



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ,
информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №7

«Обработка файлов последовательного доступа»

ДИСЦИПЛИНА: «Высокоуровневое программирование»

Выполнил: студент гр. ИУК4-11Б _____ (_____ Суриков Н.С _____)
(подпись) (Ф.И.О.)

Проверил: _____ (_____ Пчелинцева Н.И. _____)
(подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Цель работы: приобретение практических навыков процедурного программирования средствами языка C++.

Задачи:

1. Познакомиться с организацией файлов последовательного доступа.
2. Научиться создавать объекты файловых потоков для связи с текстовым файлом на внешнем носителе.
3. Изучить основные программные средства для последовательной обработки файлов указанного типа.
4. Научиться создавать пользовательское меню для демонстрации заданий лабораторной работы.

Вариант 20

Формулировка общего задания:

1. Создать файл, содержащий сведения о том, какие из 5 предложенных дисциплин желает слушать студент.

Структура записи:

- фамилия студента;
- номер группы;
- средний балл;
- 5 дисциплин, где '*' показывает выбранную дисциплину.

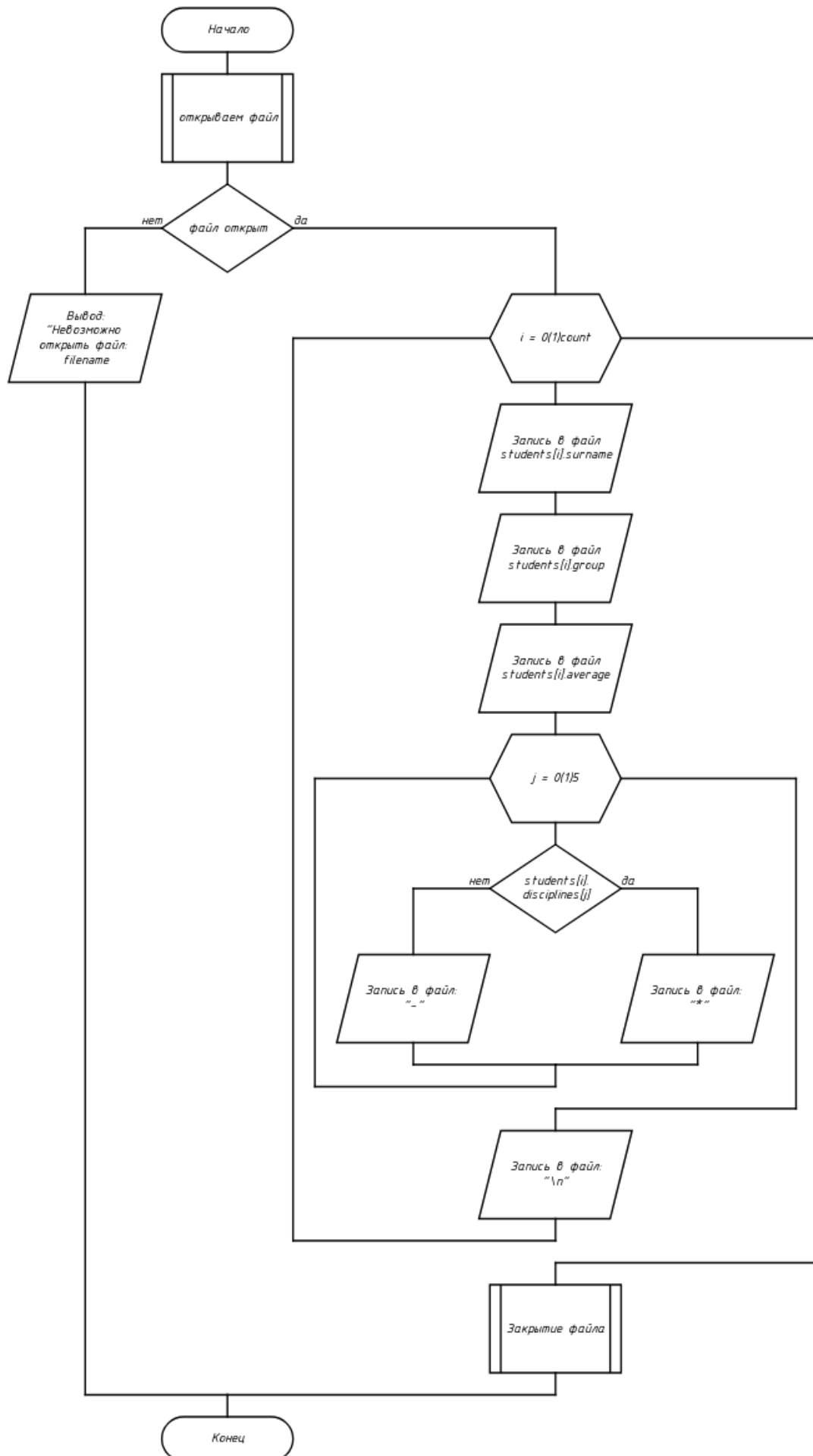
Создать файл, содержащий данные о тех, кто желает прослушать дисциплину XX.

