**Санкт-Петербургский государственный университет телекоммуникаций им.проф. М.А.Бонч-Бруевича**

**Факультет информационных систем и технологий**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Курсовая работа**

**по дисциплине «Алгоритмизация и программирование»**

Выполнил

студ.1 курса, гр.ИСТ311

направление подготовки 230400

\_\_\_\_\_\_\_\_\_\_\_\_\_ Иванов А.А.

«\_\_\_»\_\_\_\_\_\_\_\_ 2014

Принял

ст.пр.каф.БИС

\_\_\_\_\_\_\_\_\_\_\_\_\_

оценка

\_\_\_\_\_\_\_\_\_\_\_\_\_ Евстигнеев В.А.

«\_\_\_»\_\_\_\_\_\_\_\_ 2014

Санкт-Петербург, 2014

**Аннотация**

Целью проделанной работы было научиться применять полученные знания для самостоятельного решения на ПК конкретных практических задач.

По полученному заданию была спроектирована и написана программа, выполняющая требующиеся в задании действия. После этого был написан отчёт о проделанной работе.

В процессе работы были получены как навыки в разработке программ, так и опыт по части подготовки курсовых работ и отчётов по ним.

**Оглавление**

Аннотация ………………………………………………………………………….. 2

Оглавление ………………………………………………………………………..... 3

Постановка задачи …………………………………………………………………. 4

Блок-схема алгоритма ……………………………………………………………... 5

Текст программы …………………………………………………………………... 6

Руководство пользователя ………………………………………………………… 9

Заключение ............................................................................................................... 10

**Постановка задачи**

Текст задания:

# Вариант № 6

Разработать блок-схему алгоритма программы и программу, которая:

а) вводит с клавиатуры или читает из файла одномерный массив целых чисел размерностью [15] элементов и выводит его на экран. Элементы массива в файле расположены в виде одной строки;

б) многократно, в зависимости от запроса, производит подсчет количества отрицательных, или положительных, или нулевых элементов массива и выдает результат на экран;

в) по окончании работы сохраняет массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью [3 × 5] элементов с двумя пробелами между ними, а результаты количества отрицательных, положительных и нулевых элементов массива записываются в другой файл.

Таким образом, требуется составить по заданию блок-схему, иллюстрирующую принцип работы программы, и на её основе написать саму программу в любой удобной среде.

**Блок-схема алгоритма**

Начало

Неправильная команда

while(1)

Меню

Ввод массива вручную

Чтение массива из файла

default

1

2

i, c1=1, sum, sw, sw1, M[15]

sw1

while(c1)

Меню

switch(sw1)

1

1

1

1

1

Конец

Запись данных в файл

Сумма нулевых элементов

Сумма положительных

Сумма отрицательных

**Текст программы**

#include <cstdlib>

#include <iostream>

#include <fstream.h>

using namespace std;

int main(int argc, char\*\* argv)

{

int i, sw, sw1, c1=1, s1=0, s2=0, s3=0, M[15];

printf("\n\nМЕНЮ:\n1) Ввести массив с клавиатуры\n2) Чтение массива из файла\n\n");

while(c1)

{

cout<<"\nВведите команду: ";

cin>>sw1;

switch(sw1)

{

case 1:

{

for (i = 0; i <15; i++)

{

cout << "Введите " << i << " элемент массива: ";

cin >> M[i];

}

for (int i = 0; i <15; i++)

{

cout<<" "<<M[i];

}

c1=0;

}

break;

case 2:

{

std::ifstream file("massiv1.txt");

for (i = 0; i <15; i++)

{

file >> M[i];

}

cout<<"\nМассив:";

for (i = 0; i <15; i++)

{

cout<<" "<<M[i];

}

c1=0;

}

break;

default: printf("\nОшибка: неправильный символ.\n\n");

}

}

printf("\n\n\nМЕНЮ:\n1) Число отрицательных элементов\n2) Число положительных элементов\n3) Число нулевых элементов\n4) Запись даннных в файл\n5) Выход из программы");

while(1)

{

cout<<"\n\n\nВведите команду ";

cin>>sw;

switch(sw)

{

case 1:

{

for (i = 0; i <15; i++)

if (M[i]<0)

s1=s1+1;

cout<<"\nЧисло отрицательных элементов: "<<s1;

}

break;

case 2:

{

for (i = 0; i <15; i++)

if (M[i]>0)

s2=s2+1;

cout<<"\nЧисло положительных элементов: "<<s2;

}

break;

case 3:

{

for (i = 0; i <15; i++)

if (M[i]==0)

s3++;

cout<<"\nЧисло нулевых элементов: "<<s3;

}

break;

case 4:

{

FILE \*f1, \*f2;

f1=fopen("massiv2.txt", "w");

f2=fopen("S.txt", "w");

for (int j=0; j<5; j++)

{

for (int k=0; k<3; k++)

{

fprintf(f1," %d", M[k+(j\*3)]);

}

fprintf(f1, "\r\n");

}

fprintf(f2,"\n %d %d %d", s1, s2, s3);

fclose(f1);

fclose(f2);

cout<<"\nПроизведена запись в файл.";

}

break;

default: return 0;

