Trabalho vetores e matrizes

**Matrizes**

**Exercício 3**

class HelloWorld {

   public static void main(String[] args) {

       int[][] array = new int[4][4];

           for (int i=0;i<4;i++){

               for (int j=0;j<4;j++){

                   array[i][j]=i\*j;

               }

           }

           for (int x=0;x<4;x++){

               for (int y=0;y<4;y++){

                   System.out.print("\t"+array[x][y]);

               }

           }

   }

}

**Exercício 4**

#include

#include

int main() {

int MAT[4][4], linha, coluna, maior=0, posicao\_linha, posicao\_coluna;

for(linha=0; linha<4; linha++)

{

for(coluna=0; coluna<4; coluna++)

{

printf("Digite o valor de MAT[%d][%d]: ", linha, coluna);

scanf("%d", &MAT[linha][coluna]);

if(maior < MAT[linha][coluna])

{

maior = MAT[linha][coluna];

posicao\_linha = linha;

posicao\_coluna = coluna;

}

}

printf("\n");

}

for(linha=0; linha<4; linha++)

{

for(coluna=0; coluna<4; coluna++)

{

printf("[%d]", MAT[linha][coluna]);

}

printf("\n");

}

printf("O maior valor digitado foi %d, na linha %d, coluna %d ", maior, posicao\_linha, posicao\_coluna);

return 0;

}

**Exercício 5**

#include <stdio.h>

int main(){

int matriz[5][5];

int valorX;

int encontrado = 0; // Verifica se o Valor foi encontrado

int linhaEncontrado;

int colunaEncontrado;

int contador = 0; //Conta os Numeros Digitados

char ordinal = 176; // Tabela ASCII

// Leitura da Matriz

for (int i = 0; i < 5; i ++){

for (int j = 0; j < 5; j ++){

printf("Digite o %d%c Valor da Matriz 5x5: ",contador + 1, ordinal);

fflush(stdout);

scanf("%d", &matriz[i][j]);

contador ++;

}

}

// Leitura do Valor X

printf("\nDigite o Valor a ser buscado na Matriz: ");

fflush(stdout);

scanf("%d", &valorX);

// Busca do Valor X na Matriz

for (int i = 0; i < 5; i ++){

for (int j = 0; j < 5; j ++){

if (matriz[i][j] == valorX){

encontrado = 1;

linhaEncontrado = i;

colunaEncontrado = j;

break; //Valor encontrado, encerra a busca

}

}

if (encontrado)

break; //Valor encontrado, encerra a busca

}

//Exibindo a Matriz

printf("\nMatriz Resultante:\n");

for (int i = 0; i < 5; i ++){

for (int j = 0; j < 5; j ++){

printf("%d ", matriz[i][j]);

}

printf("\n");

}

// Exibindo o Resultado

if (encontrado){

printf("\nO Valor %d foi encontrado na linha %d e coluna %d.", valorX, linhaEncontrado + 1, colunaEncontrado + 1);

} else {

printf("\nO Valor %d não foi encontrado na Matriz!!!", valorX);

}

return 0;

}

**Exercício 6**

#include <stdio.h>

int main(){

int matriz1[4][4];

int matriz2[4][4];

int matrizResultante[4][4];

int contador1 = 0;

int contador2 = 0;

char ordinal = 176; // Tabela ASCII

// Lendo Elementos da Matriz 1

printf(".:: Matriz 1 ::.\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("Digite o %d%c Elemento da Matriz 1: ", contador1 + 1, ordinal);

fflush(stdout);

scanf("%d", &matriz1[i][j]);

contador1 ++;

}

}

// Exibindo a Matriz 1

printf("\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("%d ", matriz1[i][j]);

}

printf("\n");

}

// Lendo Elementos da Matriz 2

printf("\n\n.:: Matriz 2 ::.\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("Digite o %d%c Elemento da Matriz 2: ", contador2 + 1, ordinal);

fflush(stdout);

scanf("%d", &matriz2[i][j]);

contador2 ++;

}

}

// Exibindo a Matriz 2

printf("\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("%d ", matriz2[i][j]);

}

printf("\n");

}

// Exibindo da Matriz Resultante

printf("\n.:: Matriz 3 ::.\n");

printf("A Matriz Resultante dos Maiores Elementos e:\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

if (matriz1[i][j] > matriz2[i][j]){

matrizResultante[i][j] = matriz1[i][j];

} else {

matrizResultante[i][j] = matriz2[i][j];

}

printf("%d ", matrizResultante[i][j]);

}

printf("\n");

}

return 0;

}

**Exercício 7**

#include <stdio.h>

int main(){

int matriz[10][10];

// Preenchendo a Matriz

for (int i = 0; i < 10; i ++){

for (int j = 0; j < 10; j ++){

if (i < j){

matriz[i][j] = 2 \* i + 7 \* j - 2;

} else {

if (i == j){

matriz[i][j] = 3 \* i \* i - 1;

} else {

matriz[i][j] = 4 \* i \* i \* i - 5 \* j \* j + 1;

}

}

}

}

// Imprimindo a Matriz

printf(".:: MATRIZ ::.\n\n");

for (int i = 0; i < 10; i ++){

for (int j = 0; j < 10; j ++){

printf("%d ", matriz[i][j]);

}

printf("\n");

}

return 0;

}

**Exercício 8**

#include <iostream>

#include <locale.h>

#include <math.h>

#include <stdio.h>

using namespace std;

int main(){

setlocale (LC\_ALL, "portuguese");

int matriz[3][3], i, j, soma = 0;

for (i=0;i<3;i++){

for (j=0;j<3;j++){

cout<<"Digite o numero da matriz["<<i<<"]["<<j<<"]:";

cin>>matriz[i][j];

}

system ("cls");

