**Написать программу, в которой имеется двумерный массив 5х5,**

**состоящий из случайных чисел.**

**Программа подсчитывает сумму элементов центральной диагонали.**

**Сумму элементов побочной диагонали.**

**Если сумма больше 200, то программа создает два одномерных динамических массива,**

**в один записывает четные элементы массива, в другой – нечетные.**

**Написать программу, в которой имеется двумерный массив 5х5,**

**состоящий из случайных чисел. Программа подсчитывает сумму элементов побочной диагонали.**

**Если сумма больше 200, то программа создает два одномерных динамических массива,**

**в один записывает четные элементы массива, в другой – нечетные.**

**Если сумма меньше 200, то создается три одномерных динамических массива,**

**первый - с элементами большими 10 из первых двух столбцов,**

**второй - с элементами меньшими 30 из последних двух столбцов**

**и третий динамический массив заполняется элементами среднего столбца,**

**которые больше 50. Реализовать решение при помощи функций.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

#include <iostream>

#include <Windows.h>

#include <ctime>

#include <iomanip>

using namespace std;

void sozdal(int arr[][5], int size);

void Vyvod(int arr[][5], int size);

void Poboch\_Diag(int arr[][5], int size);

void Centr\_Diag(int arr[][5], int size);

void Kallich\_element(int arr[][5], int size);

void Chet\_Nechet\_Poboch\_Diag(int arr[][5], int size);

int main()

{

system("chcp 1251");

srand(time(0));

int const size = 5;

int arr[size][size];

sozdal(arr, size);

Vyvod(arr, size);

cout << endl;

Centr\_Diag(arr, size);

Poboch\_Diag(arr, size);

Chet\_Nechet\_Poboch\_Diag(arr, size);

Kallich\_element(arr, size);

return 0;

}

void sozdal(int arr[][5], int size)

{

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

arr[i][j] = rand() % 100;

}

}

}

void Vyvod(int arr[][5], int size)

{

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

cout << setw(5)<< arr[i][j] << " |";

}

cout << "\n";

}

}

//сумму элементов центральной диагонали с выводом

void Centr\_Diag(int arr[][5], int size)

{

int sum = 0;

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i == j)

sum += arr[i][j];

}

}

cout << " Сумма центральной диагонали равна: " << sum << endl;

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i == j)

cout << arr[i][j] << " ";

}

}

cout << endl;

}

//подсчитывает сумму элементов побочной с выводом

void Poboch\_Diag(int arr[][5], int size)

{

int sum = 0;

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i + j == size - 1)

sum += arr[i][j];

}

}

cout << " Сумма побочной диагонали равна: " << sum << endl;

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i + j == size - 1)

cout << arr[i][j] << " ";

}

}

cout << endl;

}

//Если сумма элементов побочной диагонали больше 200, то программа создает два одномерных динамических массива,

//в один записывает четные элементы массива, в другой – нечетные.

void Chet\_Nechet\_Poboch\_Diag(int arr[][5], int size)

{

int sum = 0;

int count\_even = 0;

int count\_odd = 0;

int a = 0, b = 0;

int\* arr1 = new int [a];

int\* arr2 = new int [b];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i + j == size - 1)

sum += arr[i][j];

}

}

if (sum > 200)

{

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (arr[i][j] % 2 == 0)

{

arr1[a] = arr[i][j];//переписываем чет

a++;

count\_even++;

}

else

{

arr2[b] = arr[i][j];//переписываем чет

b++;

count\_odd++;

}

}

}

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

cout << arr1[i] << ",";

cout << endl;

cout << " Количество четных элементов равно: " << count\_even << endl;

for (int i = 0; i < b; i++)

cout << arr2[i] << ",";

cout << endl;

cout << " Количество нечетных элементов равно: " << count\_odd << endl;

}

}

//Если сумма элементов побочной диагонали меньше 200, то создается три одномерных динамических массива,

//первый - с элементами большими 10 из первых двух столбцов,

//второй - с элементами меньшими 30 из последних двух столбцов

//и третий динамический массив заполняется элементами среднего столбца,

//которые больше 50.

void Kallich\_element(int arr[][5], int size)

{

int sum = 0;

int count\_desiat = 0;

int count\_tridcat = 0;

int count\_piatdesiat = 0;

int a = 0, b = 0,v=0;

int\* arr1 = new int[a];

int\* arr2 = new int[b];

int\* arr3 = new int[v];

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (i + j == size - 1)

sum += arr[i][j];

}

}

if (sum < 200)

{

for (int i = 0; i < size; i++)

{

for (int j = 0; j < size; j++)

{

if (arr[i][j] > 10 && i < size && j <= 1)

{

arr1[a] = arr[i][j];//переписываем чет

a++;

count\_desiat++;

}

else if (arr[i][j] < 30 && i < size && j >= 3)

{

arr2[b] = arr[i][j];//переписываем чет

b++;

count\_tridcat++;

}

else if (arr[i][j] > 50 && i < size && j == 2)

{

arr3[v] = arr[i][j];//переписываем чет

v++;

count\_piatdesiat++;

}

}

}

cout << " Количество элементов более 10 из первых двух столбцов: " << count\_desiat << endl;

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

cout << arr1[i] << ",";

cout << endl;

cout << " Количество элементов менее 30 из последних двух столбцов: " << count\_tridcat << endl;

for (int i = 0; i < b; i++)

cout << arr2[i] << ",";

cout << endl;

cout << " Количество элементов более 50 из среднего столбца: " << count\_piatdesiat << endl;

for (int i = 0; i < v; i++)

cout << arr3[i] << ",";

cout << endl;

}

}