#include <iostream>

#include "extention.hpp"

//template <typename T>

//void bauble\_sort(T arr[], const int n)

//{

// T change = 0; //для замены

// for (int i = 0; i < n - 1; i++) {

// for (int j = 0; j < n - i - 1; j++) {

// if (arr[j] > arr[j+1])

// {

// change = arr[j];

// arr[j] = arr[j + 1];

// arr[j + 1] = arr[j];

// change = 0;

// }

// }

// }

//}

using namespace std;

using namespace ext;

int main()

{

setlocale(LC\_ALL, "Rus");

// //Задание 1

// //Объявите массив для хранения 10 - ти целочисленных элементов

// int arr1[10];

// //Объявите константный массив для хранения 20 - ти вещественных элементов

// const float arr2[20] = {};

// //Объявите массив для хранения 2-х символьных элементов

// char arr3[2];

// //Пусть есть массив объявленный следующим образом:

// int data[6] = { 2, 4, 8, 16, 32, 64 };

// //Какой размер имеет данный массив?

// //6

// //Данные какого типа он хранит

// //целочисленного

// //Чему равен элемент массива с индексом 0?

// //2

// //Чему равен элемент массива с индексом 3?

// //16

// //Какой индекс у элемента 8?

// //2

// //Какой индекс у элемента 32?

// //4

// //Присвойте первому элементу массива значение 1.

// data[1] = 1;

// //Утройте значение последнего элемента в массиве.

// data[6] = data[6] \* 3;

// //Выведите элементы массива на экран через пробел.

// for (int i = 0; i < 6; ++i)

// {

// cout << data[i] << " ";

// }

// //Объявите массив для хранения 5 - ти элеменов любого типа и проинизиализируете его с помощью списка инициализации

// int arr5[5] = { 1,2,3,4,5 };

// //Выведите объявленный массив на экран.

// cout << endl;

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

// {

// cout << arr5[i] << " ";

// }

// //Можно ли не указывать размер массива при его создании ?

// //можно, но только при условии наличия НЕ пустого списка инициализации

// //Если можно, то продемострируйте это.

// int arr4[] = { 1,2,3 };

//

// //Пусть массив объявили следующим образом :

// int data2[5] = { 1, 2 };

// //Каков размер данного массива ?

// //5

// //Чему равен элемент с индексов 1 ? Чему равен элемент с индексом 2 ?

// //С индексом 1 - 2; с индексом 2 - 0

// //Проверьте ответы с помощью дебагера.

// //Выведите все элементы массива на экран.

// cout << endl;

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

// {

// cout << data2[i] << " " << endl;

// }

// //Пусть массив объявлен следующим образом :

// const char alpha[3] = { 'a', 'b', 'c' };

// //Каков размер данного массива ?

// //3

// //Чему равен элемент с индексов 0 ? Чему равен индекс у элемента 'c'

// //С индексом 0 - 'a';у жлемента 'c' - 2

// //Допустимы ли следующие выражения :

// alpha[0];

// //alpha[3];

// //нет, тк. индекс находится вне границ массива

// //alpha[1] = 'd';

// //alpha[1] = 'b';

// //нет, тк. массив является константным, его значения менять нельзя

// //Выведите все элементы массива на экран.

// cout << endl;

// for (int i = 0; i < 3; ++i)

// {

// cout << alpha[i] << " " << endl;

// }

//

//

//

// /\*ЗАДАНИЕ 2\*/

// int arr[20];

//

// for (int i = 0; i < 20; ++i)

// {

// arr[i] = i;

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

// }

// cout << endl;

//

//

// for (int i = 0; i < 20; ++i)

// {

// arr[i] = i \* 2;

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

// }

// cout << endl;

//

//

// for (int i = 0; i < 20; ++i)

// {

// arr[i] = 0;

// }

//

//

// cout << endl;

//

// arr[0] = 1;

// int k = 0;

// for (int j = 1; j < 20; ++j)

// {

// arr[j] = arr[j - 1] \* 2;

// k += 1;

// if (k % 5 == 0)

// {

// for (int i = k - 5; i < k; ++i) { cout << arr[i] << " "; }

// cout << endl;

// }

// }

//

//

//

// /\*ЗАДАНИЕ 3\*/

// int data[10];

