
•             int N; //declaracao da variavel do tipo inteiro N
•             int[,] A; //aplicando Matriz
•
•             N = int.Parse(Console.ReadLine()); //Leitura da quantidade
de linhas e colunas
•             A = new int[N, N]; //aplicação da quantidade de linhas e
colunas na MATriz
•
•             //Leitura dos dados sequenciados na mesma linha atribuindo
nas colunas, declarado como i=Linha e j=Coluna
•             for (int i=0; i<N; i++){
•                 string[] s = Console.ReadLine().Split(' ');
//leitura na mesma linha, separando o valor de cada célula por um espaço
•                 for (int j=0; j<N; j++){
•                     A[i, j] = int.Parse(s[j]);
•                 }
•             }
•             // para ler na diagonal atribui-se a variavel "i" para
captar os valores na diagonal da matriz seguindo: (0x0;1x1;2x2) = [i, i]
concatenado com espaço
•             Console.WriteLine("Diagonal Principal: ");
•             for (int i=0; i<N; i++){
•                 Console.Write(A[i, i] + " ");
•             }
•             Console.WriteLine();
•             //para leitura de numeros negativos, utilizou-se dois "for"
para percorrer todas as células da matriz e através da variavel declarada
"cont" (contador), iniciando a contagem "se" houver algum numero negativo.
•             int cont = 0;
•             for (int i=0; i<N; i++){
•                 for (int j=0; j<N; j++){
•                     if (A[i, j] < 0){
•                         cont++;
•                     }
•                 }
•             }
•             Console.WriteLine("Quantidade de Negativos = " + cont);
•
•             Console.ReadKey();
•         }
•     }
• }
```