

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Уфимский государственный нефтяной технический университет»  
Институт цифровых систем, автоматизации и энергетики (ИТ-институт)

Кафедра вычислительной техники и инженерной кибернетики

## **СОЗДАНИЕ ДИНАМИЧЕСКОЙ СТРУКТУРЫ ДАННЫХ «ГРУППА»**

Пояснительная записка к курсовой работе по дисциплине

### **Программирование**

Вариант 3

Студент группы БПО18-01

\_\_\_\_\_ М.А. Ширшова

(Дата, подпись)

Руководитель, ст. преподаватель

\_\_\_\_\_ М.А. Салихова

(Дата, подпись)

	нед	балл	доп	итог
проект				
релиз				
теория				

Уфа 2021

## СОДЕРЖАНИЕ

Введение	2
1 Техническое задание	4
1.1 Содержательное описание	5
1.2 Назначение программы	5
1.3 Математическое описание задачи и функциональные требования к программе	6
1.4 Интерфейс ввода/вывода	8
1.5 Технические требования к программе	10
2 Описание алгоритма	
2.1 Расчетные формулы	11
2.2 Описание переменных	12
2.3 Схемы алгоритмов отдельных задач	14
2.4 Структура программы	25
3 Описание программы	
3.1 Код программы для консольного приложения	34
3.2 Код программы для оконного приложения	43
4 Анализ результатов	
4.1 Результаты программы для консольного приложения	59
4.2 Результаты программы для оконного приложения	66
Заключение	69
Список использованных источников	70

## ВВЕДЕНИЕ

Структура – это совокупность данных различного типа, которые лежат в непрерывной области памяти, и объединённые общим именем.

Использование структур очень удобно, если приходится иметь дело с большим объёмом данных. XXI век является веком информации, которой с каждым днём становится всё больше и больше. Поэтому базы данных стали чаще использоваться не только на работе, но и в обычной жизни. Как известно, структуры являются в какой-то степени базами данных. Другими словами структура – это тип данных, который может включать в себя несколько полей.

ВУЗ старается приобщить всех студентов к какой-либо внеучебной деятельности, потому что каждый специалист должен быть всесторонне развит. Этому способствуют не только изучение профилирующих предметов, что, разумеется, необходимо, но и изучение общих дисциплин, и посещение различных факультативов, студенческих клубов, театра и так далее.

Но при этом нельзя забывать про успеваемость. Люди приходят в ВУЗ в первую очередь за знаниями, а всё остальное – это приятное дополнение в студенческой жизни.

Мой программный продукт предназначен для того, чтобы можно было узнать обстановку в группе по итогам каждого семестра. Из этого вытекают цель и задачи моей работы.

Цель работы:

- изучение синтаксиса структур на языке C++;

Инструментом для достижения цели является MICROSOFT VISUAL STUDIO 2010 EXPRESS.

Следовательно, задачами работы будут:

- разработка основных алгоритмов;
- реализация программы.

Конечно, в каждом учебном заведении существуют базы, где записана информация о студенте. Но зачастую в ней прописаны только общие сведения, такие как ФИО и успеваемость.

В моём же варианте записаны данные о том, какой медицинской группе принадлежит студент и вид той учебной деятельности, которой он занимается.

Таким образом, можно сделать вывод, что реализованный программный продукт имеет некие преимущества по сравнению с аналогами.

# 1 Техническое задание

## Задание на курсовое проектирование

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Уфимский государственный нефтяной технический университет»  
Институт цифровых систем, автоматизации и энергетики (ИГ-институт)

---

### ЗАДАНИЕ на курсовое проектирование по дисциплине “Программирование”

---

студенту Ширшовой Марии Андреевне

Тема работы: «Создание динамической структуры «Группа»», Вариант 3

1. Цель:

1.1 изучение синтаксиса структур на языке C++;

1.2 реализовать программный продукт, удобный в использовании в учебных заведениях.

2. Задачи:

2.1 Создать структуру «Группа», включающую не менее 3 полей;

2.2 Создать сущности «БПО-18-01» и «БПО-18-02» описанной структуры;

2.3 Создать структуру «Студент», содержащую не менее 5 полей, одно из которых называется «группа» и указывает, в какой группе учится студент;

2.4 Создать динамические массивы студентов;

2.5 Написать функции для ввода значений с клавиатуры, из файла и вывода на экран;

2.6 Написать 3 функции обработки данных о студентах;

2.7 Написать функции для вывода сведений о вузе, которому принадлежит факультет;

2.8 Разработать форму в приложении с графическим интерфейсом пользователя.

3. Технические условия:

3.1. Язык программирования C++,

инструментальная среда MICROSOFT VISUAL STUDIO 2010 EXPRESS

3.2. Системные требования:

3.2.1 1024 МБ ОЗУ (1,5 Гб при выполнении на виртуальной машине);

3.2.2 процессор с тактовой частотой 1,6 ГГц или большей;

3.2.3 5,5 Гб доступного пространства на жёстком диске;

3.2.4 Windows Server 2008 и 2003 R2, Windows 7, Windows Vista, Windows XP, Windows 10

4. Стадии и этапы разработки:

4.1. Уточнение и утверждение технического задания 7 неделя

4.2. Разработка алгоритмов 8 неделя

4.3. Разработка программы 8 неделя

4.4. Отладка программы и оформление пояснительной записки 9 неделя

4.5. Защита курсовой работы 11 неделя

Руководитель проекта,

ст. преподаватель

\_\_\_\_\_  
(Подпись)

М.А. Салихова

Исполнитель проекта,

студент

\_\_\_\_\_  
(Подпись)

М.А. Ширшова

Дата выдачи “16” сентября 2021 г.

## **1.1 Содержательное описание**

Для достижения цели и реализации задач необходимо создать структуру «Группа», включающую не менее 3 полей.

Для нее создать сущности «БПО-18-01» и «БПО-18-02». Сущности – это объекты, которые будут иметь тип данных имени структуры, к которой относятся.

Далее создаем структуру «Студенты», содержащую не менее 5 полей.

Создадим текстовые файлы \*.txt, в которых будет храниться информация о каждой группе (сколько групп, столько файлов).

В главной программе создадим динамические массивы студентов, а затем по желанию пользователя вызывать ту или иную функцию с помощью разработанного меню.

## **1.2 Назначение программы**

Созданный программный продукт – программное средство, предназначенное для создания и изменения сведений о студентах. Оно предназначено для преподавателей, которым необходима информация о той или иной группе, или о том или ином студенте.

### 1.3 Математическое описание задачи и функциональные требования к программе

Для реализации консольного приложения необходимо создать структуры «Группа» и «Студент» со следующими полями:

Таблица 1 – Поля структуры «Группа» («Group»)

Тип данных	Обозначение в программе	Толкование
char*	specialty	Название группы
int	quantity	Количество человек в группе
char*	headman	Фамилия старосты
char*	curator	Фамилия куратора
char*	faculty	Факультет, которому принадлежит группа

Таблица 2 – Поля структуры «Студент» («Student»)

Тип данных	Обозначение в программе	Толкование
Group	group	Группа, которой принадлежит студент
char*	name2	Фамилия студента
char*	name1	Имя студента
float	ball	Средний балл
char*	love_subject	Любимый предмет студента
char*	activities	Внеучебная деятельность
int	medical	Медицинская группа
bool	deleted	Признак отчисления студента

Для реализации оконного приложения необходимо создать классы «Группа» и «Студент» со следующими полями:

Таблица 3 – Поля класса «Группа» («Group»)

Тип данных	Обозначение в программе	Толкование
System::String^	specialty	Название группы
int	quantity	Количество человек в группе
System::String^	headman	Фамилия старосты
System::String^	curator	Фамилия куратора
System::String^	faculty	Факультет, которому принадлежит группа

Таблица 4 – Поля класса «Студент» («Student»)

Тип данных	Обозначение в программе	Толкование
------------	-------------------------	------------

Group	group	Группа, которой принадлежит студент
System::String^	name2	Фамилия студента
System::String^	name1	Имя студента
double	ball	Средний балл
System::String^	love_subject	Любимый предмет студента
System::String^	activities	Внеучебная деятельность
int	medical	Медицинская группа
bool	deleted	Признак отчисления студента

Данный выбор полей мне кажется наиболее рациональным из-за того, что:

1) Группа:

- каждая группа обязательно принадлежит какому-либо факультету в пределах данного университета;
- у каждой группы свое название, отличающееся от всех остальных;
- в каждой группе определённое количество человек;
- у каждой группы обязательно есть староста и куратор.

2) Студент:

- студент – это человек, а у каждого человека есть фамилия и имя;
- каждый студент принадлежит какой-то определенной группе, в которой он обучается;
- у каждого студента есть увлечения, которыми он может заняться во внеучебное время;
- каждый человек имеет право на свое мнение – любить или нет ему тот или иной предмет;
- после прохождения медицинского осмотра, студента определяют в определенную медицинскую группу;
- если студента отчислили его надо удалить из базы.

По заданию необходимо написать как минимум 3 функции обработки данных студентов. Анализируя выбранные поля, можно создать ряд функций, таких как:



- у скольких студентов балл выше среднего балла группы;
- сколько студентов занимаются тем или иным видом внеучебной деятельности;
- скольким студентам нравится тот или иной предмет;
- у кого самая длинная фамилия.

Для реализации вышеперечисленные функции, необходимо:

- 1) рассчитать средний балл группы;
- 2) уметь работать с функциями строк:
  - посимвольное сравнение строк;
  - сравнение длин строк;
- 3) реализовать счетчик событий, удовлетворяющих условию;
- 4) предусмотреть удаление и изменения.

#### **1.4 Интерфейс ввода/вывода**

Для консольного приложения ввод данных будет осуществляться по желанию пользователя – либо с клавиатуры, либо из заранее подготовленных файлов:

- 1cppstudio.txt, где хранятся данные о группе БПО-18-01;
- 2cppstudio.txt, где хранятся данные о группе БПО-18-02;
- 3cppstudio.txt, где хранятся данные о группе БПОи-18-01.

Для того чтобы пользователю было удобно работать с данным программным продуктом, прописаны специальные подсказки – в какой строке что следует ввести. Если же будут введены некорректные данные, то программа выдаст ошибку.

После вывода сведений о студентах экран очищается.

Для удобства пользователя было создано меню. Нажимая на одну из перечисленных кнопок, он вызовет определенную функцию. После ее выполнения, экран очищается, и меню снова всплывет. И так будет до тех пор, пока пользователь не решит выйти, нажав на соответствующую кнопку, приведенную в меню.

С интерфейсом меню можно ознакомиться в таблице 5.

Таблица 5 – Интерфейс меню

1-	Студенты из БПО-18-01, чей средний балл выше среднего балла группы
2-	Сколько студентов из БПОи-18-01 занимаются спортивным видом внеурочной деятельности
3-	Сколько студентов из БПО-18-02 любят физику
4-	Студент с самой длинной фамилией
5-	Вывод сведений о ВУЗе
6-	Вывод сведений о группе, в которой учится студент
7-	Изменить
8-	Удалить
9-	Вывод списка группы БПО-18-01
10-	Вывод списка группы БПО-18-02
11-	Вывод списка группы БПОи-18-01
0-	Выход

Для оконного приложения ввод данных будет осуществляться по желанию пользователя – либо с клавиатуры в специальные поля, либо из заранее приготовленного файла

– `crpstudio.txt`, где хранятся все данные о БПО-18;

Создадим 2 вкладки – Студент и Список. Переходя между ними, пользователь сможет вводить данные, просмотреть их вывод в таблице и вызвать некоторые функции, нажав на соответствующие кнопки.

В первой вкладке заполняются все сведения о студенте и его группе, а также представляются результаты функций (добавление с клавиатуры, из файла, поиск, и функции, связанные со всем потоком).

Во второй вкладке данные о студенте выводятся в таблицу, здесь же их можно удалить и изменить.

## 1.5 Технические требования к программе

Минимальные системные требования – набор условий, необходимых для возможности запуска и работы программного продукта. Рекомендуемые системные требования – набор характеристик, подразумевающих оптимальную работу большей части возможностей продукта.

### 1) Требования к функциональным характеристикам.

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

- добавлять, удалять, изменять, выводить характеристики о студентах;
- выводить информацию о группах;
- считывать данные из файла и с клавиатуры;
- осуществлять поиск фамилий студентов;
- рассчитывать среднее значение;
- сравнивать строки с заданной константой.

### 2) Условия эксплуатации.

Для корректной работы программы пользователю необходимо вводить верные значения (соответственно прописанным подсказкам).

- 3) Требования к оборудованию:
- 1024 МБ ОЗУ (1,5 ГБ при выполнении на виртуальной машине);
  - процессор с тактовой частотой 1,6 ГГц или большей;
  - разрешение экрана 1024 × 768 и выше;
  - 5,5 Гб доступного пространства на жёстком диске;
  - поддерживаемые ОС: Windows Server 2008 и 2003 R2, Windows 7, Windows Vista (кроме Starter Edition), Windows XP, Windows 10;
  - поддерживаемые архитектуры 32-разрядные и 64- разрядные.

## 2 Описание алгоритма

### 2.1 Расчетные формулы

Поиск среднего значения:

$$\text{ср.балл} = \frac{\text{балл}_1 + \text{балл}_2 + \dots + \text{балл}_N}{N} . \quad (1)$$

Поиск количества элементов, удовлетворяющих условию равенства:

$$\begin{aligned} k &= 0; \\ x = y - &> k = k + 1 . \end{aligned} \quad (2)$$

Поиск максимального значения:

$$\begin{aligned} \text{max} &= a[1]; \\ \text{max} < a[2] - &> \text{max} = a[2] . \end{aligned} \quad (3)$$

Формула (1) используется в функции, выводящей фамилии и имена студентов из группы БПО-18-01 имеют средний балл выше среднего балла группы.

