Московский авиационный институт

(Национальный исследовательский университет)

Факультет № 3.

Системы управления, информатика и электроэнергетика.

Кафедра 304.

Отчет по лабораторной работе

по учебной дисциплине

«Информатика»

на тему

«Одномерные массивы»

Группа:М30-124Бк-17

Бригада № 8

Выполнили:

Бушманов Максим

Зайцев Евгений

Принял:

Макаров Е.А.

Чечиков Ю.Б

**Москва 2017г**

Оглавление

[Блок-схема 3](#_Toc496188683)

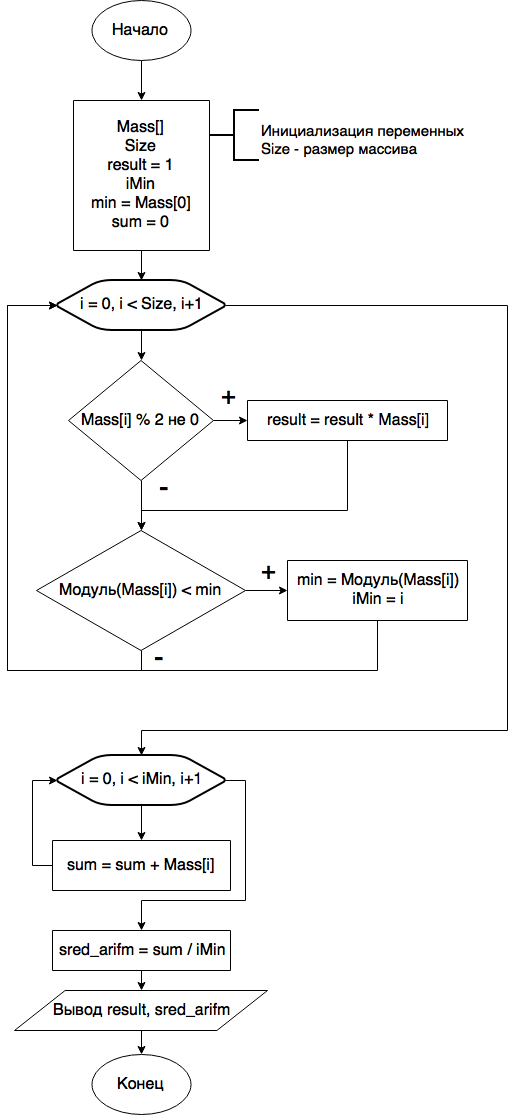
[Псевдокод 5](#_Toc496188684)

[Код программы(Программа) 7](#_Toc496188685)

[ТЕСТЫ 9](#_Toc496188686)

[Вывод 19](#_Toc496188687)

# Блок-схема

****

# Псевдокод

АЛГ Одномерные массивы

ПЕР

Mass[] // Заданный массив

Size // Размер массива

result = 1 // Результат перемножения нечетных элементов в массиве

iMin // Индекс минимального по модулю элемента в массиве

min = Mass[0] // Минимальное значение по модулю в массиве

sum = 0 // Сумма элементов до минимального значения по модулю в массиве

sred\_arifm // Среднее арифметическое элементов до минимального значения по модулю

НАЧАЛО

НЦ для i от 0 до i < Size

ЕСЛИ (Mass[i]%2 не 0) ТО

result = result \* Mass[i]

КОНЕСЛИ

ЕСЛИ (Модуль(Mass[i]) < min) ТО

min = Модуль(Mass[i])

iMin = i

КОНЕСЛИ

КЦ

НЦ для i от 0 до i < iMin

sum = sum + Mass[i]

КЦ

sred\_arifm = sum/iMin

Печать result, sred\_arifm

КОНАЛГ

# ****Код программы(Программа)****

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Progect name: Laba3 \*

Project type: Win32 Console Application \*

File name: main.cpp \*

Language: CPP, XCode 9, MSVS 2017 and above \*

Programmers: Bushmanov Maksim Sergeevich, Zaycev Evgeniy Pavlovich \*

Created: 3/11/2017 \*

Last revision: 9/11/2017 \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

using namespace std;

int main() {

int Mass[] = {2,3,4,6,-8,7,1,-9,6}; // Заданный массив

int Size = (sizeof(Mass)/sizeof(int)); // Размер массива

int result = 1; // Результат перемножения нечетных элементов в массиве

int iMin = -1; // Индекс минимального по модулю элемента в массиве

int min = Mass[0]; // Минимальное значение по модулю в массиве

double sum = 0;// Сумма элементов до минимального значения по модулю в массиве

double sred\_arifm;

