**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: *М3О-110Б-19*

*Вариант №8*

Выполнили:

*Кочкин М.А.*

*Дмитриев Д.К.*

Приняли:

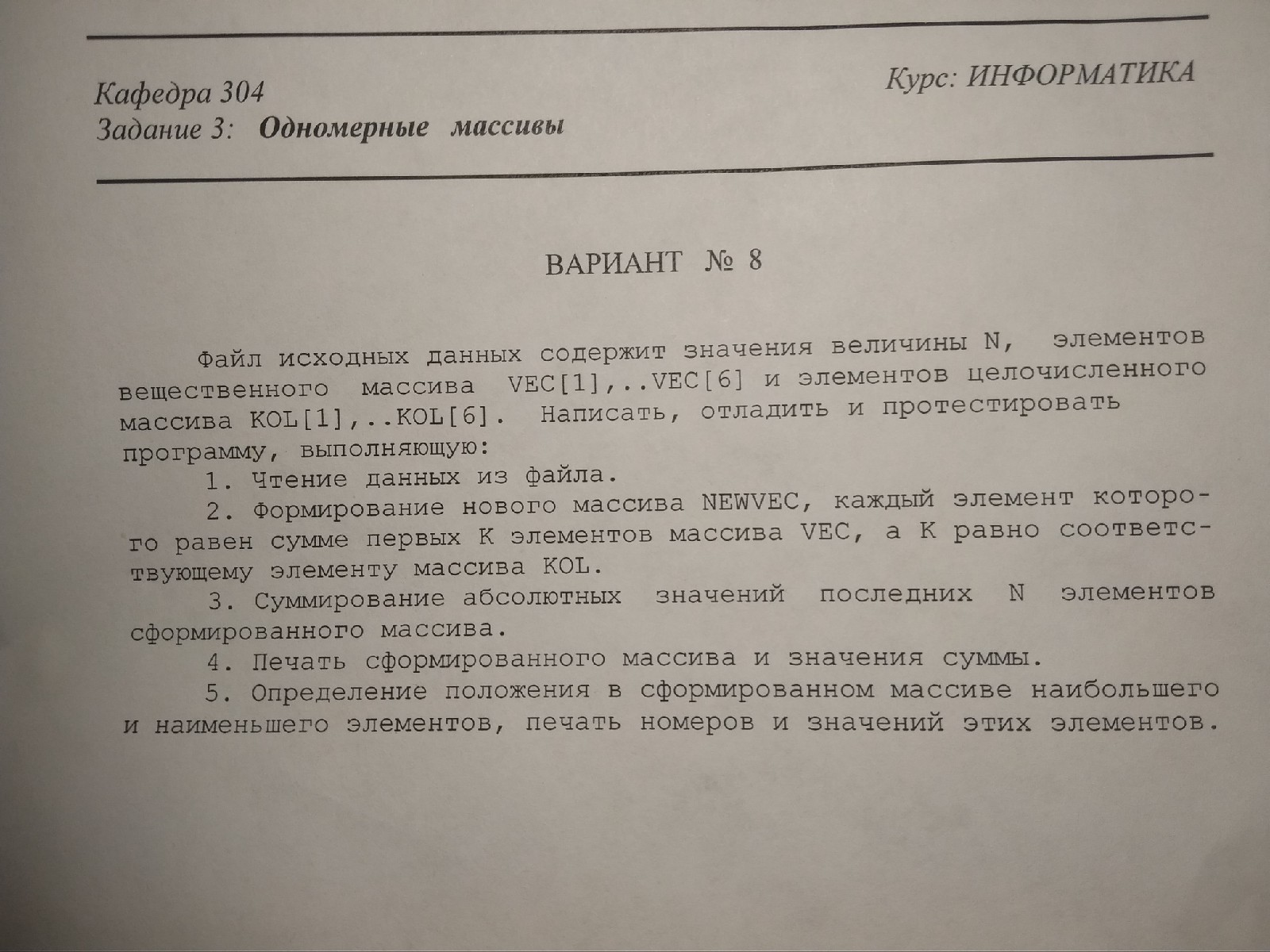
*Чечиков Ю.Б.*