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

// {

// data[i] = ext::GetRandomValue(-10, 10, true);

// }

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

// {

// cout << data[i] << " ";

// }

// cout << endl;

// int max = data[0];

// int max\_ind = 0;

// for (int i = 1; i < 10; ++i)

// {

// if (data[i] > max) { max = data[i]; max\_ind = i; }

// }

// cout << "max " << max << " max index " << max\_ind << endl;

//

//

//int data[100];

//int sum1 = 0, sum2 = 0, sum3 = 0, sum = 0;

//for (int i = 0; i < 100; ++i)

//{

// data[i] = ext::GetRandomValue(-100, 100, true);

// sum = sum + data[i];

// if (data[i] > 0) { sum1++; }

// else if (data[i] < 0) { sum2++; }

// else { sum3++; }

//}

//cout << "Сумма всех = " << sum << endl;

//cout << "Сумма всех, которые больше 0 = " << sum1 << endl;

//cout << "Сумма всех, которые меньше 0 = " << sum2 << endl;

//cout << "Сумма всех, которые 0 = " << sum3 << endl;

//int arr[100];

//int first = 0, second = 0;

//for (int i = 0; i < 100; ++i)

//{

// arr[i] = ext::GetRandomValue(1, 100, true);

// //тут можно было запихнуть вычисление цифр и проверку в этот же цикл, сделав сложную,

// //но компактную программу, но делать я этого не буду)

//}

//for (int i = 0; i < 100; ++i)

//{

// if (arr[i] != 100)

// {

// first = arr[i] / 10;

// second = arr[i] % 10;

// }

// else if (arr[i] == 100) { continue; }

// if ((first == 2) & (second == 2)) { cout << arr[i] << " "; }

// if ((first % 2 == 0) && (second % 2 == 0)) { cout << arr[i] << " " << endl; }

//}

//int arr2[100];

//for (int i = 0; i < 100; ++i)

//{

// arr2[i] = arr[i] + data[i];

//}

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

//{

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

// {

// cout << arr2[j + (i \* 10)] << " ";

// }

// cout << endl;

//}

//int arr[100];

//int count = 0, more\_than = 0, index = -1;

//for (int i = 0; i < 100; ++i)

//{

// arr[i] = GetRandomValue(1, 100, true);

//}

//cout << "Введите число из диапозона от 1 до 100 " << endl;

//cin >> count;

//for (int i = 0; i < 100; ++i)

//{

// if (arr[i] == count)

// {

// index = i;

// }

// else if (arr[i] > count) { more\_than = more\_than + 1; }

//}

//cout << "Индекс вашего элемента 7" << index << endl;

//cout << "Кол-во элементов больших, чем ваше число " << more\_than << endl;

//

////ЗАДАНИЕ 4

//int arr[100];

//for (int i = 0; i < 100; ++i)

//{

// arr[i] = GetRandomValue(1,100, true);

//}

//int n;

//cout << "Введите длину массива " << endl;

//cin >> n;

//bauble\_sort(arr,n);

//for (int i = 0; i < 100; ++i)

//{

// cout << arr[i];

//}

//НЕ РАБОТАЕТ

///\*ЗАДАНИЕ 5\*/

//int data[3][2] = { { 2, 4 },{ 8, 16 },{ 32, 64 } };

///\*

// в этом массиве 3 строки

// в каждой строке 2 элемента

// массив имеет размер для хранения 8 элементов

// массив хранит целочисленные элементы

// обратиться к элементу со значением 8 - data[1][0]

// обратиться к элементу со значением 64 - data[2][1]

// data[0][0] = 1

// data[2][1] = data[2][1] \* 2

// \*/

//int data2[4][2] = { {1,2},{3,4},{5,6},{7,8} };

//int data3[][2] = { { 1,2 },{ 3,4 },{ 5,6 },{ 7,8 } };

////можно не указывать размер только первого измерения(строк)

//int data4[][4] = { { 1, 2 },{ 3,4,5 } };

// /\*

//размер - на 8 элементов

//data[0][0] равен 1

//data[1][2] равен 5

//\*/

//for (int i = 0; i < 2; ++i)

//{

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

// {

// data4[i][j] = data4[i][j] \* 2;

// cout << data4[i][j] << endl;

// }

//}

//

////system("pause");

}