**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.

направление подготовки 230400

\_\_\_\_\_\_\_\_\_\_\_\_\_ Плиев В.

«\_\_\_»\_\_\_\_\_\_\_\_ 2015

Принял

ст.пр.каф.БИС

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

оценка

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

«\_\_\_»\_\_\_\_\_\_\_\_ 2015

Санкт-Петербург, 2015

**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

Формализация задачи ................................................................................................ 5

Блок-схема алгоритма ……………………………………………………………... 6

Текст программы …………………………………………………………………... 7

Руководство пользователя ……………………………………………………….. 10

Заключение ............................................................................................................... 12

**Постановка задачи**

Текст задания:

# Вариант № 19

Разработать блок-схему алгоритма программы и программу, которая:

а) вводит с клавиатуры или читает из файла двумерный массив целых чисел размерностью [3 × 4] элементов и выводит его на экран. Элементы массива в файле могут быть расположены произвольно;

б) многократно производит изменение матрицы путем замены указанного элемента на другой, вводимый с клавиатуры, и выводит измененную матрицу на экран;

в) по окончании работы сохраняет измененный массив в файле. Данные этого массива сохраняются в виде матрицы, размерностью [3 × 4] элементов с двумя пробелами между ними, а величины измененных (старых) элементов и значение их индексов [i,j] 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.

**Формализация задачи**

Для начала нужно подключить необходимые библиотечные модули и объявить глобальные переменные, а также определить параметры массива и дать им характеристику. Только после этого можно начинать проектировать «скелет» программы.

Предполагается что программа будет работать в два этапа, соответственно будет два последовательных меню, реализованных с помощью оператора «switch».

Первое меню будет включать в себя два пункта:

1. Ввод массива с клавиатуры
2. Чтение массива из файла

Второе меню уже три:

1. Изменение элемента массива
2. Запись данных в файл
3. Выход

**Блок-схема алгоритма**

Начало

Неправильная команда

while(1)

Меню

Ввод массива вручную

Чтение массива из файла

default

1

2

sw1

while(c1)

Меню

i, j, sw, sw1, c1=1, a1, a2

starv, stari, starj, M[3][4]

switch(sw1)

def

2

1

Конец

Запись данных в файл

Изменение элемента массива

**Текст программы**

#include <iostream>

#include <fstream>

using namespace std;

int main() {

int i, j, sw, sw1, c1 = 1, a1, a2, starv, stari, starj, M[3][4];

printf("\n\nMENU:\n1) Vvesti massiv s klaviatury\n2) Chtenie massiva iz faila\n\n");

while (c1) {

cout << "\nVvedite comandu: ";

cin>>sw1;

switch (sw1) {

case 1:

{

cout << "\nVvedite elementy massiva: \n";

for (i = 0; i < 3; i++) {

for (j = 0; j < 4; j++) {

cin >> M[i][j];

}

}

for (i = 0; i < 3; i++) {

for (j = 0; j < 4; j++) {

printf("%2d ", M[i][j]);

}

cout << endl;

}

c1 = 0;

}

break;

case 2:

{

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

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

for (j = 0; j < 4; j++) {

file >> M[i][j];

}

}

cout << "\nMassiv: \n";

for (i = 0; i < 3; i++) {

for (j = 0; j < 4; j++) {

printf("%2d ", M[i][j]);

}

cout << endl;

}

c1 = 0;

}

break;

default: printf("\nOshibka: nepravilny simvol.\n\n");

}

}

printf("\n\n\nMENU:\n1) Izmenenie elementa massiva\n2) Zapis dannyh v fayl\n3) Vyhod iz programmy");

while (1) {

cout << "\n\n\nVvedite comandu: ";

cin>>sw;

switch (sw) {

case 1:

{

cout << "\nVyberite stroku: \n";

cin>>a1;

cout << "\nVyberite stolbec: \n";

cin>>a2;

a1--;

a2--;

cout << "\nVvedite novoe znacenie elementa: \n";

cin >> M[a1][a2];

cout << "\nMassiv: \n";

for (i = 0; i < 3; i++) {

for (j = 0; j < 4; j++) {

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

}

cout << endl;

}

}

break;

case 2:

{

FILE \*f1, \*f2;

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

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

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

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

fprintf(f1, " %d", M[l][k]);

}

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

}

fprintf(f2, "\n %d %d %d", starv, stari, starj);

fclose(f1);

fclose(f2);

cout << "\nProizvedena zapis v fayl.";

}

break;

default: return 0;

