**Nome: Bruno Deluca Satil Cassiano**

**Exercício 1**

* **Dado uma Matriz de ordem 3x3, faça com que o usuário insira todos os números inteiros. Depois da matriz armazenada, o programa deve calcular a somatória da diagonal principal bem como a média de todos os números da matriz, no final exibir os resultados.**

**#include <iostream>**

**using namespace std;**

**int matriz[3][3];**

**int countV = 0;**

**int countM = 0;**

**double media = 0;**

**int total = 0;**

**int somaDiagonal = 0;**

**int count = 0;**

**int main()**

**{**

**for(countV = 0; countV < 3; countV++){**

**for(countM = 0; countM < 3; countM++){**

**cout<<"Insira o " <<countM + 1 <<"º número da " <<countV + 1 <<"º linha da matriz: ";**

**cin>>matriz[countV][countM];**

**}**

**}**

**for(countV = 0; countV < 3; countV++){**

**for(countM = 0; countM < 3; countM++){**

**count++;**

**total += matriz[countV][countM];**

**media = total / count;**

**if (countM == countV) {**

**somaDiagonal += matriz[countV][countM];**

**}**

**}**

**}**

**cout<<"A somatória da diagonal principal resulta em " <<somaDiagonal <<" e a média da matriz fica em " <<media;**

**return 0;**

**}**

**Exercício 2**

* **Monte uma matriz 4x4, e nela o programa deve pedir para que o usuário insira números que irão povoar essa matriz (se o usuário inserir um número que já exista na matriz, o programa deve exibir uma mensagem de erro pedindo para que o usuário insira um novo número)**
* **Feito a matriz com todos os números diferentes, imprima o valor dessa matriz e logo em seguida, monte uma nova matriz 4x4, onde deverão ser armazenados nessa nova matriz, os valores da primeira matriz -1.**

**#include <iostream>**

**using namespace std;**

**int matriz[4][4];**

**int matrizSubtraida[4][4];**

**int numeroInserido;**

**int countV = 0;**

**int countM = 0;**

**bool verificador = false;**

**int main()**

**{**

**for (countV = 0; countV < 4; countV++){**

**for (countM = 0; countM < 4;){**

**cout<<"Insira o " <<countM + 1 <<"º número da " <<countV + 1 <<"º linha: ";**

**cin>>numeroInserido;**

**for (int countVverificador = 0; countVverificador < 4; countVverificador++){**

**for (int countMverificador = 0; countMverificador < 4;){**

**if (matriz[countVverificador][countMverificador] == numeroInserido){**

**cout<<"Esse número já foi inserido, insira um diferente." <<endl;**

**verificador = false;**

**break;**

**} else {**

**verificador = true;**

**countMverificador++;**

**}**

**}**

**if (verificador == false) {**

**break;**

**}**

**}**

**if (verificador == true) {**

**matriz[countV][countM] = numeroInserido;**

**countM++;**

**}**

**}**

**cout<<endl <<endl;**

**}**

**cout<<"Essa foi a matriz criada: " <<endl;**

**for (countV = 0; countV < 4; countV++){**

**cout<<"A " <<countV + 1 <<"º linha da matriz é: ";**

**for (countM = 0; countM < 4; countM++){**

**cout<<"[" <<matriz[countV][countM] <<"] ";**

**}**

**cout<<endl;**

**}**

**cout<<endl;**

**cout<<"-------------------------------------------------------------------";**

**cout<<endl;**

**cout<<"Agora, a matriz com 1 subtraído em todos os seus valores: " <<endl;**

**for (countV = 0; countV < 4; countV++){**

**for (countM = 0; countM < 4; countM++) {**

**matrizSubtraida[countV][countM] = matriz[countV][countM] - 1;**

**}**

**}**

**for (countV = 0; countV < 4; countV++){**

**cout<<"A " <<countV + 1 <<"º linha da matriz subtraída é: ";**

**for (countM = 0; countM < 4; countM++){**

**cout<<"[" <<matrizSubtraida[countV][countM] <<"] ";**

**}**

**cout<<endl;**

**}**

**return 0;**

**}**

**Exercício 3**

* **Dado uma matriz de ordem 10x10, preencha a matriz apenas com números pares (a escolha do programador) e imprima na tela esses valores de forma que fiquem ordenados (alinhamento vertical e horizontal). Logo após isso, o programa deve exibir:**
* **A soma de todos os números da 10 coluna**
* **A média dos números da terceira linha**
* **A multiplicação do número da esquerda superior pelo número da direita inferior.**
* **A diminuição do número da direita superior pelo número da esquerda inferior.**