// Проверяем каждый элемент массива

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

// Находим нечетные элементы в массиве

if ((Mass[i]%2) != 0){

result \*= Mass[i]; // Перемножаем их

}

// Находим минимальный по модулю элемент и его индекс

if (abs(Mass[i]) < min){

min = abs(Mass[i]);

iMin = i;

}

}

// Вычисление суммы элементов до минимального по модулю элемента

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

sum += Mass[i];

}

sred\_arifm = sum/iMin; // Вычисляем среднее арифметическое до минимального по модулю элементов

cout << "Перемноженные нечетные элементы в массиве: " << result << endl;

cout << "Минимальный по модулю элемент массива: " << min << endl;

cout << "Среднее арифметическое до минимального по модулю элемента: " << sred\_arifm << endl;

return 0;

}

# ТЕСТЫ

**I. Корректные тесты**

**Тест №1**

1) Цель: проверить реализацию программы, при A = B

2) Исходные данные: A = 4, B = 4;

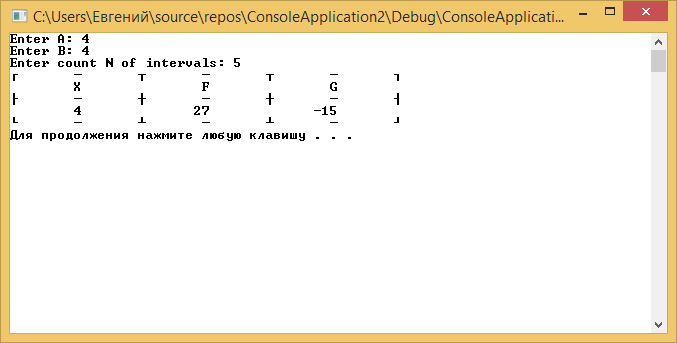
3) Протокол расчетов:

x = A = 4;

F(x) = (4 – 1) ^ 3 = 27;

G(x) = 1 – (4\*4) = -15;

4) Полученный результат:



5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

**Тест №2**

1) Цель: проверить работу программы, при A < B и N = 20

2) Исходные данные: N = 20, A = -1, B = 4;

3) Протокол расчетов:

h = (4 – (-1)) /20 =0,25;

N + 1 = 21

i = 0;

x = (-1) + (0 \* 0,25) = -1;

F(x) = ((-1) – 1) ^ 3 = -8;

G(x) = 1 – ((-1)\*(-1)) = 0;

i = 0 + 1 = 1;

i < N;

x = (-1) + (12 \* 0,25) = 2;

F(x) = (2 – 1) ^ 3 = 1;

G(x) = 1 – (2\*2) = -3;

i = 12 + 1 = 13;

i < N

x = (-1) + (20 \* 0,25) = 4;

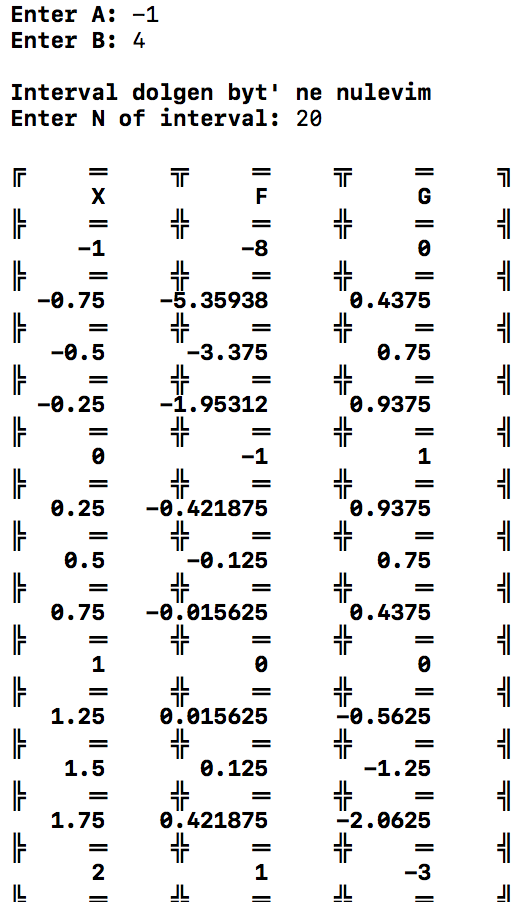
F(x) = (4 – 1) ^ 3 = 27;