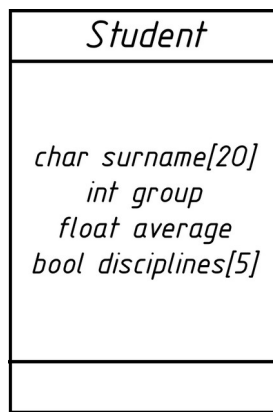
Если желающих больше 10, то отобрать тех студентов, у которых более высокий средний балл

2. Написать программу сравнения двух файлов: должна печататься первая строка, в которой они различаются. Если файлы идентичны, то выдать сообщение.

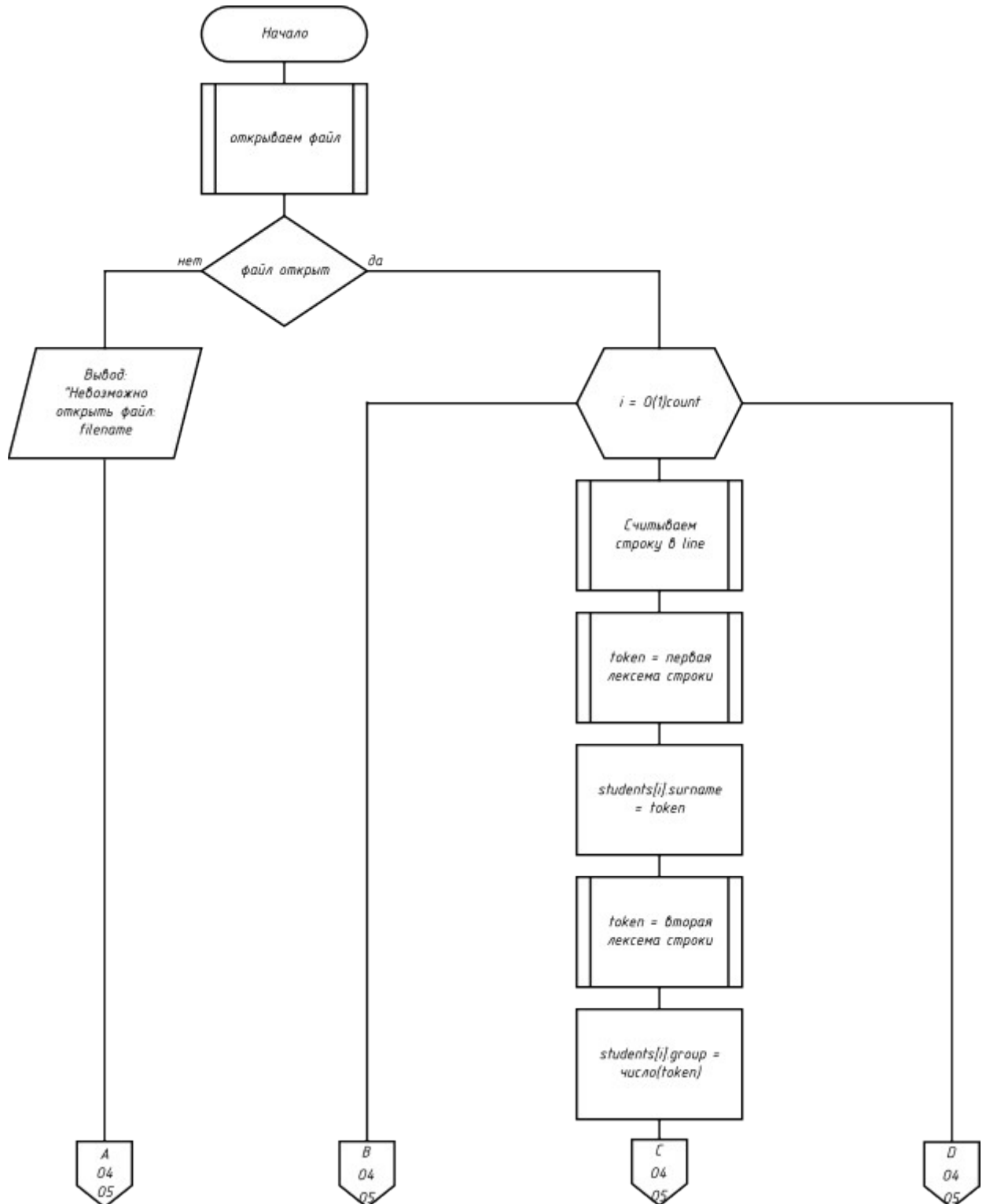
Блок схемы подпрограмм:

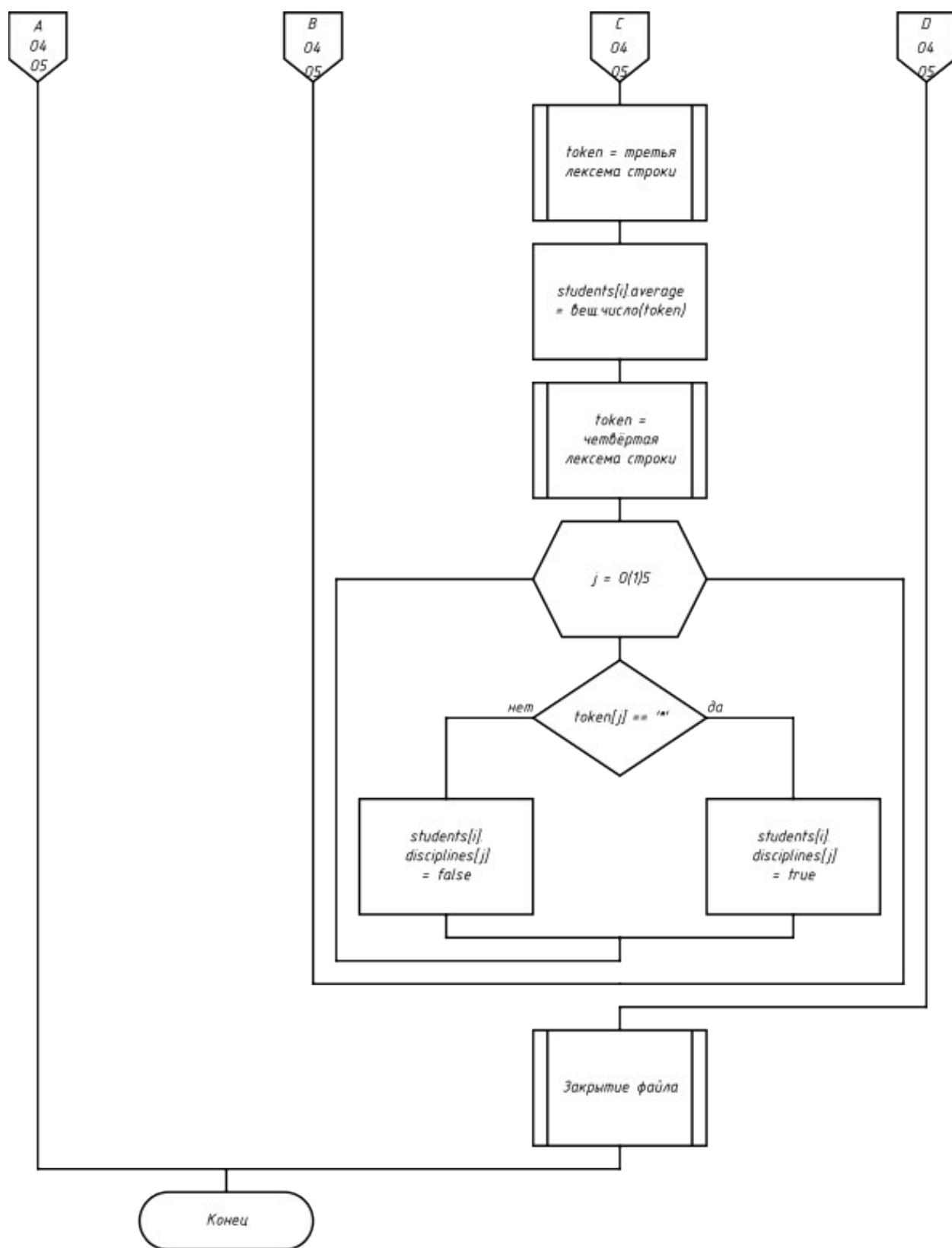
void writeStudents(Student students[], int count, const char *filename)