**#include <iostream>**

**using namespace std;**

**int matriz[10][10];**

**int countV = 0;**

**int countM = 0;**

**int count = 0;**

**int somaDecimaColuna = 0;**

**int total = 0;**

**double mediaTerceiraLinha = 0;**

**int multiplicacao = 0;**

**int subtracao = 0;**

**int main()**

**{**

**for (countV = 0; countV < 10; countV++) {**

**for (countM = 0; countM < 10;){**

**if ((count % 2) == 0){**

**matriz[countV][countM] = count;**

**countM++;**

**}**

**count++;**

**}**

**}**

**for (countV = 0; countV < 10; countV++) {**

**cout<<countV + 1 <<"ª linha - ";**

**for (countM = 0; countM < 10; countM++) {**

**cout<<"[" <<matriz[countV][countM] <<"] ";**

**}**

**cout<<endl <<endl;**

**}**

**for (countV = 0; countV < 10; countV++) {**

**for (countM = 0; countM < 10; countM++) {**

**if (countM == 9) {**

**somaDecimaColuna += matriz[countV][countM];**

**}**

**if (countV == 2) {**

**total += matriz[countV][countM];**

**mediaTerceiraLinha = total / 9;**

**}**

**}**

**}**

**multiplicacao = matriz[0][0] \* matriz[9][9];**

**subtracao = matriz[0][9] - matriz[9][0];**

**cout<<"A soma de todos os números da décima coluna será " <<somaDecimaColuna <<endl <<endl;**

**cout<<"A média dos números da terceira linha é " <<mediaTerceiraLinha <<endl <<endl;**

**cout<<"A multiplacação do número da esquerda superior pelo número da direita inferior será " <<multiplicacao <<endl <<endl;**

**cout<<"A subtração do número da direita superior pelo número da esquerda inferior é " <<subtracao <<endl <<endl;**

**return 0;**

**}**

**Exercício 4**

* **Faça um algoritmo que construa uma matriz 30 por 30 de números inteiros e insira nela todos os números impares possíveis e imprima ordenadamente essa matriz (alinhamento vertical e horizontal).**

**Depois de construída, colocar o conteúdo de sua diagonal principal dentro de um vetor e depois do vetor montado, imprimir esse vetor.**

**#include <iostream>**

**using namespace std;**

**int matriz[30][30];**

**int vetor[30];**

**int countL = 0;**

**int countC = 0;**

**int countV = 0;**

**int main()**

**{**

**cout<<"Esta é a matriz de números impares: " <<endl <<endl;**

**for (countL = 0; countL < 30; countL++) {**

**for (countC = 0; countC < 30;) {**

**if ((countV % 2) == 1) {**

**matriz[countL][countC] = countV;**

**countC++;**

**}**

**countV++;**

**}**

**}**

**for (countL = 0; countL < 30; countL++) {**

**cout<<countL + 1 <<"ª linha - ";**

**for (countC = 0; countC < 30; countC++) {**

**cout<<"[" <<matriz[countL][countC] <<"] ";**

**}**

**cout<<endl <<endl;**

**}**

**cout<<"-----------------------------------------------------------------------------------------" <<endl <<endl;**

**cout<<"Este é o vetor montado com os números da diagonal principal da matriz de números impares: " <<endl <<endl;**

**countV = 0;**

**for (countL = 0; countL < 30; countL++) {**

**for (countC = 0; countC < 30; countC++) {**

**if (countL == countC) {**

**vetor[countV] = matriz[countL][countC];**

**countV++;**

**}**

**}**

**}**

**for (countV = 0; countV < 30; countV++) {**

**cout<<"[" <<vetor[countV] <<"] ";**

**}**

**return 0;**

**}**

**Exercício 5**

* **Crie uma matriz de ordem 10x10 e nela insira todos os números pares possíveis a partir de 0 (até encher a matriz), depois crie outra matriz 10x10 e nela insira todos os números impares possíveis a partir de 0.**
* **No final crie uma terceira matriz que deverá conter a soma dessas 2 matrizes (pares e impares) imprima as 3 matrizes.**