Формула (2) используется в функциях, определяющих количество студентов предпочитающих физику в БПО-18-02 и спортивных вид внеучебной деятельности в БПОи-18-01. Также в этих функциях выводятся фамилии и имена этих студентов.

Формула (3) используется в функции поиска самой длинной фамилии в группе. Затем сравнением получившихся значений определяется максимальная длина фамилии на всем потоке.

## 2.2 Описание переменных

С переменными для консольного приложения можно ознакомиться в таблице 6 и таблице 7.

Таблица 6 – Имена переменных и их типы для консольного приложения

Смысл переменных	Обозначение		Тип переменной	Примечания
	в алгоритме	в программе		
<u>ИСХОДНЫЕ</u>	G1	G1	Group	В объявлении функции формальные параметры а, N, X, фактическими переменными которых являются а1, N1, G1 а2, N2, G2 а3, N3, G3
	G2	G2		
	G3	G3		
	a1	a1	Student*	
	a2	a2		
	a3	a3		
	N1	N1	int	
	N2	N2		
	N3	N3		
<u>ПРОМЕЖУТОЧНЫЕ</u>	Otvet	Otvet	int	
	i	i		
	num	num		
	ans	ans	char*	
	field	field		
	tmp	tmp		
	temp	temp	float	
	O	O		
	sr	sr		
<u>РЕЗУЛЬТАТЫ</u>	max1	max1	char*	Результат функции max записывается в max1, max2, max3
	max2	max2		
	max3	max3		
	max k	max k	int	

Таблица 7 – Имена переменных и толкование для консольного приложения

Обозначение	Толкование
G1	Сущность БПО-18-01
G2	Сущность БПО-18-02
G3	Сущность БПОи-18-01
a1	Массив студентов БПО-18-01
a2	Массив студентов БПО-18-02
a3	Массив студентов БПОи-18-01
N1	Количество человек в БПО-18-01
N2	Количество человек в БПО-18-02

N3	Количество человек в БПОи-18-01
Otvet	Выбор пункта из меню
i	Счетчик цикла
num	Номер удаляемой или изменяемой структуры
ans	Переменная для проверки изменить все поля или только некоторые
field	Изменяемое поле
tmp	Переменная для сравнения введенной фамилии с фамилиями из базы
temp	Переменная для перекодирования из 866 в 1251
O	Переменная для проверки какой файл открыть для чтения
sr	Средний балл группы БПО-18-01
max1	Самая длинная фамилия в БПО-18-01
max2	Самая длинная фамилия в БПО-18-02
max3	Самая длинная фамилия в БПОи-18-01
max	Самая длинная фамилия в функции
k	Счетчик количества студентов, удовлетворяющих условию

С переменными для оконного приложения можно ознакомиться в таблице 8 и таблице 9.

Таблица 8 – Имена переменных и их типы для оконного приложения

Смысл переменных	Обозначение	Тип переменной	Примечания
	в программе		
<u>ИСХОДНЫЕ</u>	GroupList StudentList StudentCount str	Group^ Student^ int String^	
<u>ПРОМЕЖУТОЧНЫЕ</u>	filename tsrt tmp count i x flag	String^   int  bool	
<u>РЕЗУЛЬТАТЫ</u>	sred min k	double String^ int	

Таблица 9 – Имена переменных и толкование для оконного приложения

Обозначение	Толкование
GroupList	Массив сущностей
StudentList	Массив студентов
StudentCount	Количество студентов
fileName	Указатель на файл
str	Считываемая строка из файла

tstr	Строка, считанная из файла, разделенная на слова
count	Счетчик строк, выводимых в таблице
tmp	Слово, с которым будут сравниваться элементы массива
flag	Показатель есть ли совпадения с tmp
i	Счетчик цикла
x	Счетчик цикла
sred	Средний балл
min	Самая короткая фамилия
k	Счетчик количества студентов, удовлетворяющих условию

## 2.3 Схемы алгоритмов отдельных задач

Блок -схемы функций и процедур представлены на рисунках 1-10.