Москва 2019

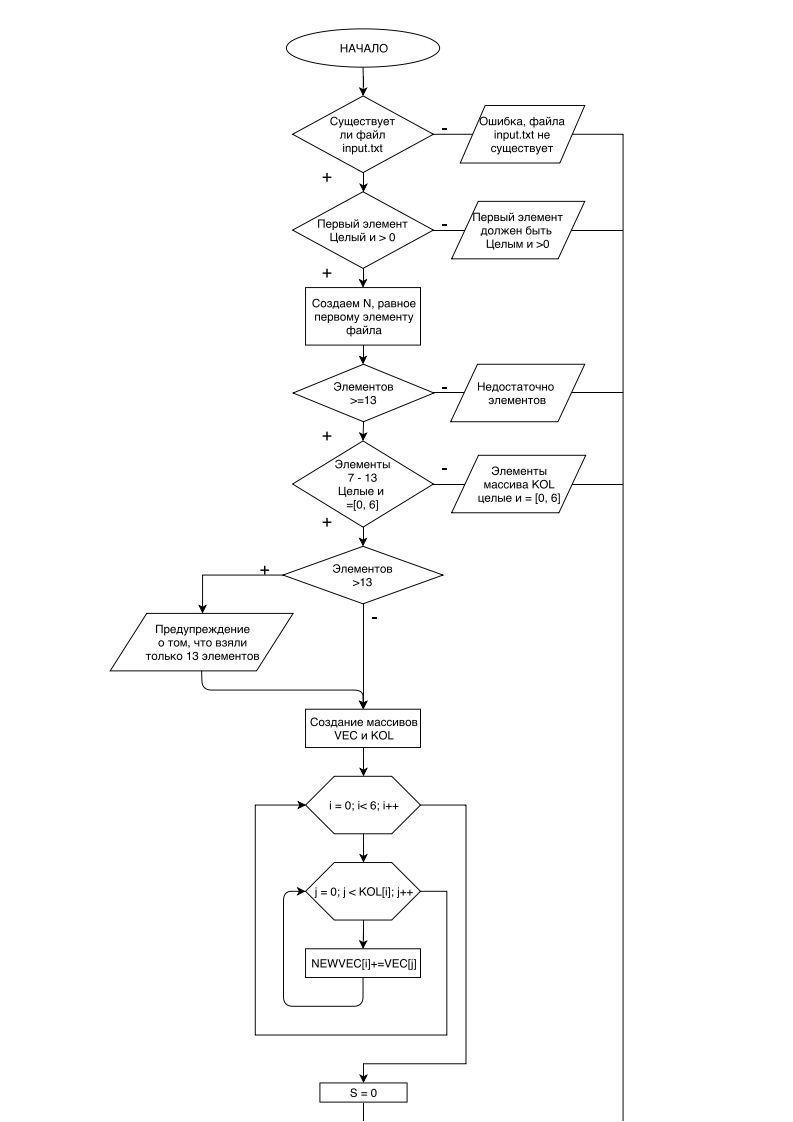
**Содержание**

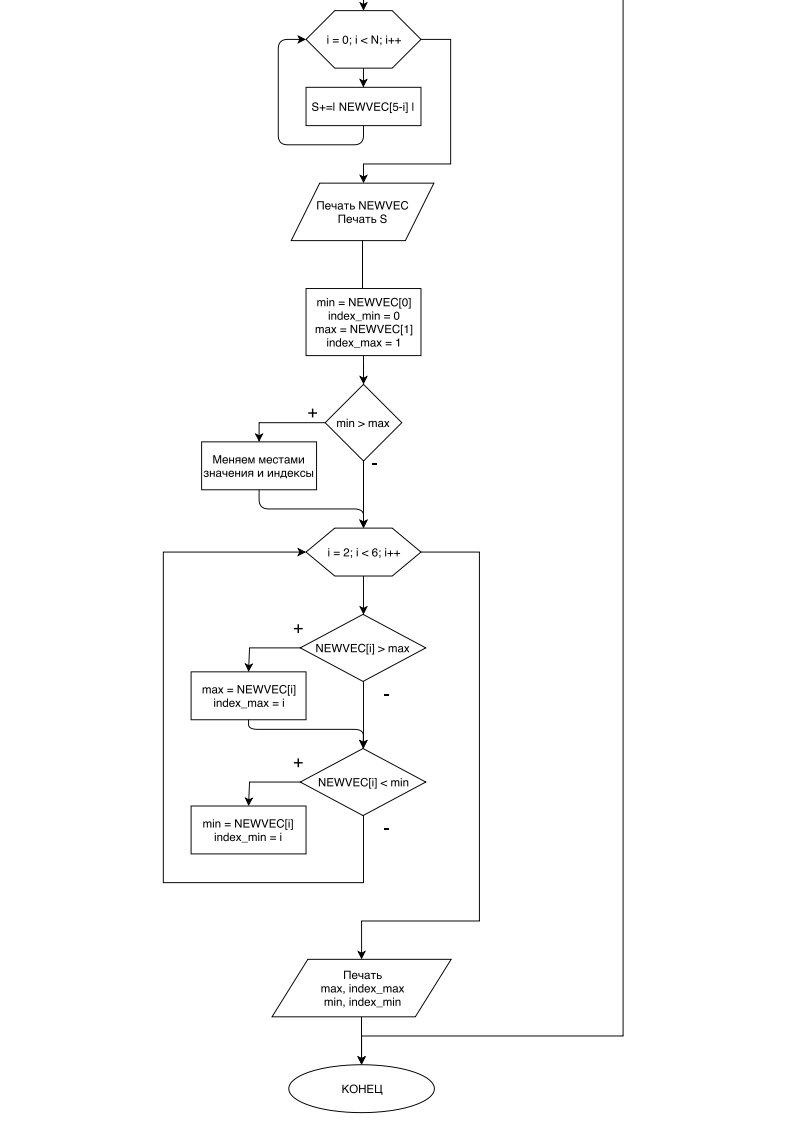
1. Задание
2. Блок-схема
3. Псевдокод
4. Код программы
5. Тесты программы
6. Вывод