**#include <iostream>**

**using namespace std;**

**int matrizPar[10][10];**

**int matrizImpar[10][10];**

**int matrizResultante[10][10];**

**int countL = 0;**

**int countC = 0;**

**int count = 0;**

**int main()**

**{**

**for (countL = 0; countL < 10; countL++) {**

**for (countC = 0; countC < 10;) {**

**if ((count % 2) == 0) {**

**matrizPar[countL][countC] = count;**

**countC++;**

**}**

**count++;**

**}**

**}**

**count = 0;**

**for (countL = 0; countL < 10; countL++) {**

**for (countC = 0; countC < 10;) {**

**if ((count % 2) == 1) {**

**matrizImpar[countL][countC] = count;**

**countC++;**

**}**

**count++;**

**}**

**}**

**for (countL = 0; countL < 10; countL++) {**

**for (countC = 0; countC < 10; countC++) {**

**matrizResultante[countL][countC] = matrizPar[countL][countC] + matrizImpar[countL][countC];**

**}**

**}**

**cout<<"Matriz de números par:" <<endl <<endl;**

**for (countL = 0; countL < 10; countL++) {**

**cout<<countL + 1 <<"ª linha - ";**

**for (countC = 0; countC < 10; countC++) {**

**cout<<"[" <<matrizPar[countL][countC] <<"] ";**

**}**

**cout<<endl <<endl;**

**}**

**cout<<"----------------------------------------------------------------------"<<endl <<endl;**

**cout<<"Matriz de números impar:" <<endl <<endl;**

**for (countL = 0; countL < 10; countL++) {**

**cout<<countL + 1 <<"ª linha - ";**

**for (countC = 0; countC < 10; countC++) {**

**cout<<"[" <<matrizImpar[countL][countC] <<"] ";**

**}**

**cout<<endl <<endl;**

**}**

**cout<<"----------------------------------------------------------------------"<<endl <<endl;**

**cout<<"Matriz resultante da soma das matrizes de números pares e impares: " <<endl <<endl;**

**for (countL = 0; countL < 10; countL++) {**

**cout<<countL + 1 <<"ª linha - ";**

**for (countC = 0; countC < 10; countC++) {**

**cout<<"[" <<matrizResultante[countL][countC] <<"]";**

**}**

**cout<<endl <<endl;**

**}**

**return 0;**

**}**

**Exercício 6 (difícil)**

**Faça um jogo da velha para 2 jogadores em uma matriz 3x3. O jogador 1 deve escolher em qual lugar na matriz quer inserir seu número (1 nesse caso) Logo após é a vez do jogador numero 2 escolher o lugar onde quer colocar o seu número, no caso 2.**

**O jogo termina se o jogador numero 1 ou o jogador numero 2 fizerem 3 números em sequencia (despreze jogos em diagonal, considere apenas sequencias na vertical ou na horizontal). O jogo também deve terminar se todos os 9 espaços forem preenchidos.**

**OBS: Alguns editores de texto modificam a quantidade de underlines seguidas, então caso o programa pareça estar bugado, nessa parte dele “cout<<"\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" <<endl;” coloque 5 underlines entre cada pipe (barra vertical).**