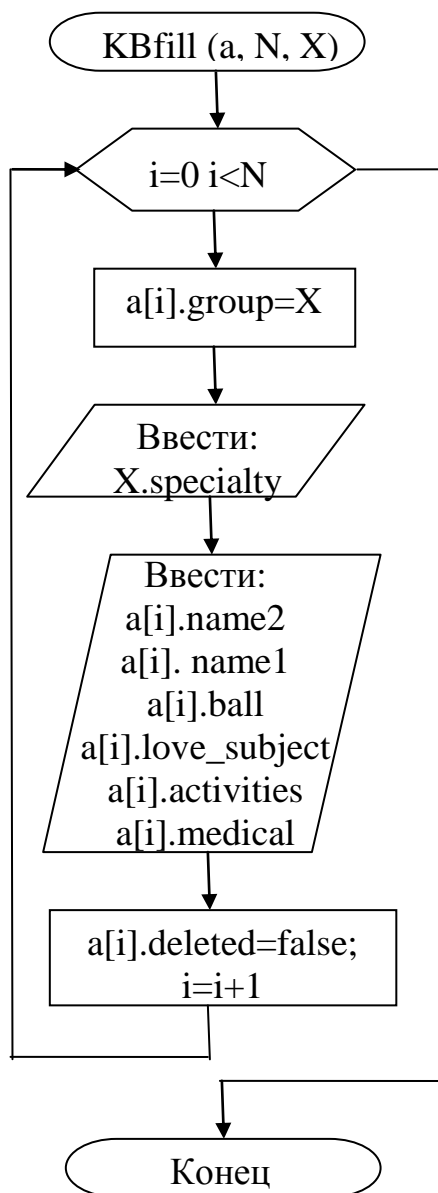


Рисунок 1 – Блок-схема алгоритма заполнения структуры с клавиатуры



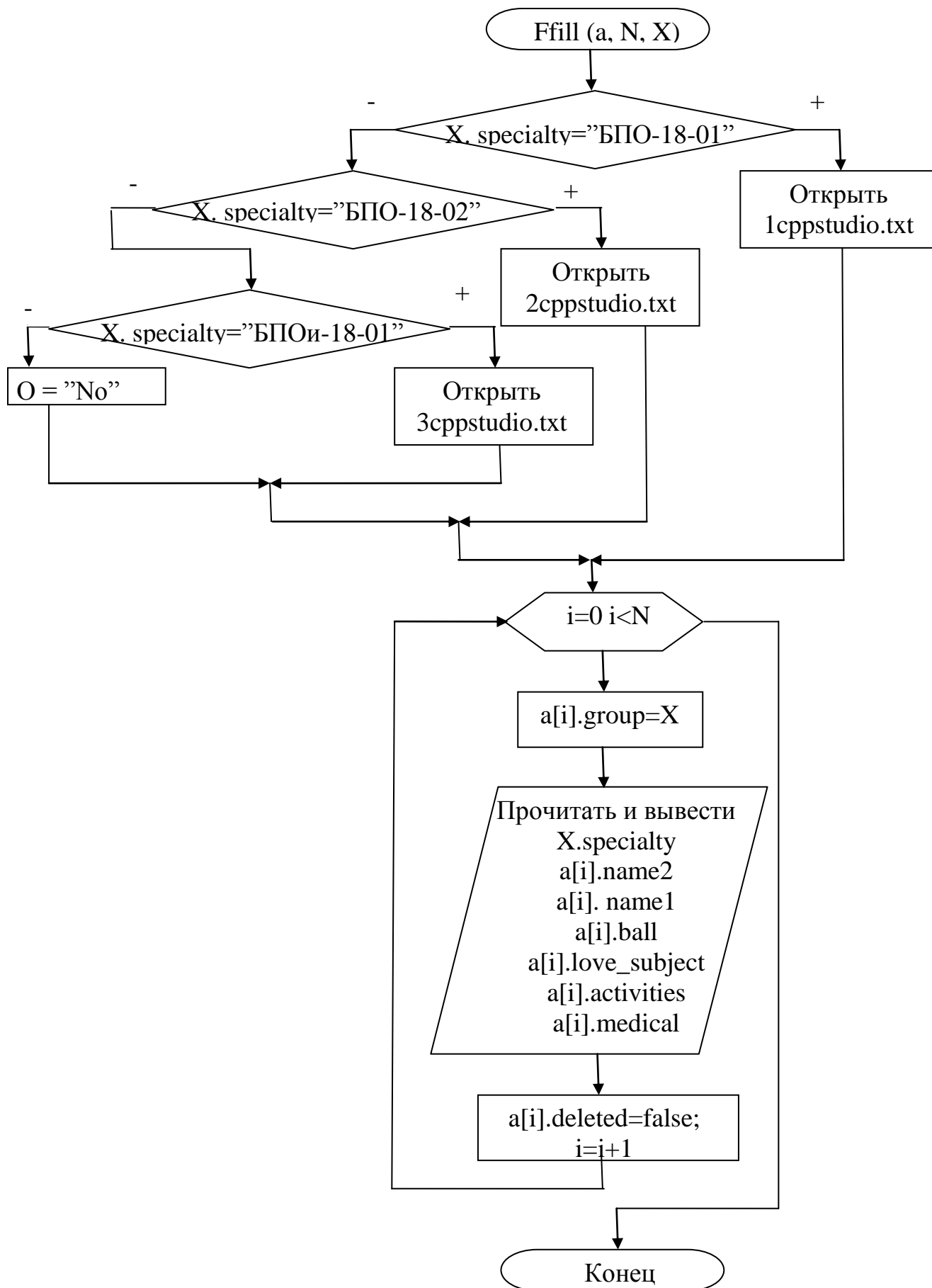


Рисунок 2 – Блок-схема алгоритма заполнения структуры из файла

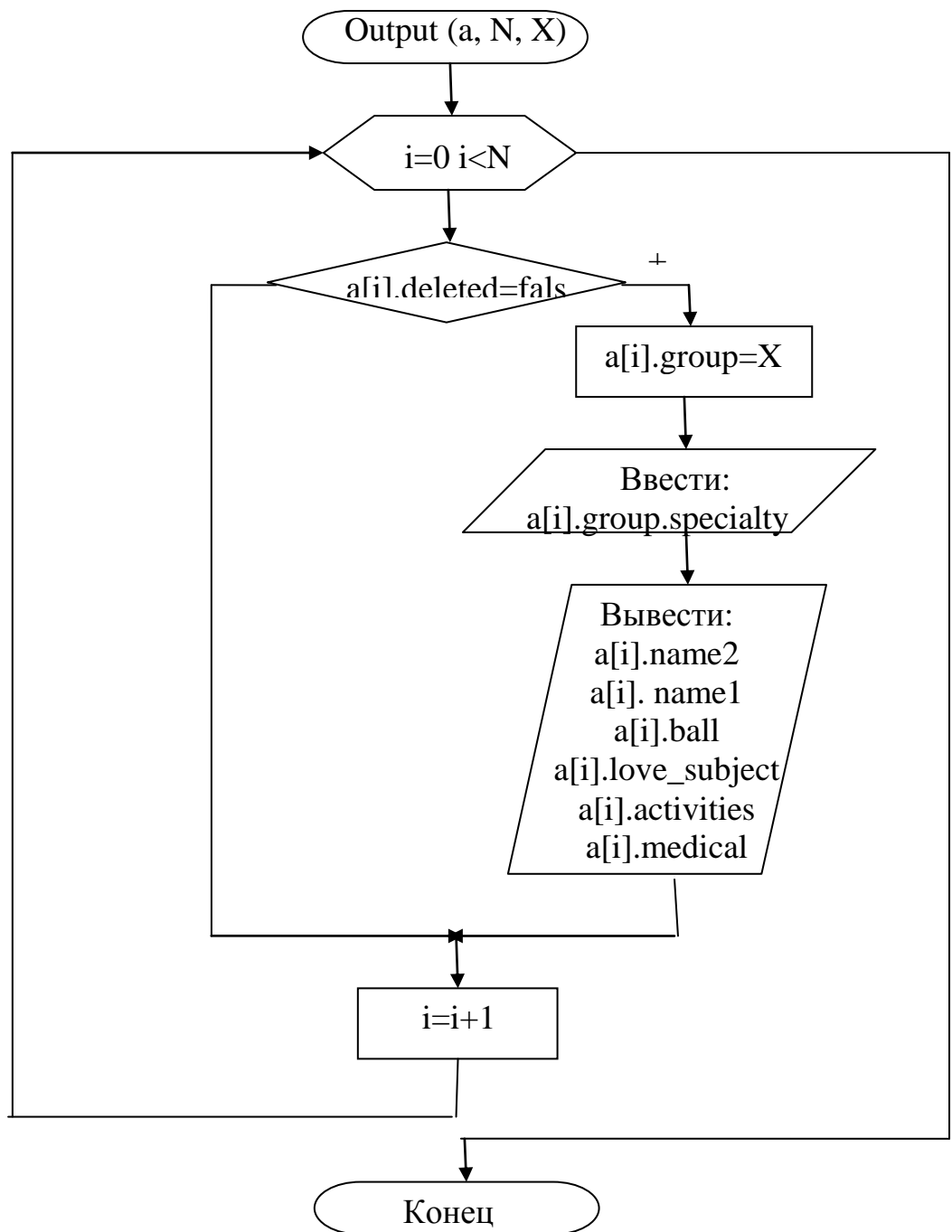


Рисунок 3 – Блок-схема алгоритма вывода на экран

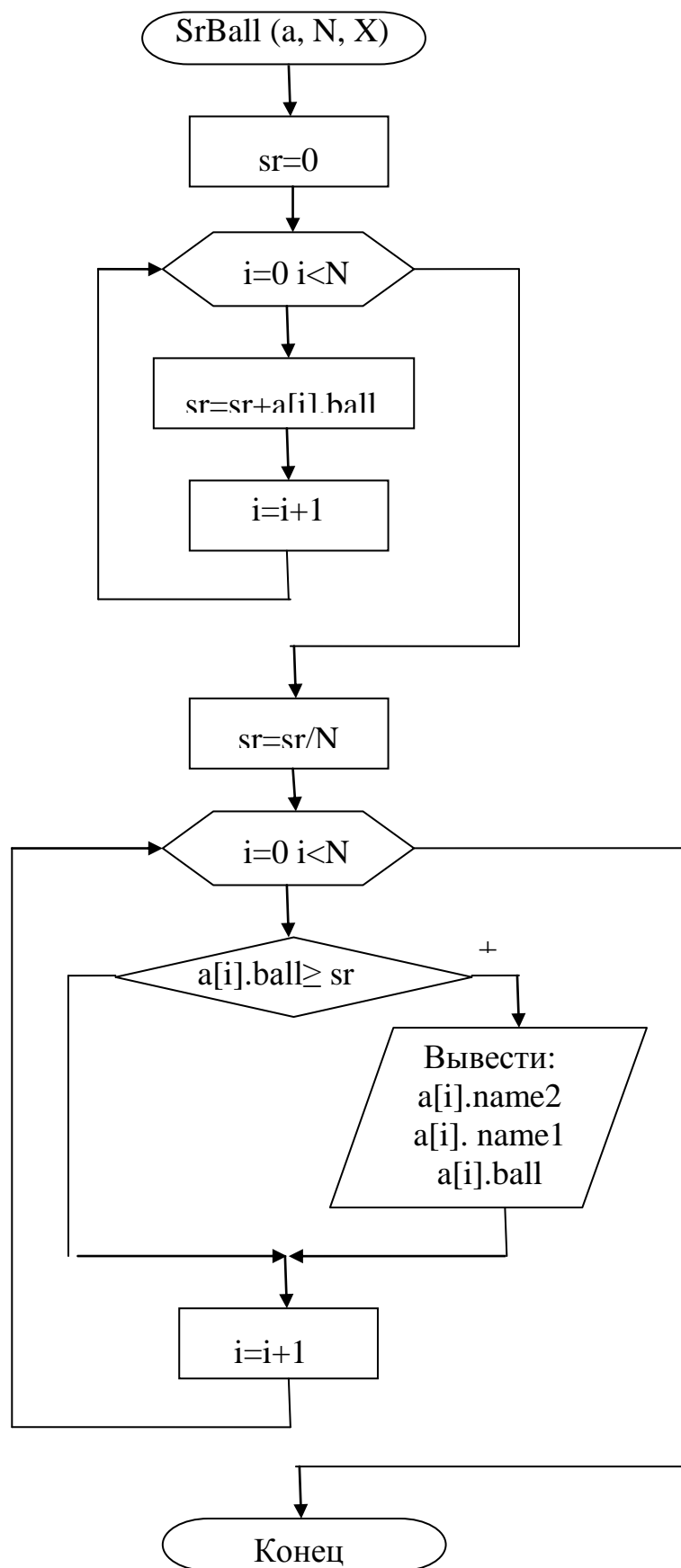


Рисунок 4 – Блок-схема алгоритма вычисления среднего балла группы

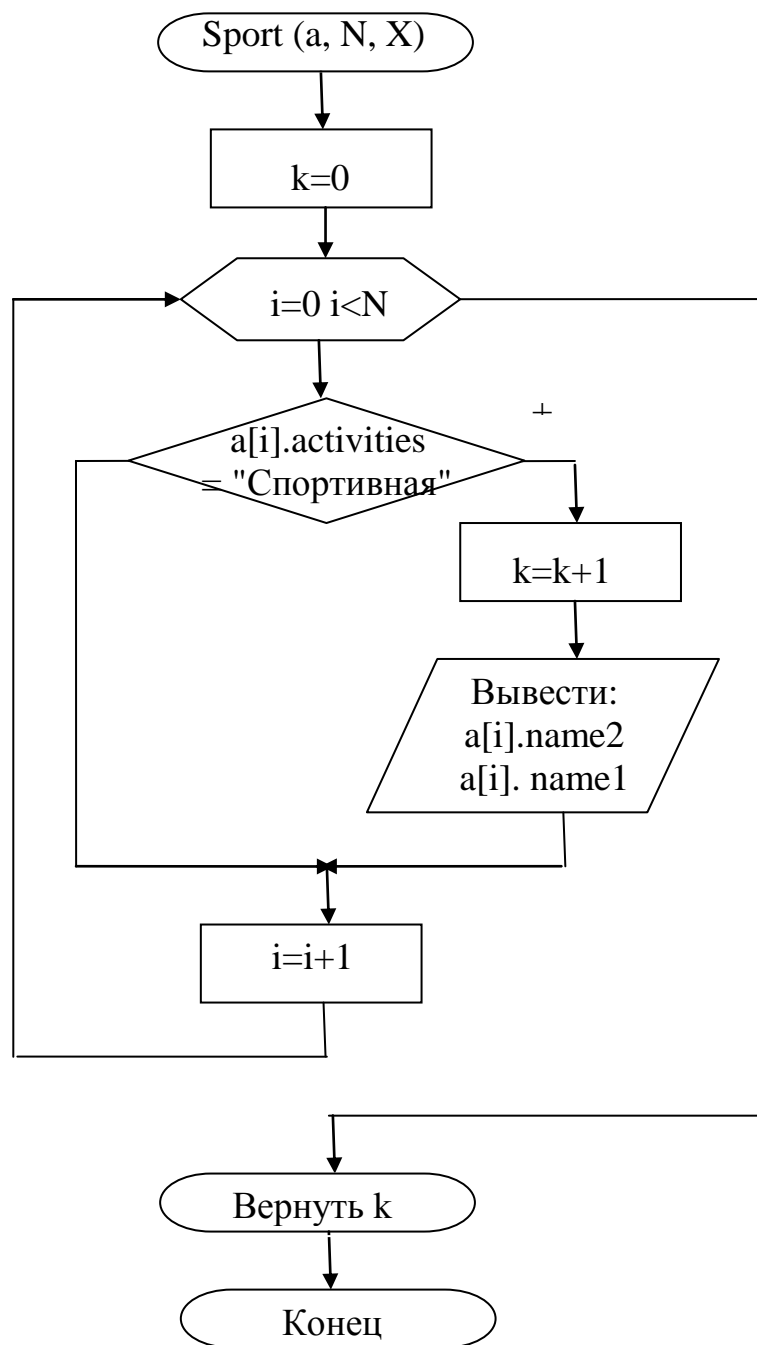


Рисунок 5 – Блок-схема алгоритма определения количества студентов, которые занимаются спортом

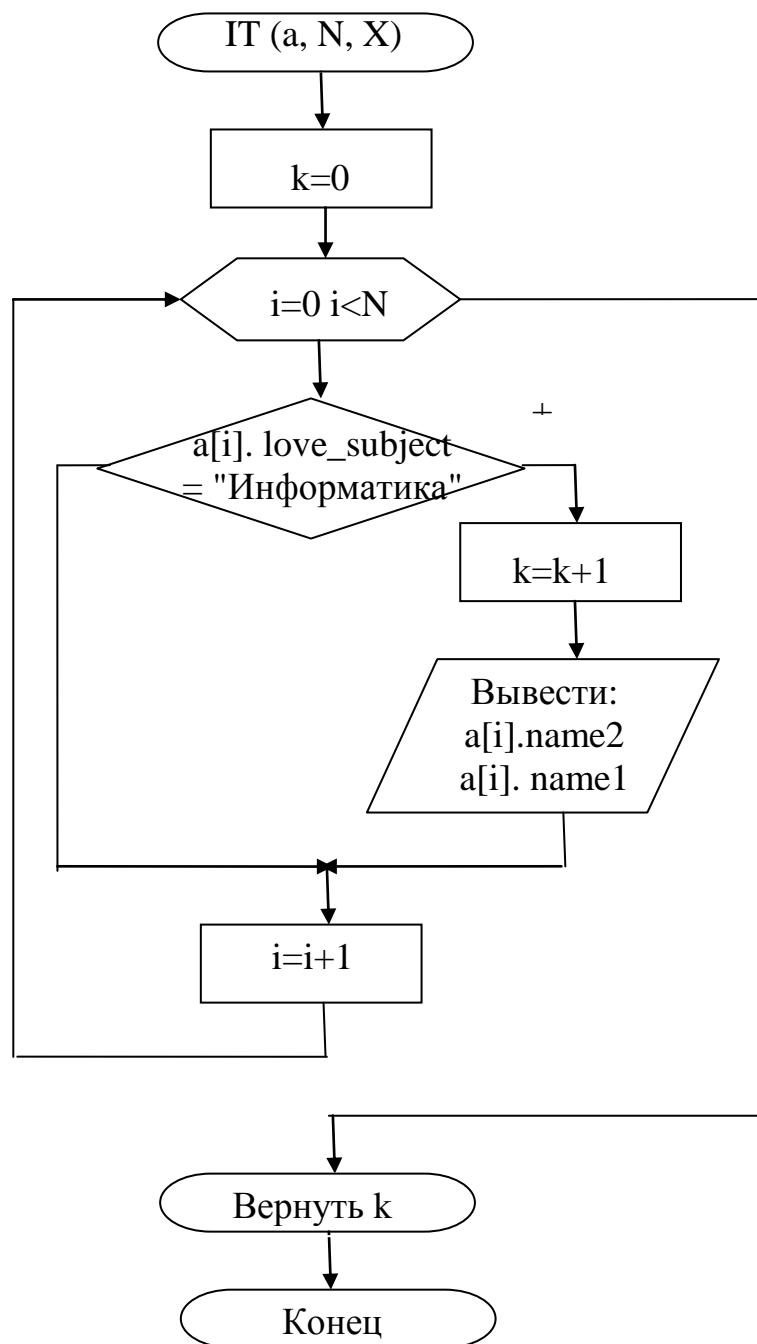


Рисунок 6 – Блок-схема алгоритма определения количества студентов, которые любят информатику