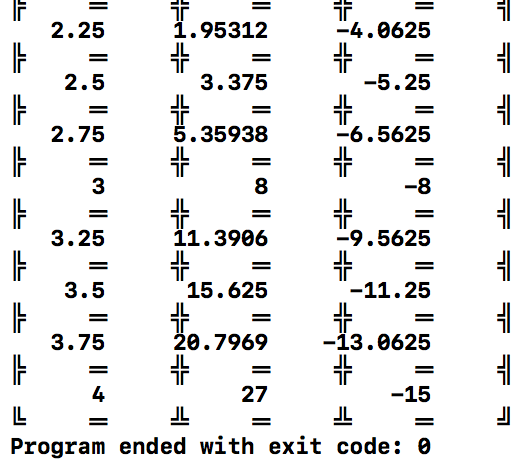
G(x) = 1 – (4\*4) = -15;

i = 20 + 1 = 21;

i > N;

4) Полученный результат:





5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

**Тест №3**

1) Цель: проверить чувствительность алгоритма, при N = 10 и A < B

2) Исходные данные: N = 10, A = 1, B = 3;

3) Протокол расчетов:

h = (3 – 1) /10 =0.2;

i = 0;

x = 1 + (0 \* 0.2) = 1;

F(x) = (1 – 1) ^ 3 = 0;

G(x) = 1 – (1\*1) = 0;

i = 0 + 1 = 1;

i < = N;

x = 1 + (1 \* 0.2) = 1.2;

F(x) = (1.2 – 1) ^ 3 = 0.008;

G(x) = 1 – (1.2\*1.2) = -0.44;

i = 1 + 1 = 2;

i < = N

x = 1 + (2 \* 0.2) = 1.4;

F(x) = (1.4 – 1) ^ 3 = 0.064;

G(x) = 1 – (1.4\*1.4) = -0.96;

i = 2 + 1 = 3;

i <= N;

x = 1 + (3 \* 0.2) = 1.6;

F(x) = (1.6 – 1) ^ 3 = 0.216;

G(x) = 1 – (1.6\*1.6) = -1.56;

i = 3 + 1 = 4;

i <= N;

x = 1 + (4 \* 0.2) = 1.8;

F(x) = (1.8 – 1) ^ 3 = 0.512;

G(x) = 1 – (1.8\*1.8) = -2.24;

i = 4 + 1 = 5;

i <= N;

x = 1 + (5 \* 0.2) = 2;

F(x) = (2 – 1) ^ 3 = 1;

G(x) = 1 – (2\*2) = -3;

i = 5 + 1 = 6;

i <= N;

x = 1 + (6 \* 0.2) = 2.2;

F(x) = (2.2 – 1) ^ 3 = 1.728;

G(x) = 1 – (2.2\*2.2) = -3.84;

i = 6 + 1 = 7;

i <= N;

x = 1 + (7 \* 0.2) = 2.4;

F(x) = (2.4 – 1) ^ 3 = 2.744;

G(x) = 1 – (2.4\*2.4) = -4.76;

i = 7 + 1 = 8;

i <= N;

x = 1 + (8 \* 0.2) = 2.6;

F(x) = (2.6 – 1) ^ 3 = 4.096;

G(x) = 1 – (2.6\*2.6) = -5.76;

i = 8 + 1 = 9;

i <= N;

x = 1 + (9 \* 0.2) = 2.8;

F(x) = (2.8 – 1) ^ 3 = 5.832;

G(x) = 1 – (2.8\*2.8) = -6.84;

i = 9 + 1 = 10;

i <= N;

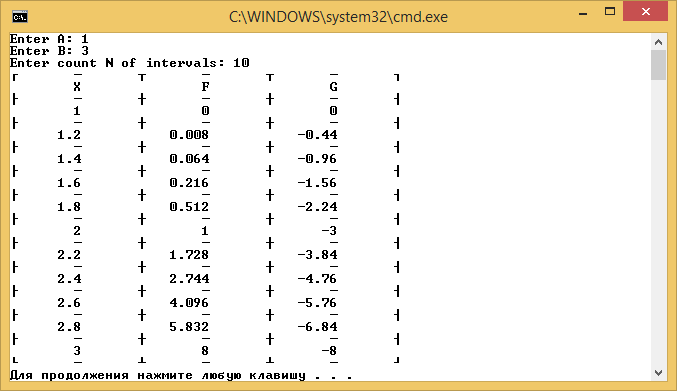
x = 1 + (10 \* 0.2) = 3;

