**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.ИСТ511

направление подготовки

Абдуллаев Т.Ф

«\_\_\_»\_\_\_\_\_\_\_\_ 2016

Принял

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

оценка

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

«\_\_\_»\_\_\_\_\_\_\_\_ 2016

Санкт-Петербург, 2016

**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 № 1

Разработать блок-схему алгоритма программы и программу, которая:

а) вводит с клавиатуры или читает из файла двумерный массив вещественных значений размерностью [4 × 5] элементов и выводит его на экран. Вещественные элементы должны содержать не более 3-х знаков после запятой;

б) производит поиск *max* или *min* элемента в указанной строке матрицы и результат выдает на экран. Предусмотреть возможность многократного поиска, по запросу пользователя;

в) по окончании работы сохраняет матрицу в файле до следующего сеанса работы, а результаты поиска (*max* или *min* элемента для запрашиваемых строк) – в другом файле.

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)

Ввод массива вручную

Чтение массива из файла

while(c1)

Вывод меню

0

Ввод sw1

switch(sw1)

1 2

Вывод меню

1 2 3

Запись в файл

Конец

Минимальный элемент

Максимальный элемент

**Текстпрограммы**

#include<cstdlib>

#include <iostream>

#include <fstream>

using namespace std;

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

{

inti, j, sw, sw1, c1=1, a1, a2;

float min, max, M[4][5];

printf("\n\nМЕНЮ:\n1) Ввестимассивсклавиатуры\n2) Чтениемассиваизфайла\n\n");

while(c1)

{

cout<<"\nВведите команду: ";

cin>>sw1;

switch(sw1)

{

case 1:

{

cout<<"\nВведите элементы массива: \n";

for(i = 0; i< 4; i++)

{

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

{

cin>> M[i][j];

}

}

for(i = 0; i< 4; i++)

{

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

{

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

}

cout<<endl;

}

c1=0;

}

break;

case 2:

{

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

for (i = 0; i<20; i++)

{

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

{

file>> M[i][j];

}

}

cout<<"\nМассив: \n";

for(i = 0; i< 4; i++)

{

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

{

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

}

cout<<endl;

}

c1=0;

}

break;

default: printf("\nОшибка: неправильныйсимвол.\n\n");

}

}

printf("\n\n\nМЕНЮ:\n1) Поиск максимального\n2) Поиск минимального\n3) Запись данных в файл\n4) Выход из программы");

while(1)

{

cout<<"\n\n\nВведите команду ";

cin>>sw;

switch(sw)

{

case 1:

{

cout<<"\nВыберитестроку: \n";

cin>>a1;

max= M[a1][0];

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

{

if(M[a1][j]>max)

{

max=M[a1][j];

}

}

cout<<"\nMax: "<<max;

}

break;

case 2:

{

cout<<"\nВыберитестроку: \n";

cin>>a2;

min= M[a2][0];

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

{

if(M[a2][j]<min)

{

min=M[a2][j];

}

}

cout<<"\nMin: "<<min;

}

break;

case 3:

{

FILE \*f1, \*f2;

f1=fopen("massiv1.txt", "w");

f2=fopen("max\_min.txt", "w");

for (int l=0; l<4; l++)

{

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

{

fprintf(f1," %.3f", M[l][k]);

}

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

}

fprintf(f2,"\n %.3f %.3f %d %d", max, min, a1, a2);

fclose(f1);

fclose(f2);

cout<<"\nПроизведена запись в файл.";

}

break;

default: return 0;