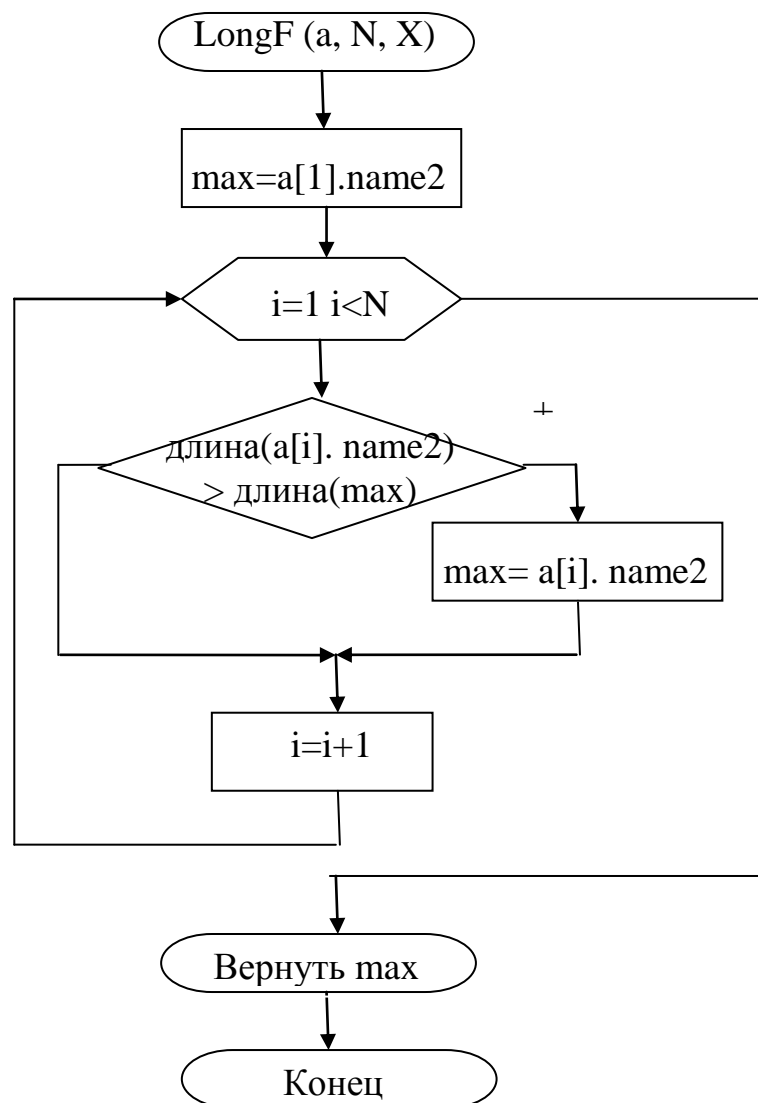
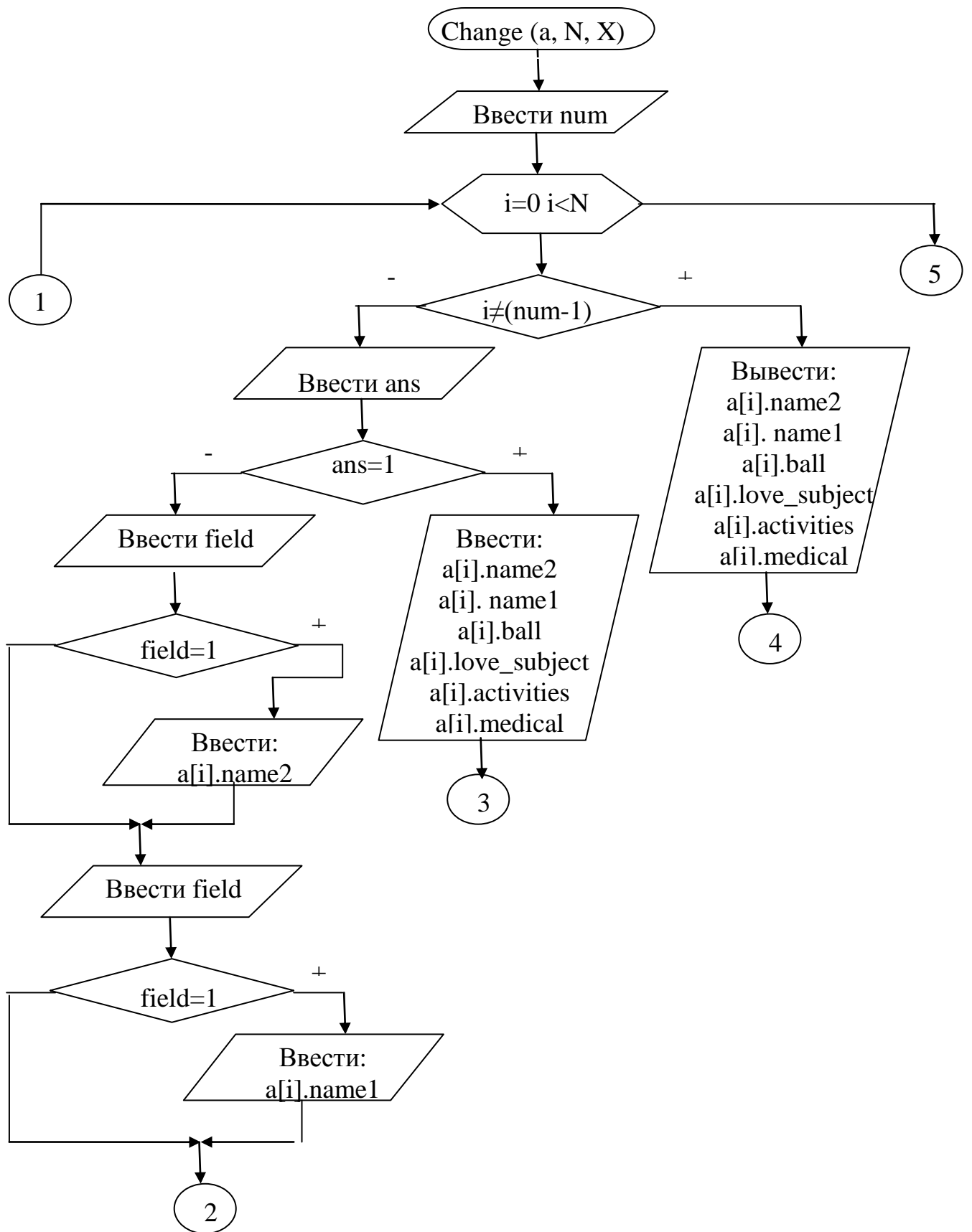
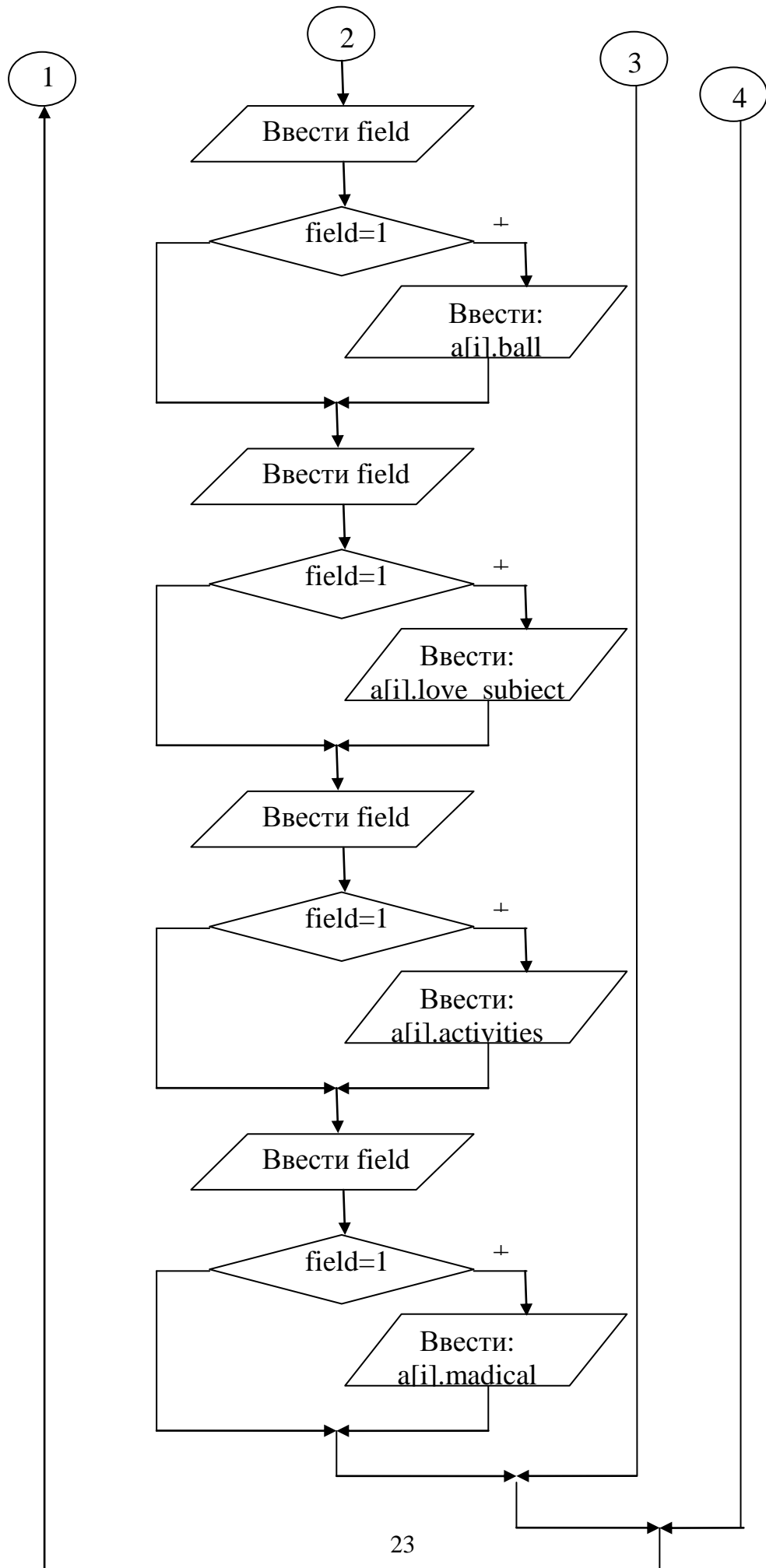