}

}

}

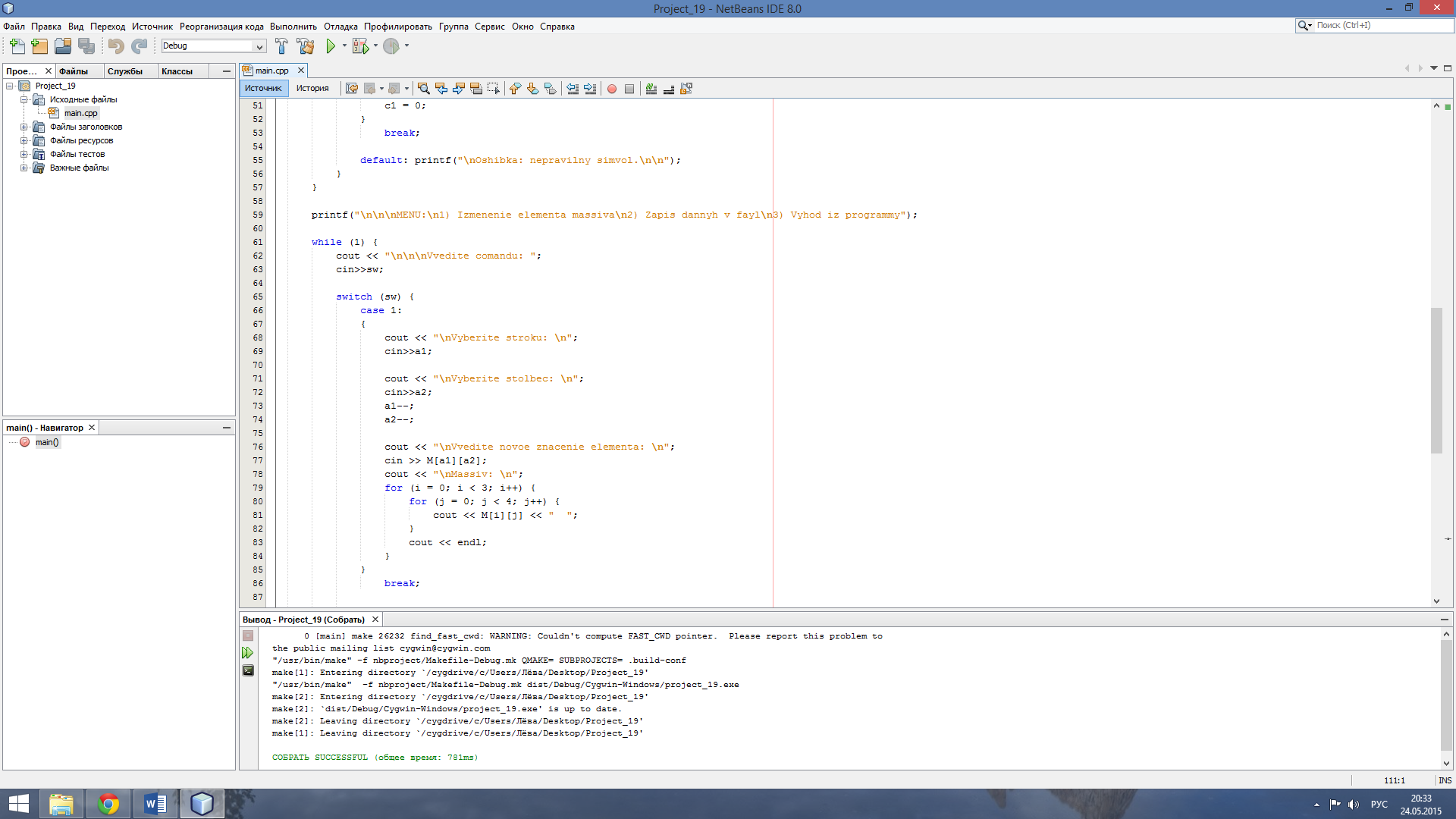
****

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. Команда «1» позволяет изменить любой элемент массива; после её запуска нужно будет выбрать по номерам строки и столбца желаемый элемент и после этого ввести его новое значение. Команда «2» сохраняет полученный массив и измененного элемента в файлы. Ввод каких-либо других чисел останавливает работу программы. Иными словами, для прекращения работы программы достаточно ввести какое-либо число кроме номеров команд.

**Идентификаторы и их назначение:**

M[3][4] – исходный массив

i, j, sw, sw1, c1=1, a1, a2 – целочисленные переменные

starv – старая величина измененного элемента массива

stari – номер строки измененного элемента массива

starj – номер столбца измененного элемннета массива



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.

Наиболее сложным разделом пояснительной записки стала разработка массива, работающего с вещественными числами, и реализация функции записи этого массива в файл.

**Список использованной литературы**

1. Керниган Б., Ритчи Д. Язык программирования Си. – СПб, 2012. – 352 с.