}

}

}

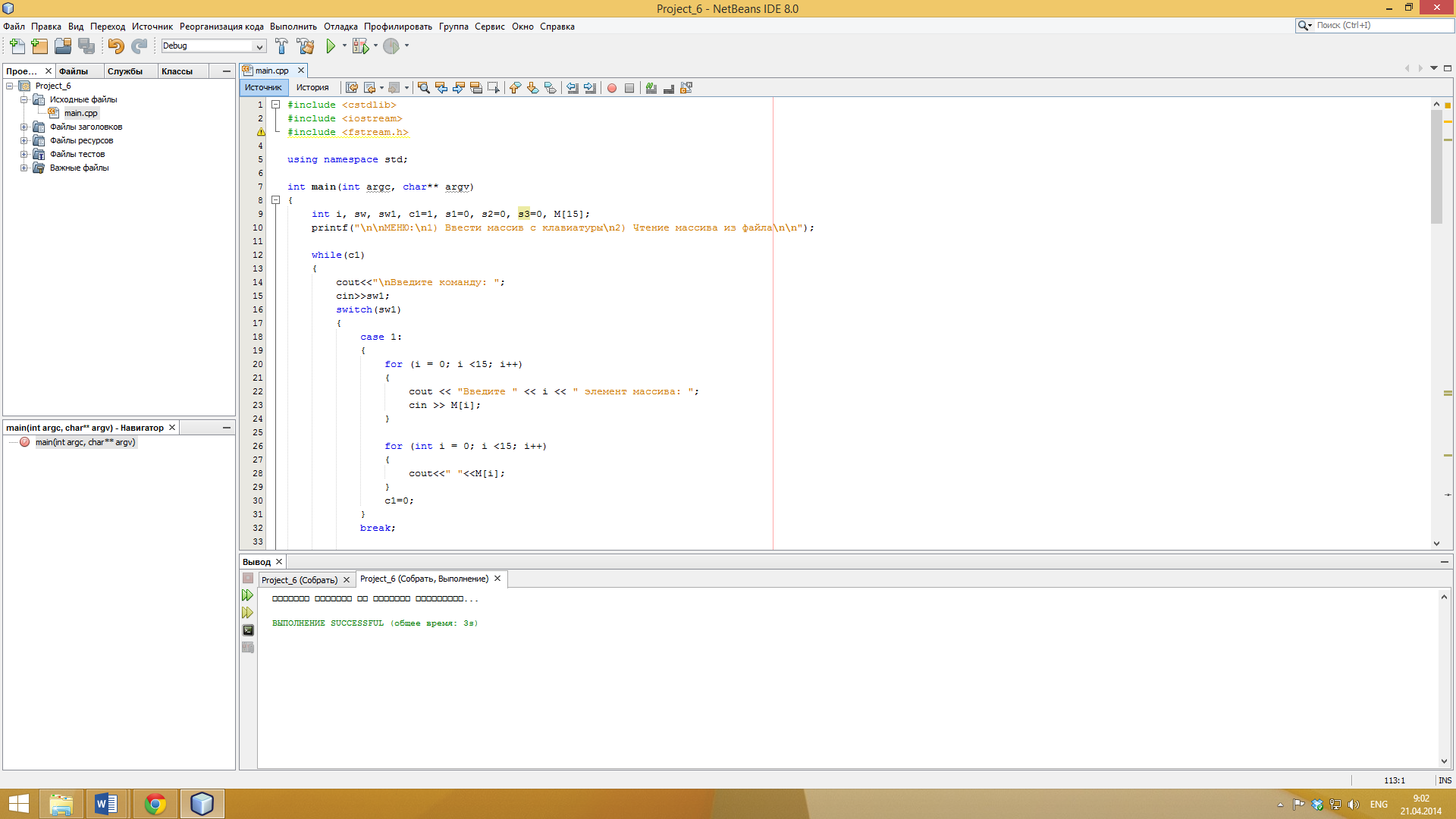
****

Рисунок 1 – Процесс разработки программы в NetBeans

**Руководство пользователя**

Программа разработана и запускается в среде NetBeans. Языком написания программы является C++.

Назначение программы – обработка одномерных целочисленных массивов, демонстрация чтения из файла и записи в файл.

В программу можно водить только целые числа, в том числе и в качестве управляющих команд. Введение символов может привести к сбою программы.

На первом этапе работы программы пользователю предлагается ввести массив, который программа будет обрабатывать, либо загрузить готовый массив из файла. Соответственно, команда «1» запускает ввод массива пользователем, команда «2» загружает готовый массив, и программа переходит ко второму этапу работы. Введение каких-любо других цифр на данном этапе безрезультативно.

На втором этапе можно произвести три действия с массивом. По команде «1» программа считает сумму всех элементов. Команда «2» позволяет изменить любой элемент массива; после её запуска нужно будет выбрать по номеру желаемый элемент и после этого ввести его новое значение. Наконец, команда «3» сохраняет полученный массив и значение суммы в файлы. Ввод каких-либо других чисел останавливает работу программы. Иными словами, для прекращения работы программы достаточно ввести какое-либо число кроме номеров команд.

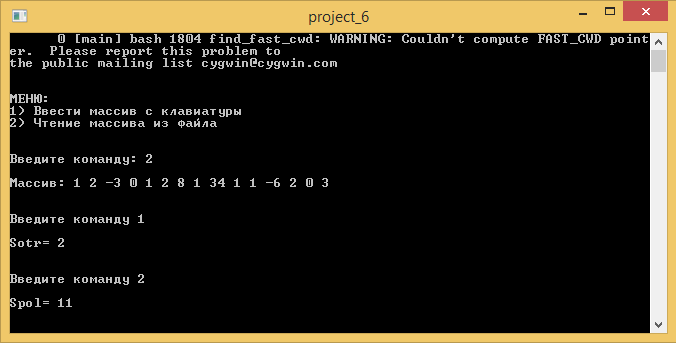


Рисунок 2 – Демонстрация работы программы

**ЗАКЛЮЧЕНИЕ**

Учебная задача решена.

По сути задача является поисковой, так как её конечный результат – демонстрационная программа, для написания которой требовалось знание определенных команд языка C++.

Для упрощения работы программы её интерфейс был написан на русском языке.

Наиболее сложным разделом пояснительной записки стало составление блок-схемы программы, для чего потребовалось определённое усердие и аккуратность.