Рисунок 7 – Блок-схема алгоритма определения самой длинной фамилии







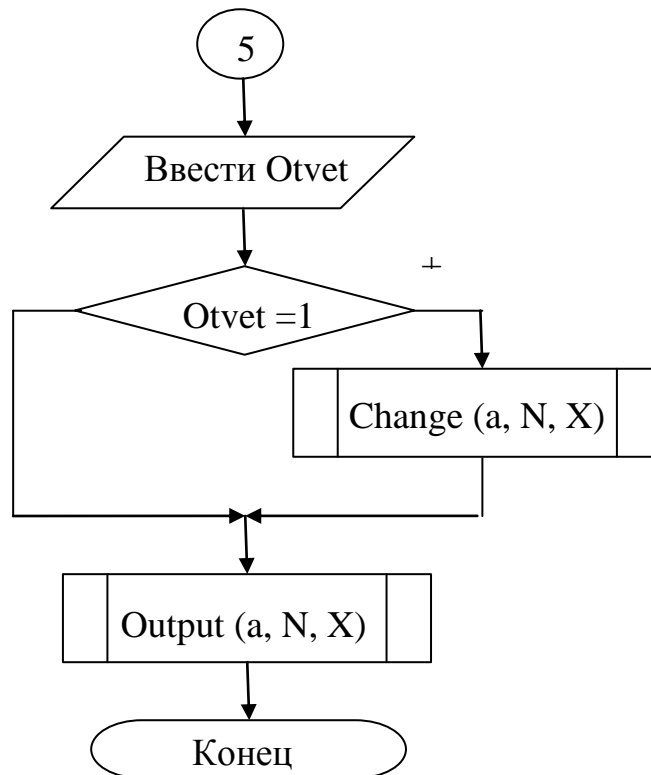


Рисунок 8 – Блок-схема алгоритма изменения структуры

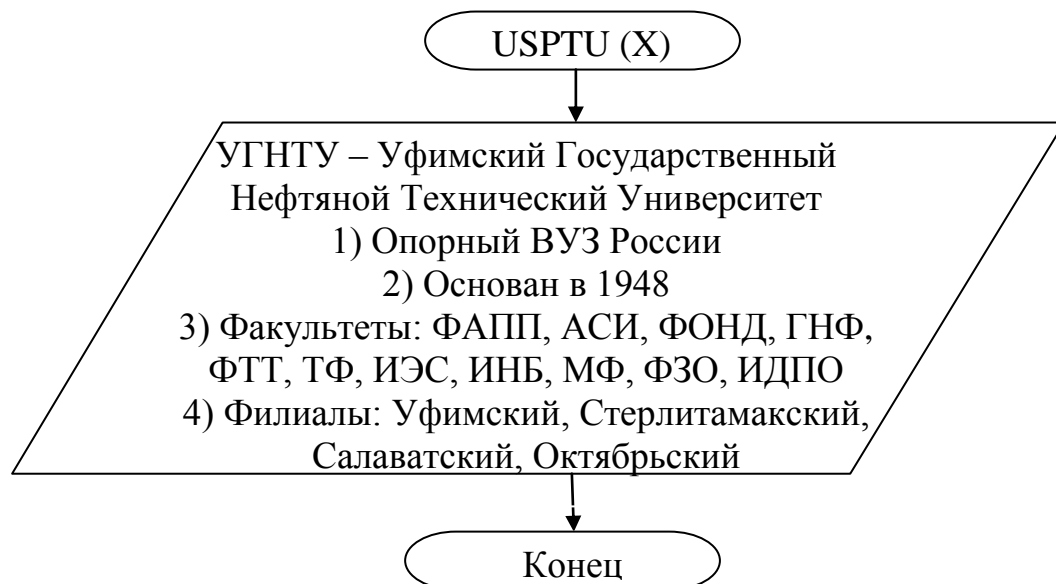


Рисунок 9 – Блок-схема алгоритма вывода сведений о ВУЗе

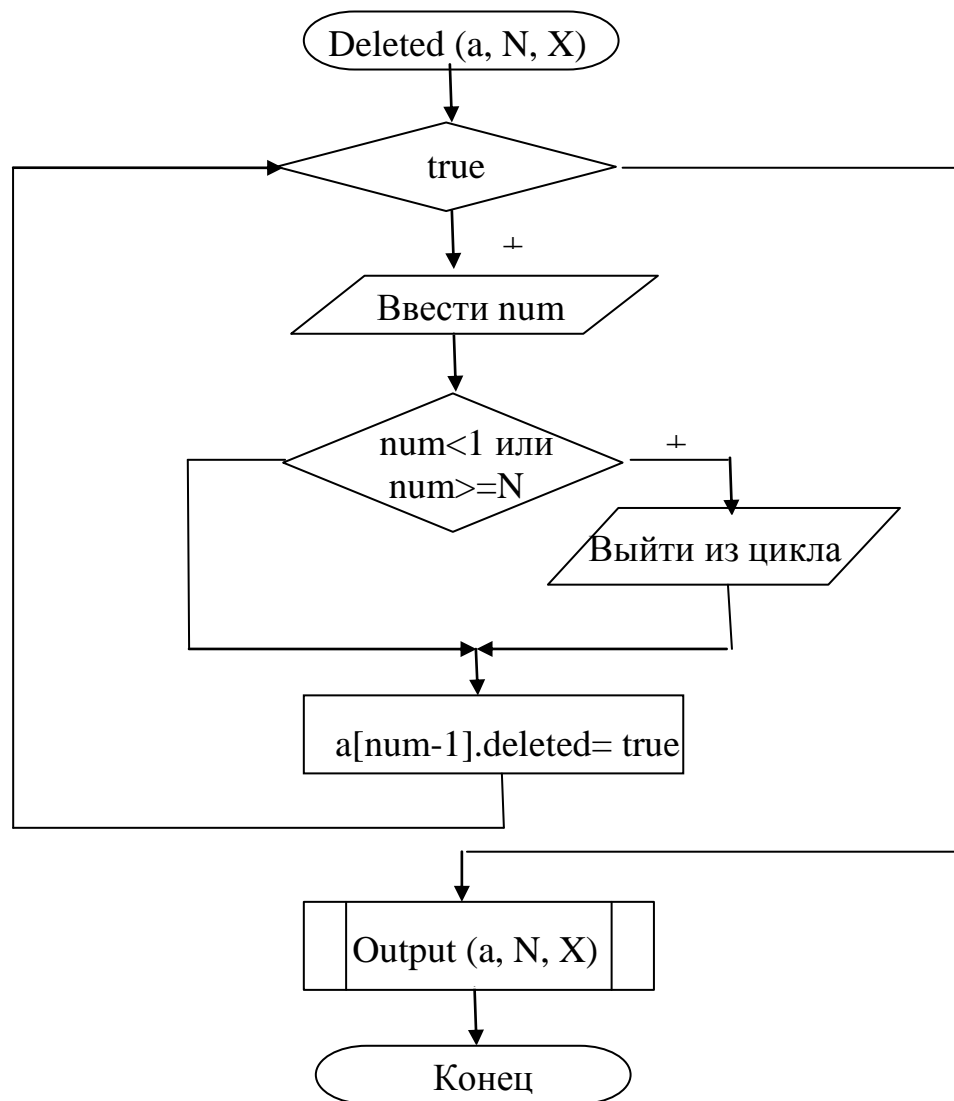
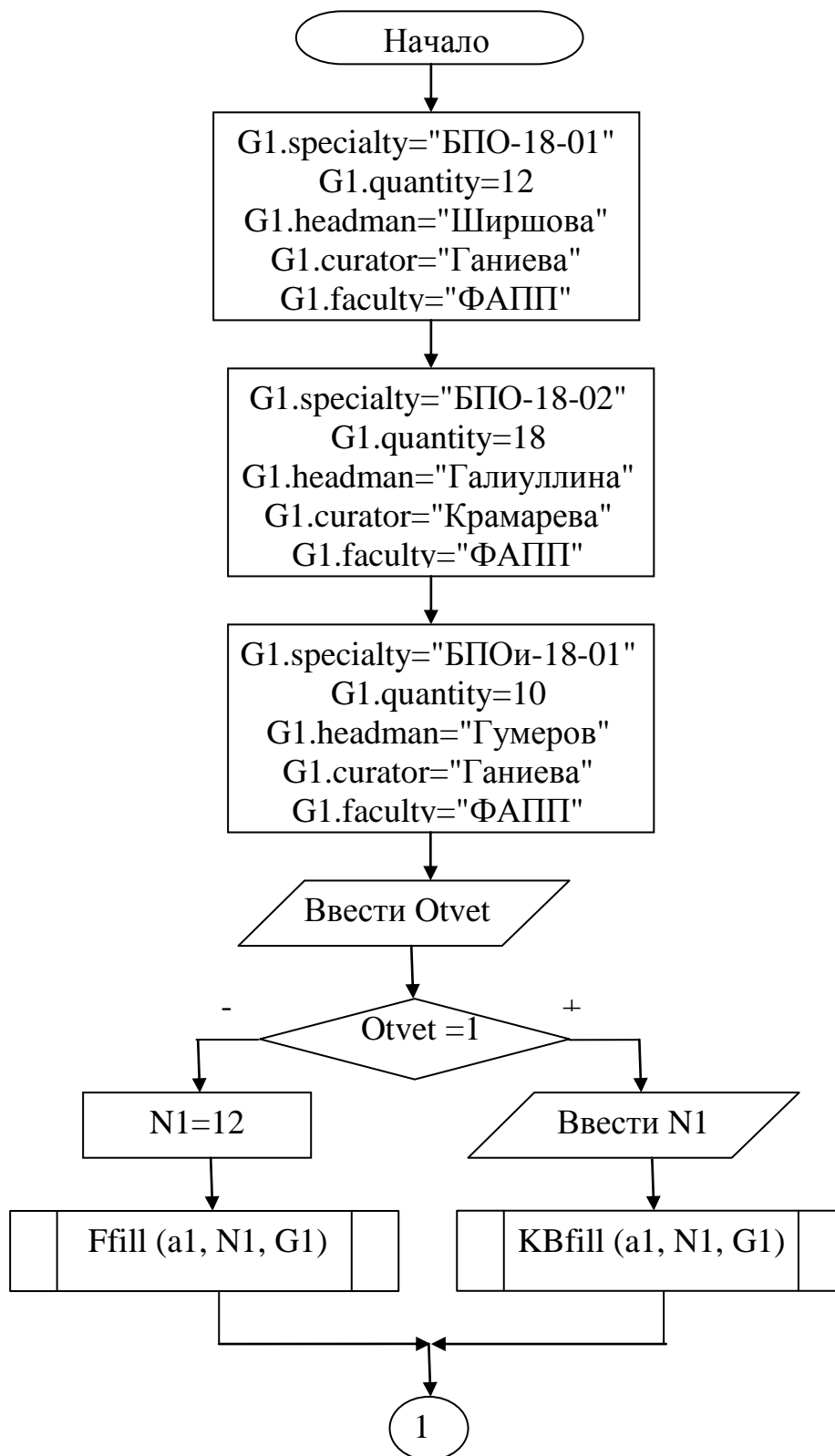
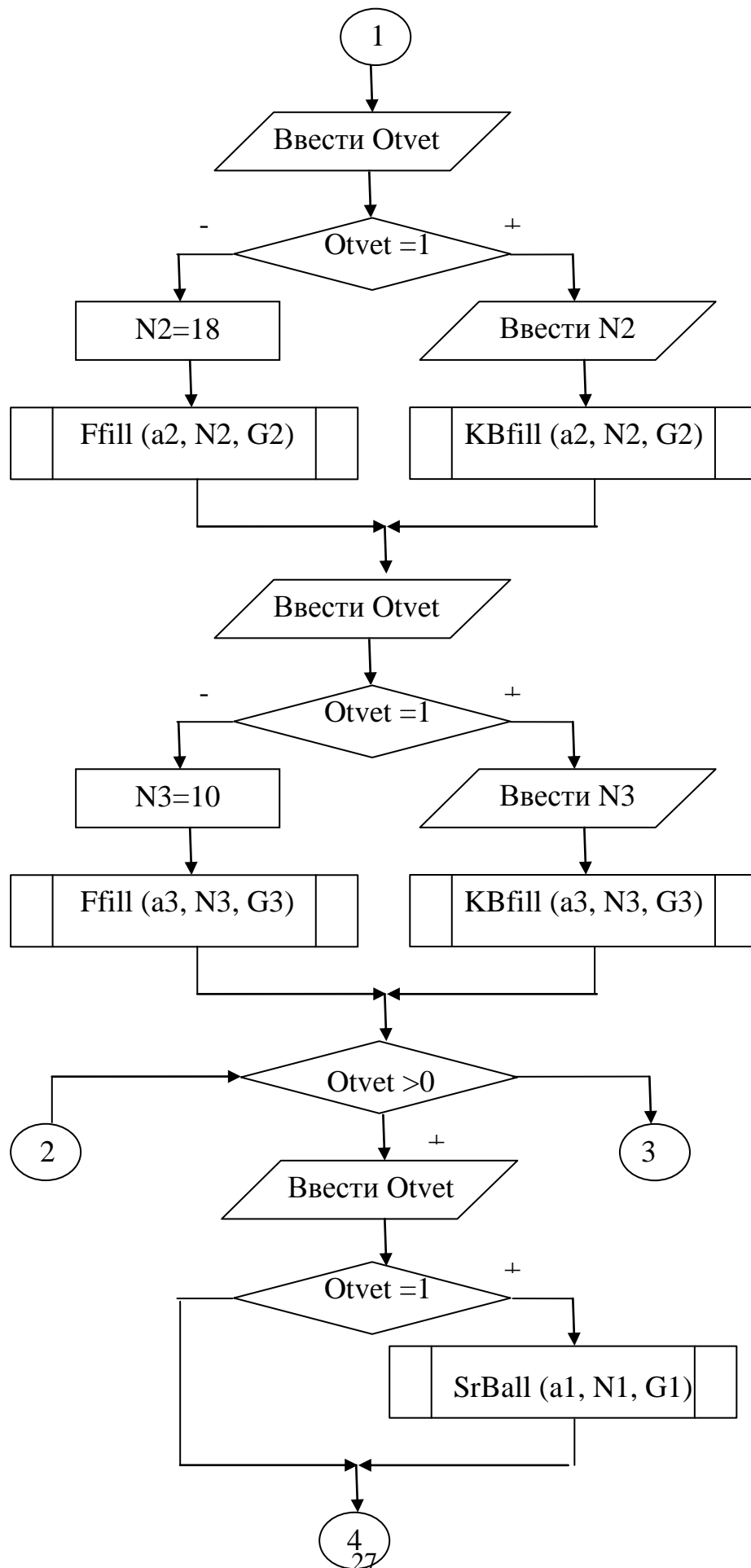


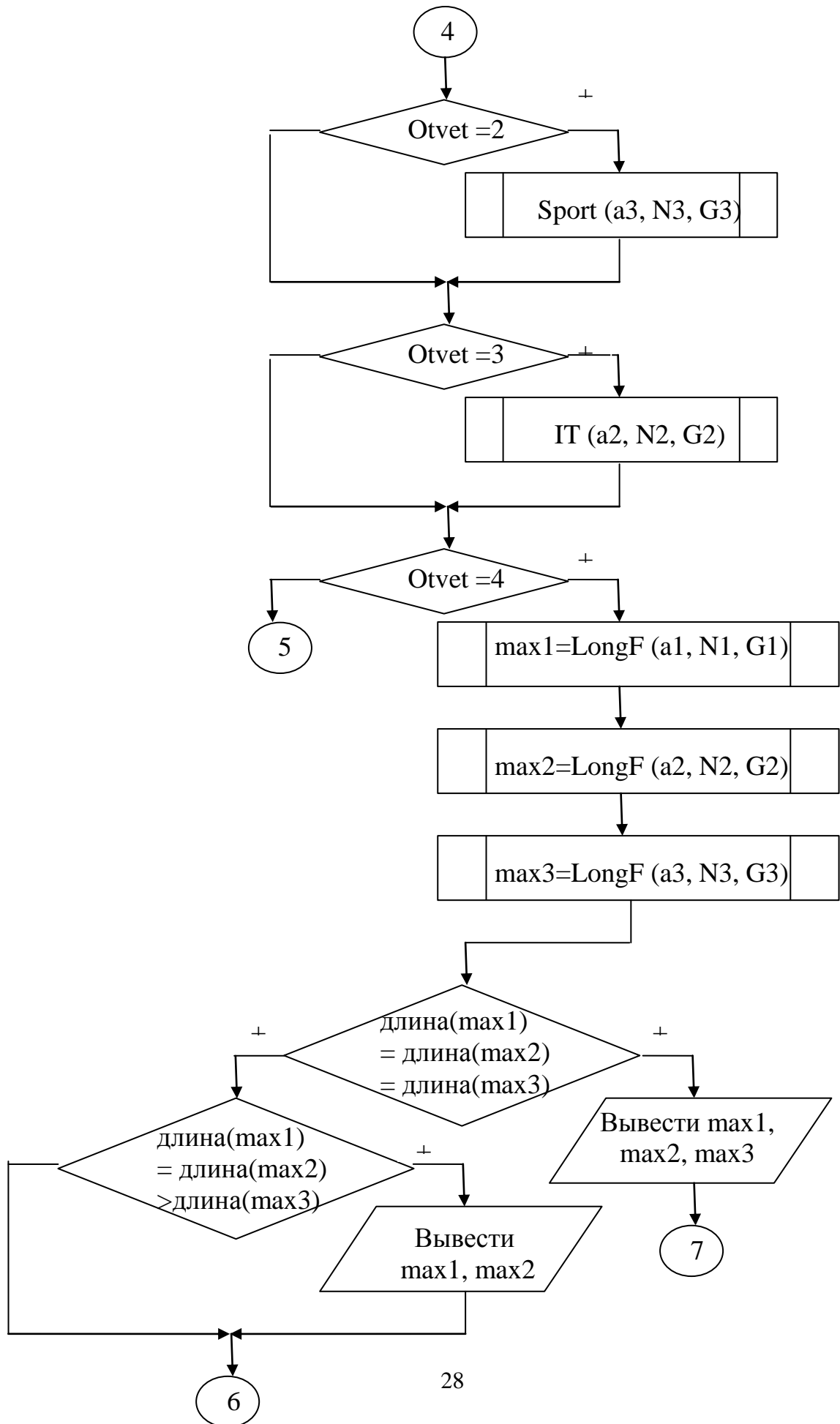
Рисунок 10 – Блок-схема алгоритма удаления структуры

