Задача 5. Одномерные массивы

Исходные данные для всех вариантов – n вещественных величин. При написании программ можно использовать как динамические, так и не – динамические массивы. Размерность последних задавать именованной константой.

1. Найти максимальный по модулю элемент массива.
2. Найти сумму элементов массива, расположенных между первым и вторым отрицательными элементами .
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает 2, а потом все остальные.

#include "stdafx.h"

#include <conio.h>

#include <iostream>

#include <cstdlib>

#include <locale>

using namespace std;

int main()

{

 setlocale(LC\_ALL, "");

 int n;

 cout << "Введите количество элементов массива: ";

 cin >> n;

 int\* arr = new int[n]; //Создание нового массива

 //Заполнение массива случайными числами

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

 arr[i] = rand()%101 - 50;

 //Вывод массива на экран

 cout << "Исходный массив: ";

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

 cout << arr[i] << ' ';

 //Поиск максимального по модулю элемента массива

 int max = arr[0];

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

 if(abs(arr[i])>max)

 max = abs(arr[i]);

 //Вывод максимального по модулю элемента на экран

 cout << "\n\nМаксимальный по модулю элемент = " << max;

 //Поиск суммы элементов массива, расположенных между первым и вторым отрицательными элементами

 //Если отрицательных элементов в массиве нет, то считается сумма всех элементов массива

 int sum = 0;

 int first = 0; //Индекс первого отрицательного элемента

 int second = n-1; //Индекс второго отрицательного элемента

//Поиск первого отрицательного элемента

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

 if(arr[i]<0) {

 first = i;

 break;

 }

 //Поиск второго отрицательного элемента

 for(int i=first+1; i<n; ++i)

 if(arr[i]<0) {

 second = i;

 break;

 }

 //Подсчет суммы

 for(int i=first+1; i<second; ++i)

 sum += arr[i];

 //Вывод суммы на экран

 cout << "\nСумма элементов между первым и вторым отрицательным = " << sum;

 //Преобразование массива таким образом, чтобы сначала располагались все элементы,

 //модуль которых не превышает 2, а потом все остальные.

 int\* tmp = new int[n]; //Массив для хранения промежуточных данных

 int j = 0; //Переменная для прохода по временному массиву

 //Перебор всех элементов с модулем <= 2

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

 if(abs(arr[i])<=2)

 tmp[j++] = arr[i];

 //Перебор всех элементов с модулем > 2

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

 if(abs(arr[i])>2)

 tmp[j++] = arr[i];

 //Копирование элементов из временного массива в исходный массив и вывод на экран

 cout << "\nМассив после преобразования: ";

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

 arr[i] = tmp[i];

 cout << arr[i] << ' ';

 }

 delete[] tmp; //Удаление временного массива

 delete[] arr; //Удаление массива

 cout << endl << endl;

 \_getch();

 ;

}

Задача 8. Функции

Функции и массивы

Решить задачу 5 (Одномерные массивы, уже решена, код выше) и оформить каждый пункт задания в виде функции. Все необходимые данные для функции должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.