}

}

}

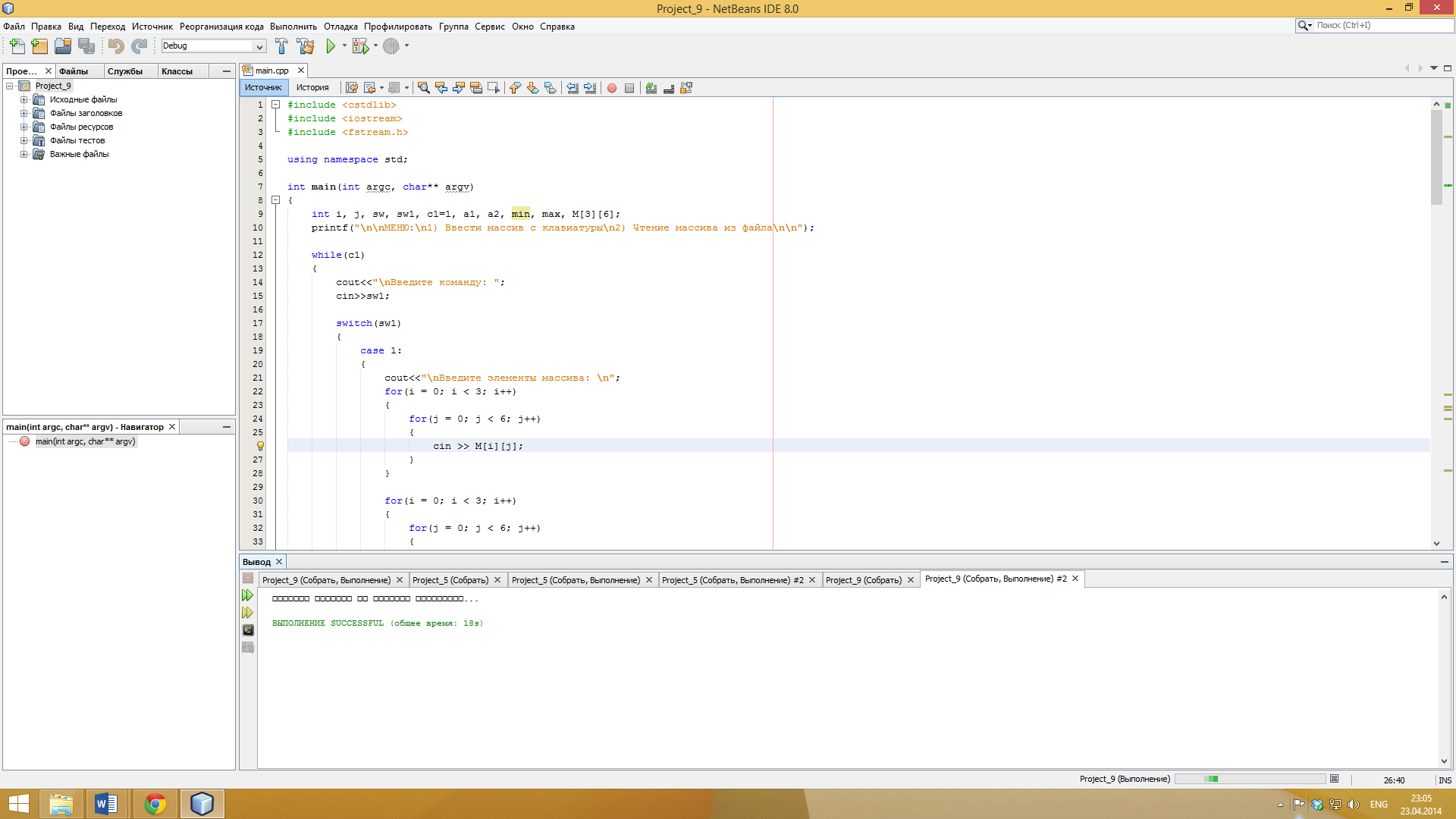
****

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 ищет в массиве максимальный элемент, по команде «2» - минимальный. Наконец, команда «3» сохраняет полученный массив и найденные элементы в файлы. Ввод каких-либо других чисел останавливает работу программы. Иными словами, для прекращения работы программы достаточно ввести какое-либо число кроме номеров команд.

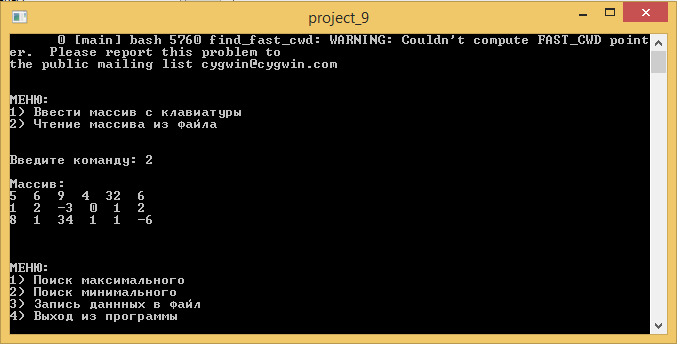


Рисунок 2 – Демонстрация работы программы

**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.