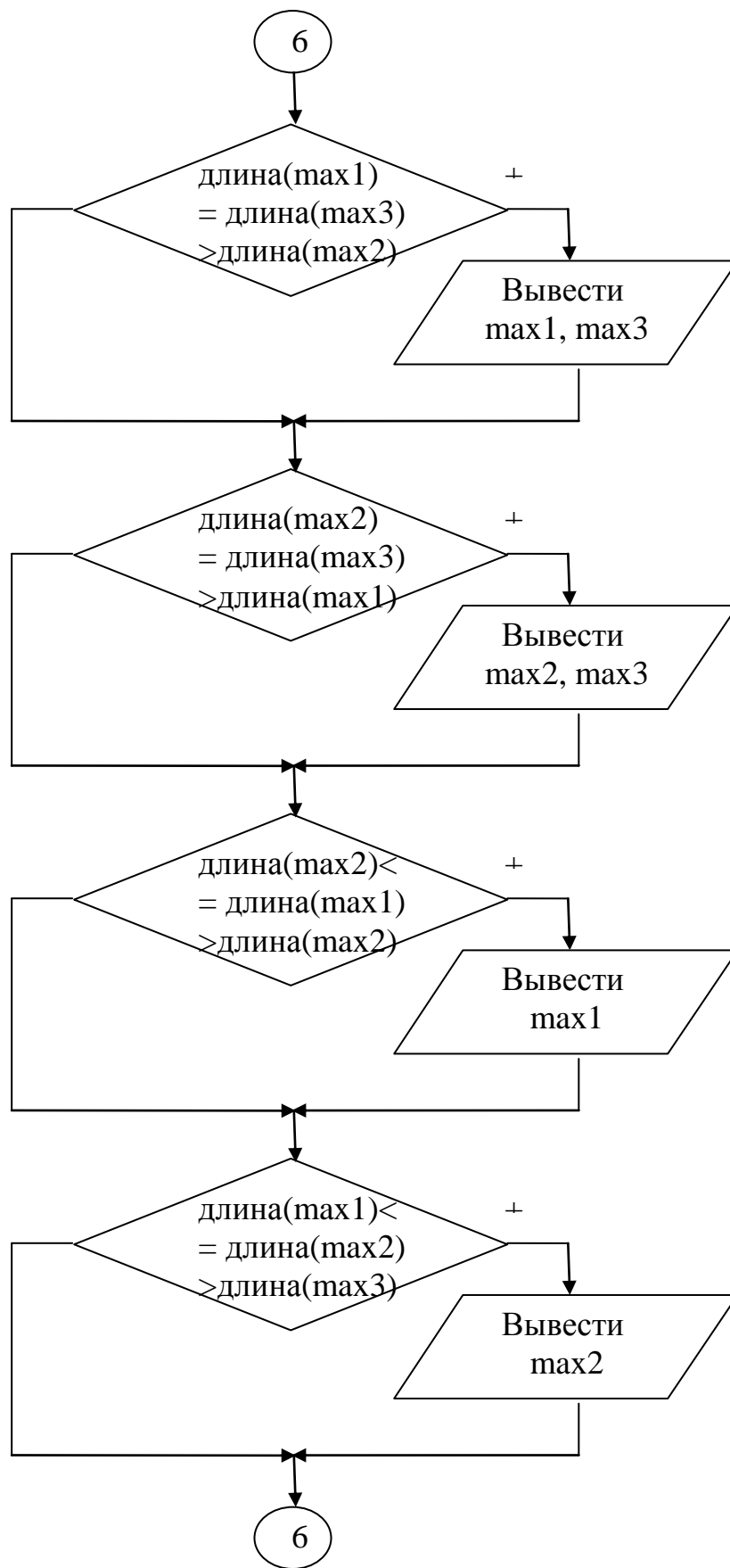
## 2.4 Структура программы

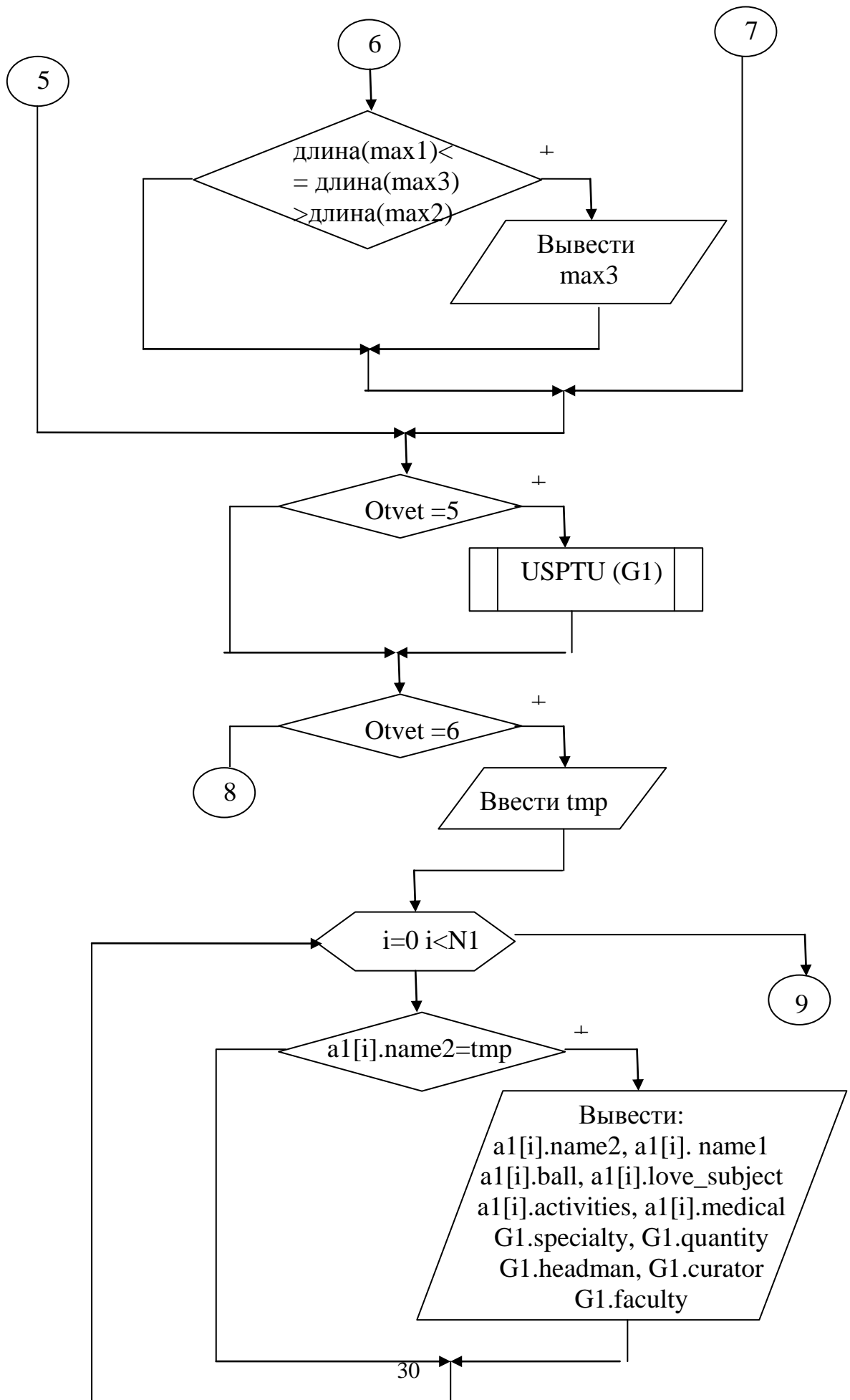
Блок-схема главной функции:

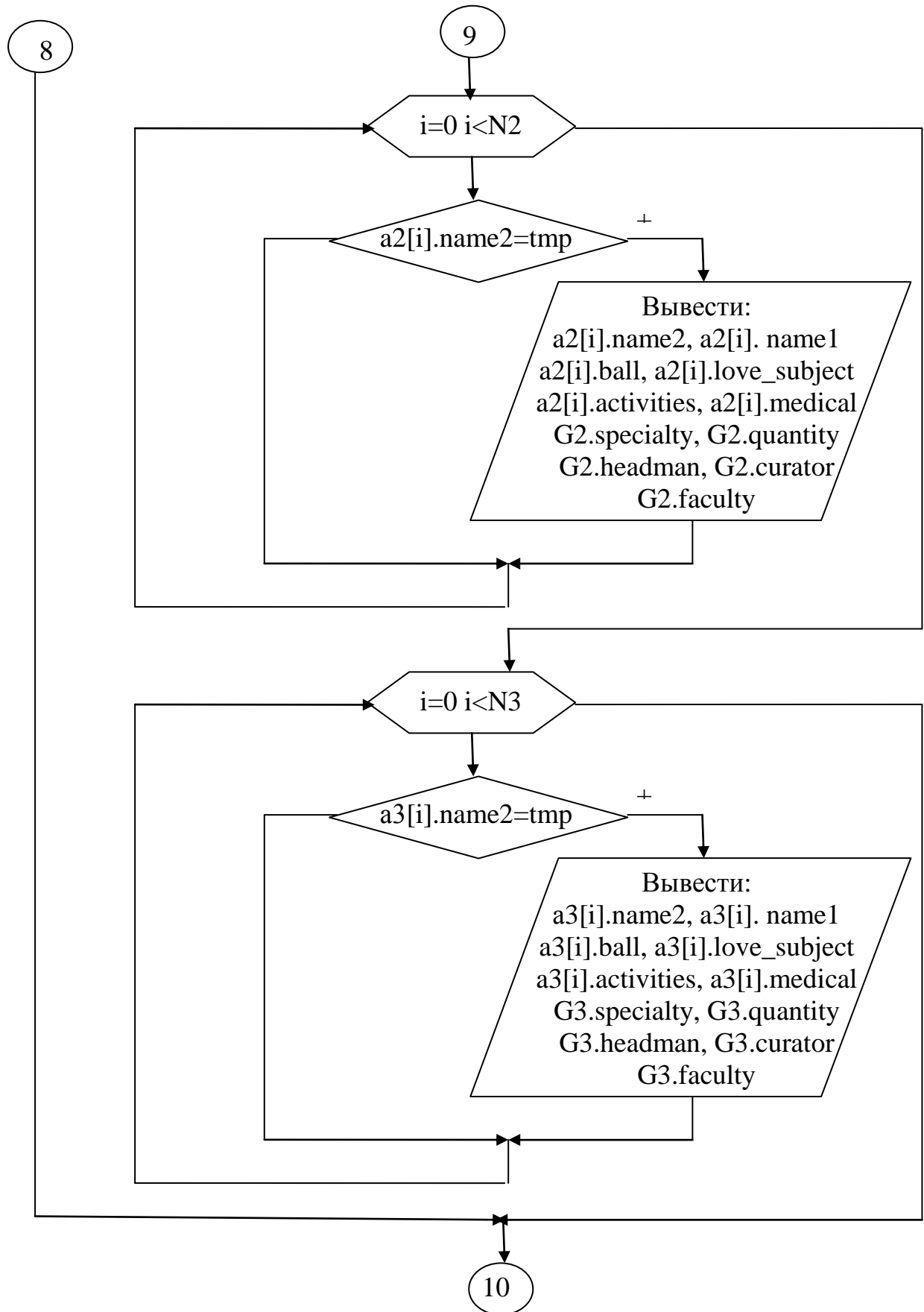




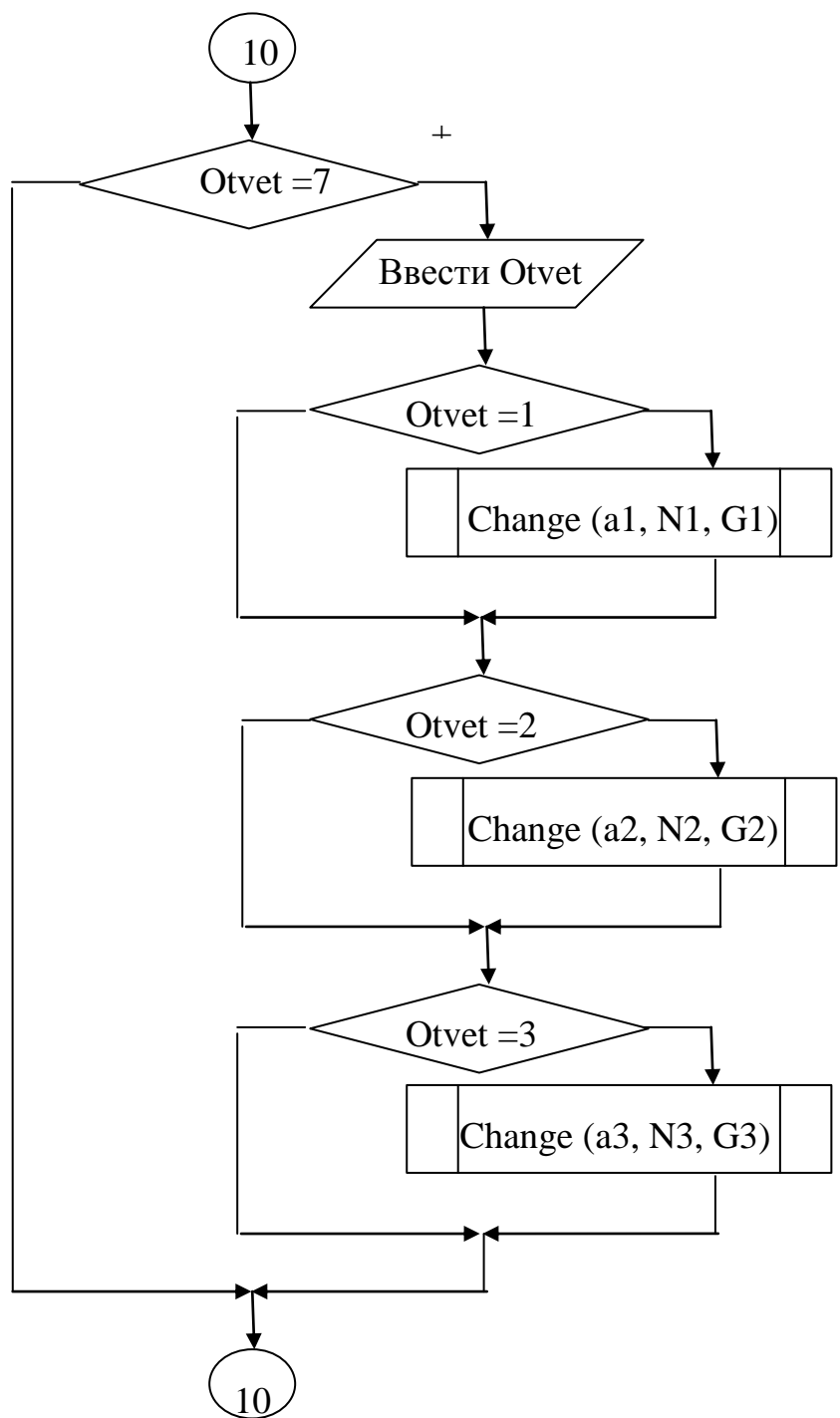


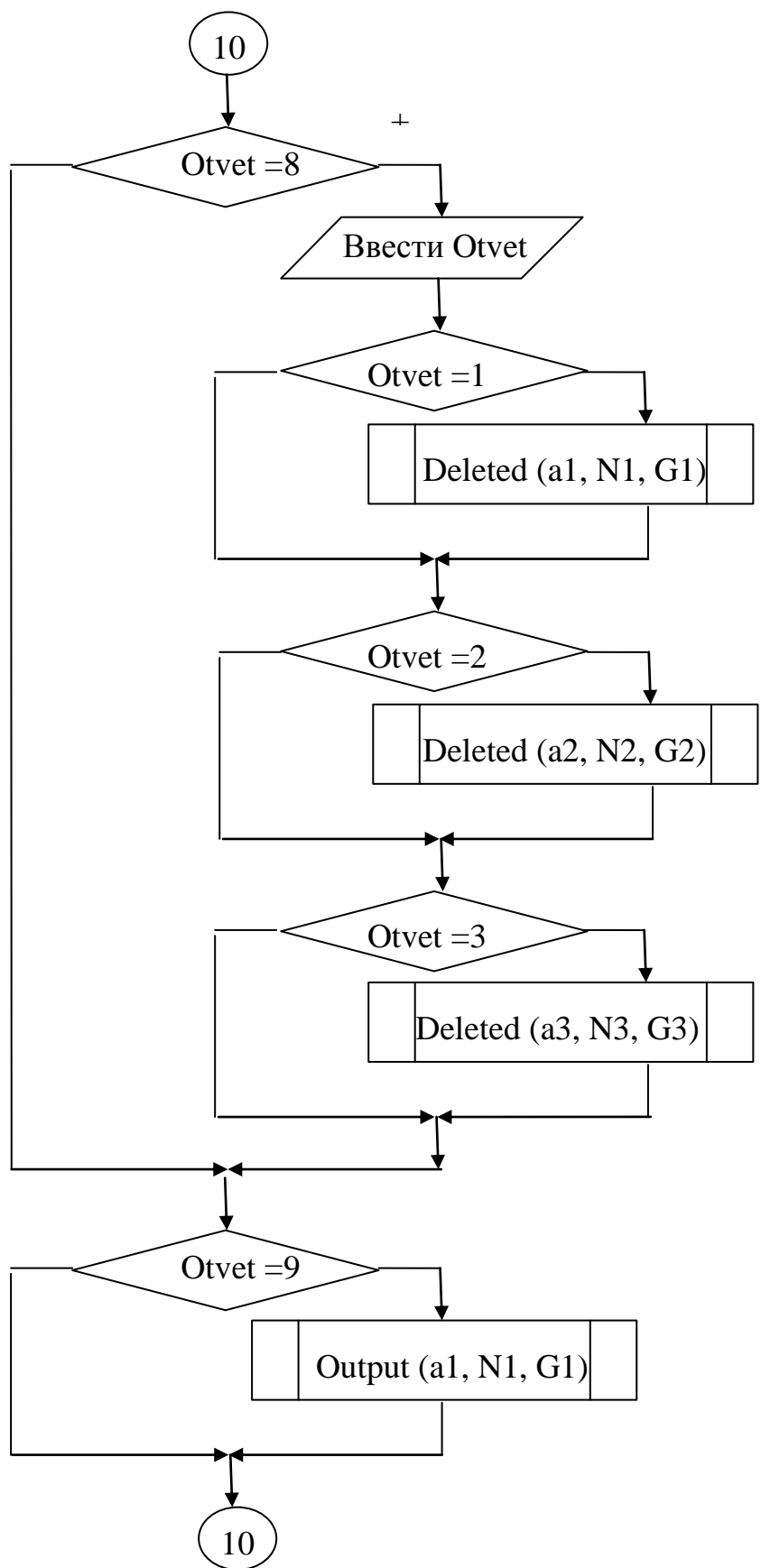












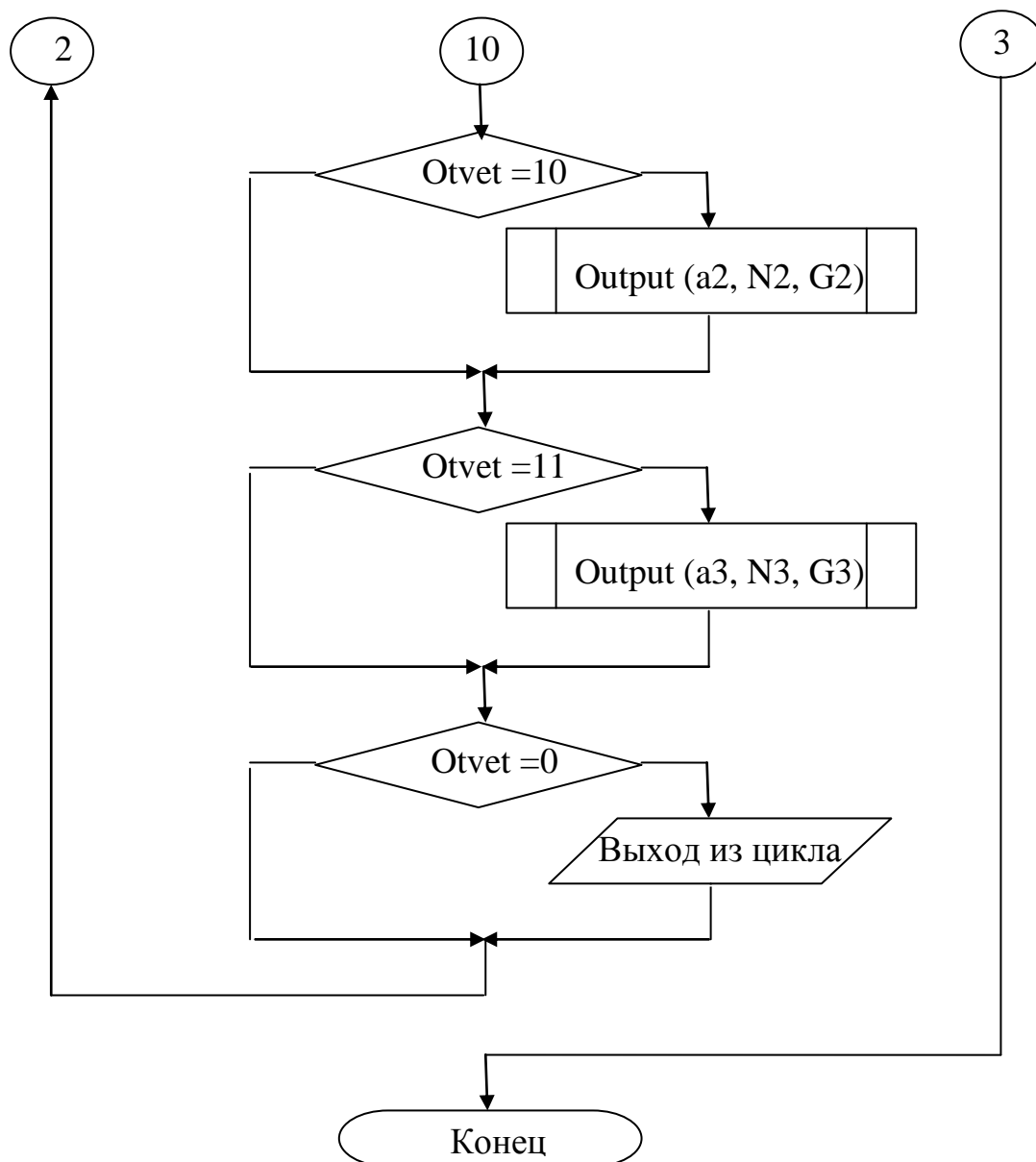


Рисунок 11 – Блок-схема главной программы

## 3 Описание программы

### 3.1 Код программы для консольного приложения

Код программы для консольного приложения представлен в листинге 1.

#### Листинг 1 - Код программы для консольного приложения

```
#include <iostream>
#include <fstream>
#include <locale.h>
#include <string.h>
#include <windows.h>

using namespace std;

struct Group
{
    char* specialty; // название группы
    int quantity; // количество человек в группе
    char* headman; // фамилия старосты
    char* curator; // фамилия куратора
    char* faculty; // факультет
};

struct Student
{
    Group group; // группа
    char* name1; // имя
    char* name2; // фамилия
    float ball; // средний балл студента
    char* love_subject; // любимый предмет
    char* activities; // внеучебная деятельность
    int medical; // медицинская группа
    bool deleted; // признак отчисления
};

void KBfill (Student* a, int N, Group X);
void Ffill (Student* a, int N, Group X);
void Output (Student* a, int N, Group X);
void SrBall (Student* a, int N, Group X);
int Sport (Student* a, int N, Group X);
int IT (Student* a, int N, Group X);
char* LongF (Student* a, int N, Group X);
void Change (Student* a, int N, Group X);
void Deleted (Student* a, int N, Group X);
void USPTU (Group X);

int main()
{
    setlocale(LC_ALL, "Russian");

    cout.width(4); // Общее количество символов
    cout.precision(2); // Количество символов после запятой
    cout.setf(ios::fixed);

    Group G1,G2,G3;
```

### 3.1 Код программы для оконного приложения:

Код программы для оконного приложения представлен в листинге2.

Листинг 2 - Код программы для оконного приложения

Файл Form1.h:

```
#pragma once

#include "My.h"
#include <fstream>

namespace forma {

    using namespace System;
    using namespace System::Runtime::InteropServices;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
    using namespace System::IO;
    using namespace System::Collections;
    using namespace std;

    /// <summary>
    /// Summary for Form1
    /// </summary>
    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            //
            //TODO: Add the constructor code here
            //
        }

    protected:
        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }

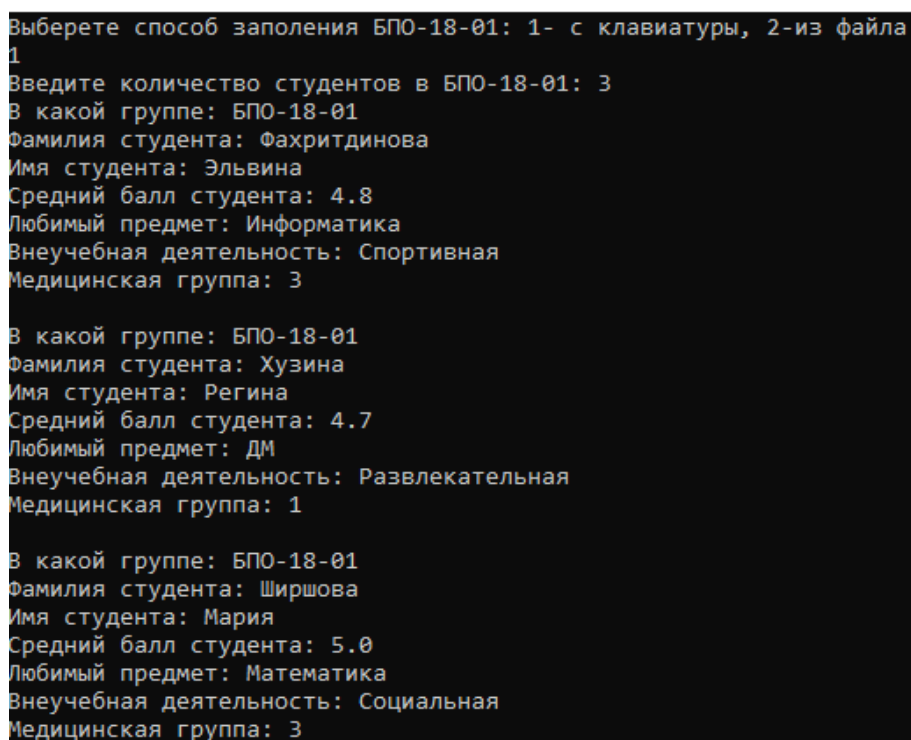
    protected:

    private:
        /// <summary>
        /// Required designer variable.
        static array< Group^ >^ GroupList = gcnew array< Group^ >(3);
```

## 4 Анализ результатов

### 4.1 Результаты программы для консольного приложения

Результаты программы для консольного приложения представлены на рисунках 12 - 23.



```
Выберете способ заполнения БПО-18-01: 1- с клавиатуры, 2-из файла
1
Введите количество студентов в БПО-18-01: 3
В какой группе: БПО-18-01
Фамилия студента: Фахритдинова
Имя студента: Эльвина
Средний балл студента: 4.8
Любимый предмет: Информатика
Внеучебная деятельность: Спортивная
Медицинская группа: 3

В какой группе: БПО-18-01
Фамилия студента: Хузина
Имя студента: Регина
Средний балл студента: 4.7
Любимый предмет: ДМ
Внеучебная деятельность: Развлекательная
Медицинская группа: 1

В какой группе: БПО-18-01
Фамилия студента: Ширшова
Имя студента: Мария
Средний балл студента: 5.0
Любимый предмет: Математика
Внеучебная деятельность: Социальная
Медицинская группа: 3
```

Рисунок 12 – Скриншот программы: заполнения структуры БПО-18-01 с клавиатуры

```
Выберете способ заполнения БПО-18-02: 1- с клавиатуры, 2-из файла
1
Введите количество студентов в БПО-18-02: 2
В какой группе: БПО-18-02
Фамилия студента: Каримова
Имя студента: Анжела
Средний балл студента: 5.0
Любимый предмет: Информатика
Внеучебная деятельность: Досугово-развлекательная
Медицинская группа: 3

В какой группе: БПО-18-02
Фамилия студента: Кузнецов
Имя студента: Владислав
Средний балл студента: 3.2
Любимый предмет: Философия
Внеучебная деятельность: Социальная
Медицинская группа: 3
```

Рисунок 13 – Скриншот программы: заполнения структуры БПО-18-02 с клавиатуры

```

Выберете способ заполнения в БПОи-18-01: 1- с клавиату
2
В какой группе: БПОи-18-01
Фамилия студента: Бодякова
Имя студента: Дарья
Средний балл студента: 5.00
Любимый предмет: Программирование
Внеучебная деятельность: Досугово-развлекательная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Гумеров
Имя студента: Эрик
Средний балл студента: 4.80
Любимый предмет: Физика
Внеучебная деятельность: Игровая
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Зуев
Имя студента: Павел
Средний балл студента: 5.00
Любимый предмет: Математика
Внеучебная деятельность: Трудовая
Медицинская группа: 1

В какой группе: БПОи-18-01
Фамилия студента: Ильясов
Имя студента: Артур
Средний балл студента: 4.20
Любимый предмет: Физика
Внеучебная деятельность: Проблемно-ценностная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Николаев
Имя студента: Анатолий
Средний балл студента: 4.70
Любимый предмет: История
Внеучебная деятельность: Социальная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Овсянникова
Имя студента: Анна
Средний балл студента: 5.00
Любимый предмет: Философия
Внеучебная деятельность: Социальная
Медицинская группа: 4

В какой группе: БПОи-18-01
Фамилия студента: Садовой
Имя студента: Данила
Средний балл студента: 4.70
Любимый предмет: ООПТ
Внеучебная деятельность: Игровая
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Сафиуллин
Имя студента: Эдуард
Средний балл студента: 3.00
Любимый предмет: ДМ
Внеучебная деятельность: Проблемно-ценностная
Медицинская группа: 3

```

Рисунок 14 – Скриншот программы: заполнения структуры БПОи-18-01 из файла



```

1- Студенты из БПО-18-01, чей средний балл выше среднего балла группы
2- Сколько студентов из БПОи-18-01 занимаются спортивным видом внеурочной деятельности
3- Сколько студентов из БПО-18-02 любят информатику
4- Студент с самой длинной фамилией
5- Вывод сведений о ВУЗе
6- Вывод сведений о группе, в которой учится студент
7- Изменить
8- Удалить
9- Вывод списка группы БПО-18-01
10- Вывод списка группы БПО-18-02
11- Вывод списка группы БПОи-18-01
0- выход

```

Рисунок 15 – Скриншот программы: меню

```

1
Средний балл группы БПО-18-01 = 4.83
Ширшова Мария 5.00
Для продолжения нажмите любую клавишу . . .