**#include <iostream>**

**#include <string>**

**#include <string.h>**

**using namespace std;**

**string matriz[3][3];**

**int countL = 0;**

**int countC = 0;**

**int linha = 0;**

**int coluna = 0;**

**string simbolo = "";**

**int jogador = 1;**

**int marcadoX = 0;**

**int marcadoO = 0;**

**bool encerrou = false;**

**bool conferencia = false;**

**int main()**

**{**

**for (countL = 0; countL < 3; countL++) {**

**for (countC = 0; countC < 3; countC++){**

**matriz[countL][countC] = " ";**

**}**

**}**

**cout<<" | | "<<endl;**

**cout<<" "<<matriz[0][0]<<" | "<<matriz[0][1]<<" | "<<matriz[0][2]<<" "<<endl;**

**cout<<"\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" <<endl;**

**cout<<" | | "<<endl;**

**cout<<" "<<matriz[1][0]<<" | "<<matriz[1][1]<<" | "<<matriz[1][2]<<" "<<endl;**

**cout<<"\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" <<endl;**

**cout<<" | | "<<endl;**

**cout<<" "<<matriz[2][0]<<" | "<<matriz[2][1]<<" | "<<matriz[2][2]<<" "<<endl;**

**cout<<" | | "<<endl;**

**cout<<endl;**

**while ((simbolo != "X") && (simbolo != "O")) {**

**cout<<"Primeiro jogador, escolha seu símbolo, X ou O: ";**

**cin>>simbolo;**

**cout<<endl <<endl;**

**}**

**for (int count = 0; count < 9; count++) {**

**cout<<endl <<endl <<"Vez do jogador " <<jogador <<endl <<endl;**

**if (count > 0) {**

**if (simbolo == "X") {**

**simbolo = "O";**

**} else {**

**simbolo = "X";**

**}**

**}**

**if (jogador == 2) {**

**jogador = 1;**

**simbolo;**

**} else {**

**jogador = 2;**

**}**

**conferencia = false;**

**while (conferencia == false) {**

**linha = -1;**

**coluna = -1;**

**while ((linha != 0) && (linha != 1) && (linha != 2)) {**

**cout<<"Insira a linha [0, 1 ou 2] em que você quer marcar: ";**

**cin>>linha;**

**cout<<endl <<endl;**

**}**

**while ((coluna != 0) && (coluna != 1) && (coluna != 2)) {**

**cout<<"Insira a coluna [0, 1 ou 2] em que você quer marcar: ";**

**cin>>coluna;**

**cout<<endl <<endl;**

**}**

**if ((matriz[linha][coluna] != "X") && (matriz[linha][coluna] != "O")) {**

**matriz[linha][coluna] = simbolo;**

**conferencia = true;**

**} else {**

**cout<<"--------------------------------" <<endl;**

**cout<<"Essa posição já está preenchida!" <<endl;**

**cout<<"--------------------------------" <<endl <<endl;**

**}**

**}**

**cout<<" | | " <<endl;**

**cout<<" "<<matriz[0][0] <<" | "<<matriz[0][1]<<" | "<<matriz[0][2]<<" "<<endl;**

**cout<<"\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" <<endl;**

**cout<<" | | " <<endl;**

**cout<<" "<<matriz[1][0] <<" | "<<matriz[1][1]<<" | "<<matriz[1][2]<<" "<<endl;**

**cout<<"\_\_\_\_\_|\_\_\_\_\_|\_\_\_\_\_" <<endl;**

**cout<<" | | " <<endl;**

**cout<<" "<<matriz[2][0] <<" | "<<matriz[2][1]<<" | "<<matriz[2][2]<<" "<<endl;**

**cout<<" | | " <<endl;**

**cout<<endl;**

**marcadoX = 0;**

**marcadoO = 0;**

**for (countL = 0; countL < 3; countL++) {**

**for (countC = 0; countC < 3; countC++){**

**if (matriz[countL][countC] == "X"){**

**marcadoX++;**

**} else if (matriz[countL][countC] == "O"){**

**marcadoO++;**

**}**

**}**

**if (marcadoX == 3) {**

**cout<<"X ganhou!";**

**encerrou = true;**

**break;**

**} else if (marcadoO == 3) {**

**cout<<"O ganhou!";**

**encerrou = true;**

**break;**

**}**

**marcadoX = 0;**

**marcadoO = 0;**

**}**

**for (countC = 0; countC < 3; countC++) {**

**for (countL = 0; countL < 3; countL++){**

**if (matriz[countL][countC] == "X"){**

**marcadoX++;**

**} else if (matriz[countL][countC] == "O"){**

**marcadoO++;**

**}**

**}**

**if (marcadoX == 3) {**

**cout<<"X ganhou!";**

**encerrou = true;**

**break;**

**} else if (marcadoO == 3) {**

**cout<<"O ganhou!";**

**encerrou = true;**

**break;**

**}**

**if (count == 8) {**

**cout<<"Foi um empate!";**

**break;**

**}**

**marcadoX = 0;**

**marcadoO = 0;**

**}**

**if (encerrou == true) {**

**break;**

**}**

**}**

**return 0;**

**}**