F(x) = (3 – 1) ^ 3 = 8;

G(x) = 1 – (3\*3) = -8;

i = 10 + 1 = 11;

i > N;

4) Полученный результат: 

5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

**Тест №4**

1) Цель: проверить реализацию программы, при A > B и N = 2

2) Исходные данные: A = 3, B = 1, N =2;

3) Протокол расчетов:

A > B

C = A = 3;

A = B = 1;

B = C = 3;

h = (3 – 1) /2 =1;

i = 0;

x = 1 + (0 \* 1) = 1;

F(x) = (1 – 1) ^ 3 = 0;

G(x) = 1 – (1\*1) = 0;

i = 0 + 1 = 1;

i < = N;

x = 1 + (1 \* 1) = 2;

F(x) = (2 – 1) ^ 3 = 1;

G(x) = 1 – (2\*2) = -3;

i = 1 + 1 = 2;

i < = N

x = 1 + (2 \* 1) = 3;

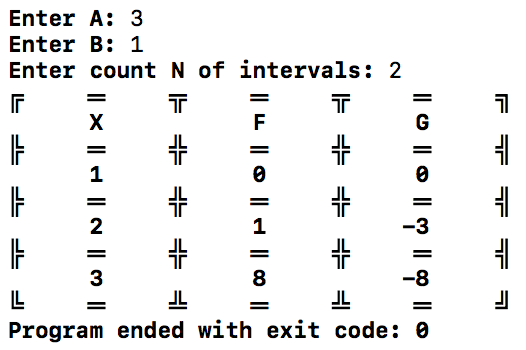
F(x) = (3 – 1) ^ 3 = 8;

G(x) = 1 – (3\*3) = -8;

i = 2 + 1 = 3;

i > N;

4) Полученный результат:



5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

**Тест №5**

1) Цель: проверить чувствительность алгоритма, при N = 1 и A < B

2) Исходные данные: N = 1, A = 1, B = 3;

3) Протокол расчетов:

h = (3 – 1) /1 = 2;

i = 0;

x = 1 + (0 \* 2) = 1;

F(x) = (1 – 1) ^ 3 = 0;

G(x) = 1 – (1\*1) = 0;

i = 0 + 1 = 1;

i < N;

x = 1 + (1 \* 2) = 3;

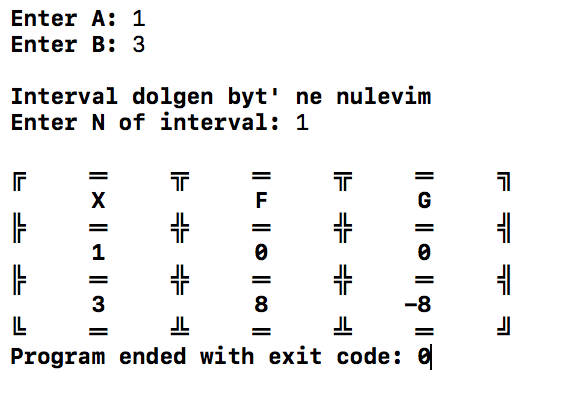
F(x) = (3 – 1) ^ 3 = 8;

G(x) = 1 – (3\*3) = -8;

i = 1 + 1 = 2;

i > N

4) Полученный результат:



5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

**II. Некорректные тесты**

**Тест №1**

1) Цель: проверить реализацию программы, при N = 0;

2) Исходные данные: N = 0, A = 1, B = 3;

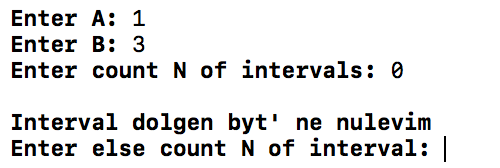
3) Протокол расчетов:

N < 1

Интервал должен быть не нулевым

Введите другое N:

4) Полученный результат:



5) Полученные результаты совпали с ожидаемыми. Тест ошибку не обнаружил.

# Вывод

Разработка программы завершена на основании, что:

1. Полученные результаты совпадают с ожидаемыми.
2. Считаем набор тестов полным.