```

Рисунок 16 – Скриншот программы: функция 1

```

2
Тазтдинова Анна
1
Для продолжения нажмите любую клавишу . . .

```

Рисунок 16 – Скриншот программы: функция 2

```

3
Каримова Анжела
1
Для продолжения нажмите любую клавишу . . .

```

Рисунок 16 – Скриншот программы: функция 3

```

4
Самая длинная фамилия в БПО-18-01 Фахритдинова
Самая длинная фамилия в БПО-18-02 Каримова
Самая длинная фамилия в БПОи-18-01 Овсянникова
Самая длинная фамилия на потоке БПО-18 Фахритдинова
Для продолжения нажмите любую клавишу . . .

```

Рисунок 17 – Скриншот программы: функция 4

```
5
УГНТУ - Уфимский Государственный Нефтяной Технический Университет
1) Опорный ВУЗ России
2) Основан в 1948
3) Факультеты: ФАПП, АСИ, ФОНД, ГНФ, ФТТ, ТФ, ИЭС, ИНБ, МФ, ФЗО, ИДПО
4) Филиалы: Уфимский, Стерлитамакский, Салаватский, Октябрьский
Для продолжения нажмите любую клавишу . . . █
```

Рисунок 18 – Скриншот программы: функция 5

```
6
Введите фамилию студента Бодякова
Бодякова Дарья 5.00 Программирование Досугово-развлекательная 3
Название группы: БПОи-18-01
Количество человек в группе: 10
Фамилия старосты группы: Гумеров
Фамилия куратора группы: Ганиева
Факультет, которому принадлежит группа: ФАПП
Для продолжения нажмите любую клавишу . . . █
```

Рисунок 19 – Скриншот программы: функция 6

```
8
В какой группе удалить? 1- БПО-18-01 2- БПО-18-02 3- БПОи-18-01 2
Какие данные вас больше не интересую? Введите номер данной структуры (см. предыдущий вывод). Нажмите 0, чтобы закончить
2
Какие данные вас больше не интересую? Введите номер данной структуры (см. предыдущий вывод). Нажмите 0, чтобы закончить
0
В какой группе: БПО-18-02
Фамилия студента: Каримова
Имя студента: Анжела
Средний балл студента: 5.00
Любимый предмет: Информатика
Внеучебная деятельность: Досугово-развлекательная
Медицинская группа: 3
Для продолжения нажмите любую клавишу . . . █
```

Рисунок 20 – Скриншот программы: функция 8

```

7
В какой группе изменить? 1- БПО-18-01 2- БПО-18-02 3- БПОи-18-01 2
Какие данные вы хотите изменить? Введите номер данной структуры (см. предыдущий вывод). 1
Сколько полей вы хотите изменить? 1- все 2- некоторые
2
Изменить фамилию студента? 1- да 2- нет 2
Фамилия студента: Каримова
Изменить имя студента? 1- да 2- нет 2
Имя студента: Анжела
Изменить средний балл студента? 1- да 2- нет 2
Средний балл: 5.00
Изменить любимый предмет студента? 1- да 2- нет 2
Любимый предмет: Информатика
Изменить внеучебную деятельность студента? 1- да 2- нет 1
Внеучебная деятельность: Туристическая
Изменить медицинскую группу студента? 1- да 2- нет 2
Медицинская группа: 3

Фамилия студента: Кузнецов
Имя студента: Владислав
Средний балл студента: 3.20
Любимый предмет: Философия
Внеучебная деятельность: Социальная
Медицинская группа: 3

Продолжить? 1-да 2-нет 1

Какие данные вы хотите изменить? Введите номер данной структуры (см. предыдущий вывод). 2
Фамилия студента: Каримова
Имя студента: Анжела
Средний балл студента: 5.00
Любимый предмет: Информатика
Внеучебная деятельность: Туристическая
Медицинская группа: 3

Сколько полей вы хотите изменить? 1- все 2- некоторые
1
В какой группе: БПО-18-02
Фамилия студента: Стариков
Имя студента: Роман
Средний балл студента: 4.7
Любимый предмет: Физика
Внеучебная деятельность: Трудовая
Медицинская группа: 4

Продолжить? 1-да 2-нет 2

В какой группе: БПО-18-02
Фамилия студента: Каримова
Имя студента: Анжела
Средний балл студента: 5.00
Любимый предмет: Информатика
Внеучебная деятельность: Туристическая
Медицинская группа: 3

В какой группе: БПО-18-02
Фамилия студента: Стариков
Имя студента: Роман
Средний балл студента: 4.70
Любимый предмет: Физика
Внеучебная деятельность: Трудовая
Медицинская группа: 4

Для продолжения нажмите любую клавишу . . . █

```

Рисунок 20 – Скриншот программы: функция 7

```
9
В какой группе: БПО-18-01
Фамилия студента: Кутлуметов
Имя студента: Мансур
Средний балл студента: 5.00
Любимый предмет: Программирование
Внеучебная деятельность: Туристическая
Медицинская группа: 3

В какой группе: БПО-18-01
Фамилия студента: Ширшова
Имя студента: Мария
Средний балл студента: 5.00
Любимый предмет: Математика
Внеучебная деятельность: Социальная
Медицинская группа: 3

Для продолжения нажмите любую клавишу . . .
```

Рисунок 21 – Скриншот программы: функция 9

```
10
В какой группе: БПО-18-02
Фамилия студента: Каримова
Имя студента: Анжела
Средний балл студента: 5.00
Любимый предмет: Информатика
Внеучебная деятельность: Туристическая
Медицинская группа: 3

В какой группе: БПО-18-02
Фамилия студента: Стариков
Имя студента: Роман
Средний балл студента: 4.70
Любимый предмет: Физика
Внеучебная деятельность: Трудовая
Медицинская группа: 4

Для продолжения нажмите любую клавишу . . .
```

Рисунок 22 – Скриншот программы: функция 10

```

11
В какой группе: БПОи-18-01
Фамилия студента: Бодякова
Имя студента: Дарья
Средний балл студента: 5.00
Любимый предмет: Программирование
Внеучебная деятельность: Досугово-развлекательная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Гумеров
Имя студента: Эрик
Средний балл студента: 4.80
Любимый предмет: Физика
Внеучебная деятельность: Игровая
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Ильясов
Имя студента: Артур
Средний балл студента: 4.20
Любимый предмет: Физика
Внеучебная деятельность: Проблемно-ценностная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Николаев
Имя студента: Анатолий
Средний балл студента: 4.70
Любимый предмет: История
Внеучебная деятельность: Социальная
Медицинская группа: 3

В какой группе: БПОи-18-01
Фамилия студента: Овсянникова
Имя студента: Анна
Средний балл студента: 5.00

```

Рисунок 23 – Скриншот программы: функция 11

```

0
Для продолжения нажмите любую клавишу . . .

```

Рисунок 23 – Скриншот программы: функция 0

## 4.2 Результаты программы для оконного приложения

Результаты программы для консольного приложения представлены на рисунках 23 - 30.

БПО-18

Студент    Список

Информация о студенте №

Группа

Фамилия

Имя

Средний балл

Любимый предмет

Внеучебная деятельность

Медицинская группа

Информация о группе

Количество человек в группе

Фамилия старосты

Фамилия куратора

Факультет

Добавить    Добавить из файла    Записать изменения

Найти по фамилии

Самая короткая фамилия

Средний балл БПО-18

Сколько человек с БПО-18 любят физику

Рисунок 23 – Скриншот вкладки «Студент»

БПО-18

Студент    Список

	Группа	Фамилия	Имя	Средний балл	Любимый предмет	Внеучебная деятельность	Медицинская группа
▶	БПО-18-01	Андреев	Никита	3,0	Информатика	Игровая	3
	БПО-18-01	Гайфуллин	Вакиль	4,8	Программирова...	Спортивная	1
	БПО-18-01	Карпов	Иван	3,7	ООПТ	Трудовая	4
	БПО-18-01	Курилов	Андрей	3,8	Физика	Спортивная	4
	БПО-18-01	Кутлумбетов	Мансур	5,0	Программирова...	Туристическая	3
	БПО-18-01	Мамбетов	Камиль	4,8	История	Трудовая	4
	БПО-18-01	Мустафин	Радмир	4,1	Физика	Спортивная	1
	БПО-18-01	Родин	Данил	3,4	ДМ	Спортивная	2
	БПО-18-01	Султанянов	Александр	4,1	Математика	Трудовая	2
	БПО-18-01	Фахритдинова	Эльвина	4,8	История	Спортивная	3
	БПО-18-01	Хузина	Регина	4,7	ООПТ	Досугово-развл...	1
	БПО-18-01	Ширшова	Мария	5,0	Математика	Социальная	3
	БПО-18-02	Агзамов	Эльдар	4,5	ДМ	Туристическая	3
	БПО-18-02	Байрамгулов	Айназ	4,2	Информатика	Игровая	3
	БПО-18-02	Бамидинов	Ринат	5,0	Программирова...	Досугово-развл...	2

Выбрать данные из текущей строки для изменений      Удалить текущую строку

Рисунок 24 – Скриншот вкладки «Список» после нажатия кнопки «Добавить из файла»

Самая короткая фамилия      Зуев

Рисунок 25 – Скриншот результата функции поиска самой короткой фамилии

Средний балл БПО-18      4,215

Сколько человек с БПО-18 любят физику     

Гайфуллин  
Кутлумбетов  
Мамбетов  
Фахритдинова  
Хузина  
Ширшова  
Агзамов  
Галиуллина  
Каримова  
Погачева  
Стариков  
Ямилова  
Бодякова  
Гумеров  
Зуев  
Николаев  
Овсянникова  
Садовой  
Тазтдинова

Рисунок 26 – Скриншот результата функции поиска среднего балла и тех, у кого балл выше среднего

Рисунок 27 – Скриншот результата функции поиска людей, которые любят физику и их количество

Рисунок 28 – Скриншот результата функции поиска студента по его фамилии и вывод сведений о его группе

▶	БПО-18-01	Мустафин	Радмир	4,1	Физика	Спортивная	1
▶	БПО-18-01	Родин	Данил	3,4	ДМ	Спортивная	2
	БПО-18-01	Мустафин	Радмир	4,1	Физика	Спортивная	1
▶	БПО-18-01	Родин	Данил	3,4	Экология	Спортивная	1

Рисунок 29 – Скриншот результата функции изменения

БПО-18-02	Юлчурун	Айдар	3,0	История	Проблемно-цен...	3
БПО-18-02	Якупов	Вадим	3,0	ДМ	Проблемно-цен...	2
БПО-18-02	Ямилова	Алия	4,5	История	Спортивная	3
БПОи-18-01	Бодякова	Дарья	5,0	Программирова...	Досугово-развл...	3
БПО-18-02	Юлчурун	Айдар	3,0	История	Проблемно-цен...	3
БПО-18-02	Ямилова	Алия	4,5	История	Спортивная	3
БПОи-18-01	Бодякова	Дарья	5,0	Программирова...	Досугово-развл...	3

Рисунок 30 – Скриншот результата функции удаления



## **ЗАКЛЮЧЕНИЕ**

Выше представленная программа работает корректно и удовлетворяет всем задачам, поставленным в курсовой работе.

Использование структур очень удобно, если приходится иметь дело с большим объёмом данных. Это значит, что можно поместить несколько переменных разного типа, объединив их общим именем. Это делает программу более компактной и более гибкой для внесения изменений.

Все поставленные цели и задачи достигнуты. Поэтому можно сделать вывод, что получилось реализовать такой программный продукт, который было бы удобно использовать в учебных заведениях.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1 Романов, Е.Л. Практикум по программированию в С++: Уч. пособие/ Е.Л. Романов. – СПб: БХВ-Петербург; Новосибирск.: Изд-во НГТУ, 2004. – 432 с.
- 2 Габдуллина, А.А. Создание приложения с графическим интерфейсом пользователя в среде MICROSOFT VISUAL STUDIO 2010: Уч. пособие/ А.А. Габдуллина, Е.В. Дружинская. – Уфа.: Изд-во УГНТУ, 2015. – 27 с.
- 3 Портал о программировании Code-Live [Электронный ресурс]. – URL: <https://code-live.ru/post/first-windows-form/> (дата обращения 25.10.2019).
- 4 Сайт о программировании METANIT [Электронный ресурс]. – URL: <https://metanit.com/sharp/windowsforms/2.1.php> (дата обращения 25.10.2019).
- 5 Служба поддержки Майкрософт [Электронный ресурс]. – URL: <https://support.microsoft.com/ru-ru/help/983509/description-of-visual-studio-2010-service-pack-1> (дата обращения 31.10.2019).