МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

### федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

(ГУАП)

КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

Преподаватель

канд. техн. наук, доцент Л.Н. Бариков

Отчёт

по лабораторной работе №7

по дисциплине ОСНОВЫ ПРОГРАММИРОВАНИЯ

на тему: «Статические одномерные массивы»

Работу выполнил

студент гр. 4941 Н.С. Горбунов

Санкт-Петербург

2020

***Цель лабораторной работы:*** *изучение структурной организации массивов и способов доступа к их элементам; совершенствование навыков процедурного программирования на языке C/C++ при решении задач обработки статических одномерных массивов.*

***Задание на программирование:*** *используя технологию процедурного программирования, разработать программу обработки одномерных статических массивов в соответствии с индивидуальным заданием.*

***Порядок выполнения работы:***

1. Получить у преподавателя индивидуальное задание и выполнить постановку задачи: сформулировать условие, определить входные и выходные данные, их ограничения.

2. Разработать математическую модель: описать с помощью формул и рисунков структуру массивов и процесс их преобразования.

3. Построить схемы алгоритмов основных функций и задачи в целом.

4. Составить программу на языке *C/C++*.

5. Входные данные на этапах тестирования и демонстрации работы преподавателю должны задаваться либо с использованием специально подобранных арифметических формул, либо вводиться с клавиатуры по запросу. **Датчики псевдослучайных чисел использовать запрещается**.

Если это явно не оговорено в конкретном варианте задания, значения элементов в каждом исходном массиве **должны быть разными**.

Выходные данные должны выводиться на экран с пояснениями.

6. Проверить и продемонстрировать преподавателю работу программы на полном наборе тестов, в том числе с ошибочными входными данными. Входные и выходные массивы должны выводиться в одном и том же формате.

7. Использовать стандартные потоковые объекты ввода/вывода ***cin*** и ***cout***.

8. Оформить отчет о лабораторной работе в составе: постановка задачи, математическая модель, схема алгоритма решения, текст программы, контрольные примеры (скриншоты).

**Задание 13**

Дан массив *y*0, *y*1,…, *y*2*n-*1. Определить количество и произведение положительных значений элементов массива, лежащих между элементом с наименьшим по абсолютной величине значением и элементом с номером *n*-1.

**Математическая модель**

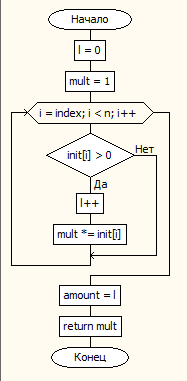
По условию задан массив с четным числом элементов. Тогда пользователь должен ввести половину размера массива. Далее следует в цикле перебрать каждый элемент массива и задать его значение пользователь.

Для решения задачи необходимо найти в массиве наименьший по абсолютной величине значения элемент и сохранить его индекс.

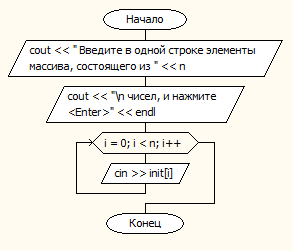
Далее следует перебрать все элементы массива, начиная с элемента с наименьшей абсолютной величиной и до конца. Каждый элемент массива следует проверить на положительность. При выполнении условия увеличить счетчик на один и найти произведения элемента с предыдущими.

В конце вывести значения счетчика и произведения. Задача решена.

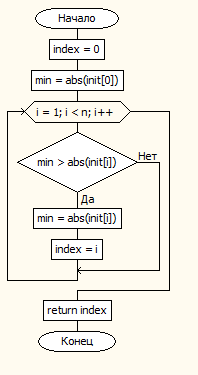
**Блок-схема arr\_mult**

****

**Блок-схема arr\_init**

****

**Блок-схема FindABS**

****

**Текст программы**

#include <iostream>

using namespace std;

const int RAZ = 10; //предельный размер массива

typedef int telem; //определение типа значений элементов массива

typedef telem tmas[RAZ]; //определение типа массива

void Arr\_Init(tmas init, int n);

int FindABSMin(tmas init, int n);

telem ArrMult(tmas init, int n, int index, int\* amount);

int main()

{

setlocale(LC\_ALL, "Russian");

tmas arr;

int N, amount = 0, MinIndex, MaxIndex;

cout << "Введите половину рамера массива <" << RAZ/2 << " ";

cin >> N;

if (N > RAZ / 2) { return 0; }

Arr\_Init(arr, 2\*N);

int Index = FindABSMin(arr, 2\*N);

if (Index < N - 1)

{

MinIndex = Index;

MaxIndex = N - 1;

}

else

{

MinIndex = N - 1;

MaxIndex = Index;

}

telem mult = ArrMult(arr, MaxIndex, MinIndex, &amount);

cout << "Произведение " << amount << " положительных элементов массива равна " << mult;

}

void Arr\_Init(tmas init, int n)

{

cout << " Введите в одной строке элементы массива, состоящего из " << n;

cout << "\n чисел, и нажмите <Enter>" << endl;

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

cin >> init[i];

}

int FindABSMin(tmas init, int n)

{

int index = 0;

telem min = abs(init[0]);

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

{

if (min > abs(init[i]))

{

min = abs(init[i]);

index = i;

}

}

return index;

}

telem ArrMult(tmas init, int n, int index, int\* amount )

{

int l = 0;

telem mult = 1;

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

{

if (init[i] > 0)

{

l++;

mult \*= init[i];

}

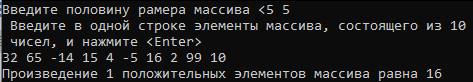
}

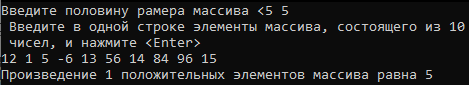
\*amount = l;

return mult;

}

**Пример работы программы**

****

****

**Вывод:** используя технологию процедурного программирования, разработал программу обработки одномерных статических массивов в соответствии с индивидуальным заданием.