}

soma = matriz[0][0] + matriz[1][1] + matriz[2][2];

cout<<"A soma da diagonal principa da matriz é:"<<soma;

return 0;

}

**Vetores**

**Exercício 11**

tapete = []

soma = 0

soma2 = 0

para i no intervalo ( 0 , 3 ):

tapete . acrescentar ( 0 )

tapete [ eu ] = []

para j no intervalo ( 0 , 3 ):

tapete [ eu ]. anexar ( entrada ( 'Número: ' ))

para i no intervalo ( 0 , 3 ):

para j no intervalo ( 0 , 3 ):

se eu == j :

soma = soma + mat [ i ][ i ]

soma2 = soma2 + mat [ i ][ 3 - i - 1 ]

para linha no tapete :

imprimir tupla ( linha )

print 'Soma da diagonal principal:' , soma

print 'Soma da diagonal secundária:' , soma2

**Exercício 12**

#include <stdio.h>

int main(){

int valores[5];

int maior;

int menor;

int soma = 0;

float media;

printf("Digite 5 Valores:\n");

fflush(stdout);

for (int i = 0; i < 5; i ++){

scanf("%d", &valores[i]);

soma += valores[i];

if (i == 0){

maior = menor = valores[i];

}else{

if (valores[i] > maior){

maior = valores[i];

}

if (valores[i] < menor){

menor = valores[i];

}

}

}

media = (float)soma / 5;

printf("\nValores Lidos: ");

for (int i = 0; i < 5; i ++){

printf("%d ", valores[i]);

}

printf("\n");

printf("Maior Valor: %d\n", maior);

printf("Menor Valor: %d\n", menor);

printf("Media dos Valores: %.2f", media);

return 0;

}

**Exercício 13**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main(){

int matriz[4][4];

// Gerando valores aleatorios no intervalo [1, 20]

srand(time(0));

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

matriz[i][j] = rand() % 20 + 1;

}

}

// Imprimindo a matriz original

printf(".:: MATRIZ ORIGINAL GERADA ::.\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("%4d ", matriz[i][j]);

}

printf("\n");

}

// Transformando a matriz em uma matriz triangular inferior

for (int i = 0; i < 4; i ++){

for (int j = i + 1; j < 4; j ++){

matriz[i][j] = 0;

}

}

// Imprimindo a Matriz Transformada

printf("\n.:: MATRIZ TRIANGULAR INFERIOR ::.\n");

for (int i = 0; i < 4; i ++){

for (int j = 0; j < 4; j ++){

printf("%4d ", matriz[i][j]);

}

printf("\n");

}

return 0;

}

**Exercício 14**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int a = 0, i, j, w, flag;

int vetor[10], iguais[10];

for (i = 0; i < 10; i++)

{

printf("Entre com o elemento[ %2d ] : ", i + 1);

scanf("%d", &vetor[i]);

}

for (i = 0; i < 10; i++)

{

for (j = 0; j < 10; j++)

{

flag = 0;

if (vetor[i] == vetor[j] && i != j)

{

for (w = 0; w < 10; w++)

{

if (iguais[w] == vetor[j])/\* se o numero já estiver nesse vetor de iguais \*/

flag = 1;

}

if (flag == 0)

{

iguais[a] = vetor[i];/\* para escrever o numero apenas uma vez \*/

a++;

}

}

}

}

printf("\n\nSaida --> ");

for (i = 0; i < a; i++)

{

printf("%d ", iguais[i]);

}

printf("\n\n");

system("pause");

printf("\n\n\n");

return 0;

}

**Exercício 15**

#include <stdio.h>

#define TAMANHO 5

int main() {

float vetor[TAMANHO];

int codigo;

char ordinal = 176; //Tabela ASCII

for (int i = 0; i < TAMANHO; i++) {

printf("Digite o %d%c Valor do Vetor de 5 posições:\n", i + 1, ordinal);

fflush(stdout);

scanf("%f", &vetor[i]);

}

printf("\nDigite o código (0, 1 ou 2):\n");

fflush(stdout);

scanf("%d", &codigo);

switch (codigo){

case 0:

printf("\nPrograma Finalizado!\n");

break;

case 1:

printf("\nVetor na Ordem Direta:\n");

for (int i = 0; i < TAMANHO; i ++){

printf("%.2f ", vetor[i]);

}

printf("\n");

break;

case 2:

printf("\nVetor na Ordem Inversa:\n");

for (int i = TAMANHO - 1; i >= 0; i --){

printf("%.2f ", vetor[i]);

}

printf("\n");

break;

default:

printf("\nCodigo Invalido!\n");

break;

}

return 0;

}

**Exercício 16**

#include <stdio.h>

#define TAMANHO 20

int main(){

int vetor[TAMANHO];

int vetorUnico[TAMANHO];

int tamanhoVetorUnico = 0;

char ordinal = 176; //Tabela ASCII

for (int i = 0; i < TAMANHO; i ++){

printf("Digite o %d%c Valor do Vetor de 20 Posicoes: ", i + 1, ordinal);

fflush(stdout);

scanf("%d", &vetor[i]);

}

//Verifica e adiciona elementos unicos ao vetorUnico

for (int i = 0; i < TAMANHO; i ++){

int elementoRepetido = 0;

for (int j = 0; j < tamanhoVetorUnico; j ++){

if (vetor[i] == vetorUnico[j]){

elementoRepetido = 1;

break;

}

}

if(!elementoRepetido){

vetorUnico[tamanhoVetorUnico] = vetor[i];

tamanhoVetorUnico ++;

}

}

printf("\nElementos do Vetor sem Repeticao:\n");

for (int i = 0; i < tamanhoVetorUnico; i ++){

printf("%d ", vetorUnico[i]);

}

printf("\n");

return 0;

}