**Задание**

****

**Блок-схема**

****

****

**Псевдокод**

Введем обозначения:

n – количество последних элементов сформированного массива

VEC – 1-ый вводимый массив

KOL – 2-ой вводимый массив

NEWVEC – сформированный массив

min – минимальный элемент массива NEWVEC

max – максимальный элемент массива NEWVEC

index\_min – индекс минимального элемента

index\_max – индекс максимального элемента

НАЧАЛО

Ввод файла

Проверка корректности файла

Проверка корректности данных из файла

Ввод n

Ввод VEC

Ввод KOL

Эхо печать n, VEC, KOL

ДЛЯ i от 0 до 6, шаг 1 ДЕЛАТЬ

НЦ

Для j от 0 до KOL[i], шаг 1 ДЕЛАТЬ

НЦ

NEWVEC[i] += VEC[j]

КЦ

КЦ

ДЛЯ i от 0 до n, шаг 1 ДЕЛАТЬ

НЦ

s += abs(NEWVEC[5 – i])

КЦ

Вывод NEWVEC

Вывод s

min = NEWVEC[0];

index\_min = 0;

max = NEWVEC[1];

index\_max = 1;

ЕСЛИ min > max

ТО

Меняем местами min, max

Меняем местами index\_min, index\_max

КОНЕЦ ЕСЛИ

ДЛЯ i от 2 до n, шаг 1 ДЕЛАТЬ

НЦ

ЕСЛИ NEWVEC[i] > max

ТО

max = NEWVEC[i]

index\_max = i

КОНЕЦ ЕСЛИ

ЕСЛИ NEWVEC[i] < min

ТО

min = NEWVEC[i]

index\_min = i

КОНЕЦ ЕСЛИ

КЦ

Вывод min, index\_min, max, index\_max

КОНЕЦ

**Код программы**

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

\*Project name :Lab3 \*

\*Project type :Win32 Console Application \*

\*File name :Lab3.cpp \*

\*Language :CPP, GCC 5.0 and above \*

\*Programmer :Кочкин Михаил Алексеевич, Дмитриев Дмитрий Константинович, М3О-110Б-19 \*

\*Modifiedby : Дмитриев Дмитрий Константинович \*

\*Created :18.11.2019 \*

\*Lastrevision :18.11.2019 \*

\*Comment :Одномерные массивы \*

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

#include <iostream>

#include <fstream>

using namespace std;

int main()

{

system("color f0");

//Открытие файла

ifstream file;

file.open("input.txt");

if (!file) //Проверка существования

{

cout << "Error! File 'input.txt.' doesn't exist" << endl;

system("pause");

return -1;

}

//Ввод N

double n;

file >> n;

if ( (n - int(n) != 0) | !(n > 0 && n < 7)) //Проверка N

{

cout << "First element should be natural and = [0, 6]" << endl;

system("pause");

return -1;

}

int count = 1; //Количество вводимых элементов

double temp;

//Подсчёт количества элементов

while (!file.eof())

{

file >> temp;

count++;

}

if (count < 13)

{

cout << "Not enough elements";

system("pause");

return -1;

}

file.seekg(0, ios::beg); //Переход к началу файла

double VEC[6];

file >> temp;

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

{

file >> VEC[i];

}

int KOL[6];

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

{

file >> temp;

//Проверка элементов массива KOL

if (temp - int(temp) != 0 || !(temp > 0 && temp < 7))

{

cout<<"The KOL elements should be natural and=[0,6]"<<endl;

system("pause");

return -1;

}

KOL[i] = int(temp);

}

if (count > 13)

{

cout << "Program took only 13 elements" << endl;

}

//Печать элементов файла, принятых в программу

cout << "N = " << n << endl;

cout << "VEC: " << endl;

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

{

cout << VEC[i] << " ";

}

cout << endl;

cout << "KOL: " << endl;

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

{

cout << KOL[i] << " ";

}

cout << endl;

int NEWVEC[6] = { 0 };

//Заполнение массива NEWVEC

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

{

for (int j = 0; j < KOL[i]; j++)

{

NEWVEC[i] += VEC[j];

}

}

int s = 0; //Сумма абсолютных значений последних N элементов NEWVEC

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

{

s += abs(NEWVEC[5 - i]);

}

cout << "NEWVEC:" << endl;

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

{

cout << NEWVEC[i] << " ";

}

cout << endl;

cout << "S = " << s << endl;

//Поиск наименьшего и наибольшего элементов и их индексов

int min = NEWVEC[0];

int index\_min = 0;

int max = NEWVEC[1];

int index\_max = 1;

if (min > max)

{

min = NEWVEC[1];

index\_min = 1;

max = NEWVEC[0];

index\_max = 0;

}

for (int i = 2; i < n; i++)

{

if (NEWVEC[i] > max)

{

max = NEWVEC[i];

index\_max = i;

}

if (NEWVEC[i] < min)

{

min = NEWVEC[i];

index\_min = i;

}

}

cout << "Max element: " << max << " with index: " << index\_max << endl;

cout << "Min element: " << min << " with index: " << index\_min << endl;

system("pause");

return 0;

}

**Корректные тесты**

**Тест 1.**

Цель: проверить работу программы при A = B.

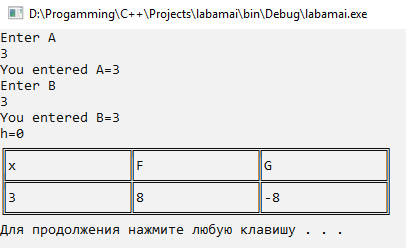
Входные значения: A=B=3.

Ожидаемый результат:



вывод в консоль однострочной таблицы с рассчитанными значениями для функций.

Работа программы:



*Полученные результаты равны ожидаемым.*

*Тест не выявил ошибки.*

**Тест 2.**

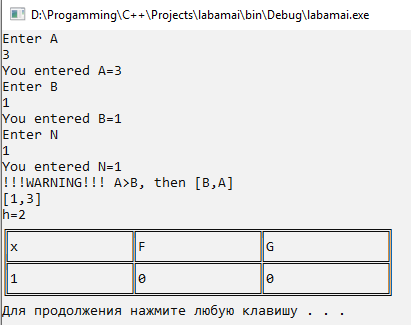
Цель: проверить работу программы при A < B.

Входные данные: A=3, В=1, N=1.

Ожидаемый результат: программа выведет предупреждение о замене местами A и B. поменяет местами A и B и выведет расчеты в виде таблицы.



Работа программы:



*Полученные результаты равны ожидаемым.*

*Тест не выявил ошибок.*

**Некорректные тесты**

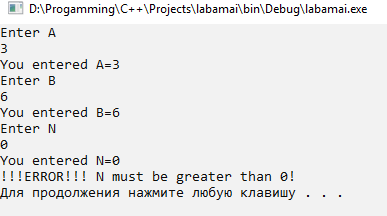
**Тест 1.**

Цель: протестировать программу с входными данными на границе допустимых N.

Входные данные: A=3, B=6, N=0.

Ожидаемый результат: вывод в консоль сообщения об ошибке ввода.

Работа программы:



*Полученные результаты равны ожидаемым.*

*Тест не выявил ошибок.*

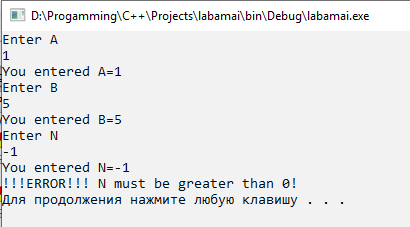
**Тест 2.**

Цель: протестировать программу с входными данными ниже границы допустимых N.

Входные данные: A=1, B=5, N=-1.

Ожидаемый результат: вывод в консоль сообщения об ошибке ввода.

Работа программы:



*Полученные результаты равны ожидаемым.*

*Тест не выявил ошибок.*

**Вывод**

Разработка программы успешно завершена, т.к.:

1. Полученные результаты совпадают с ожидаемыми;
2. Набор тестов считаем полным.