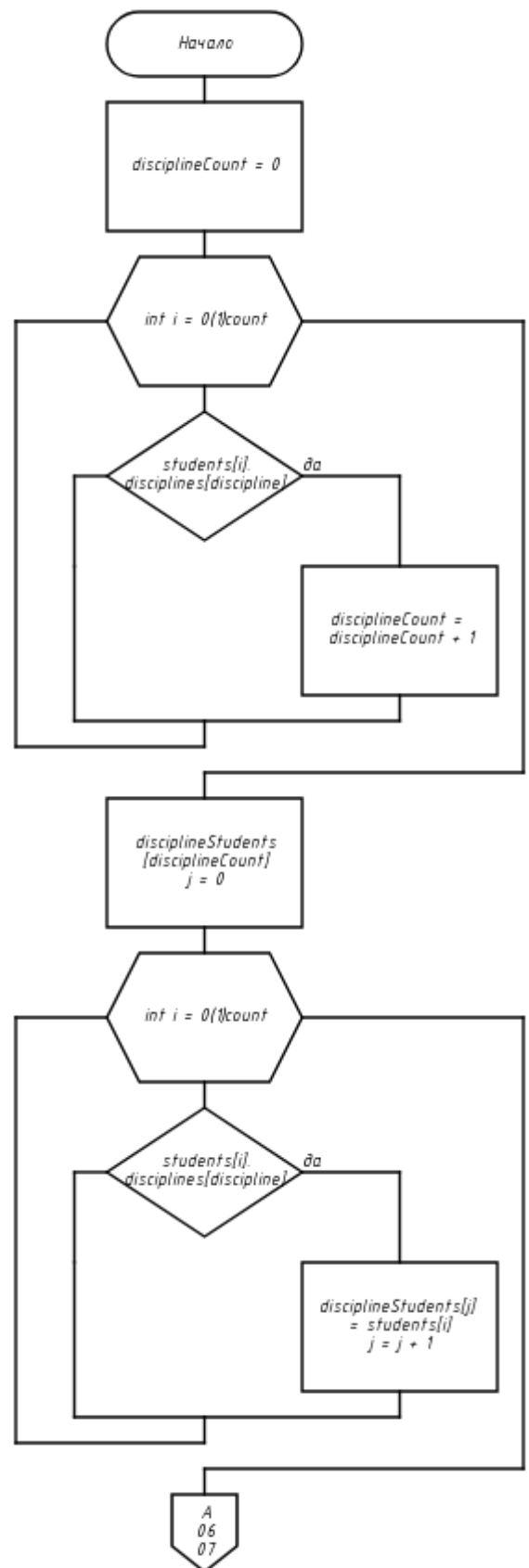
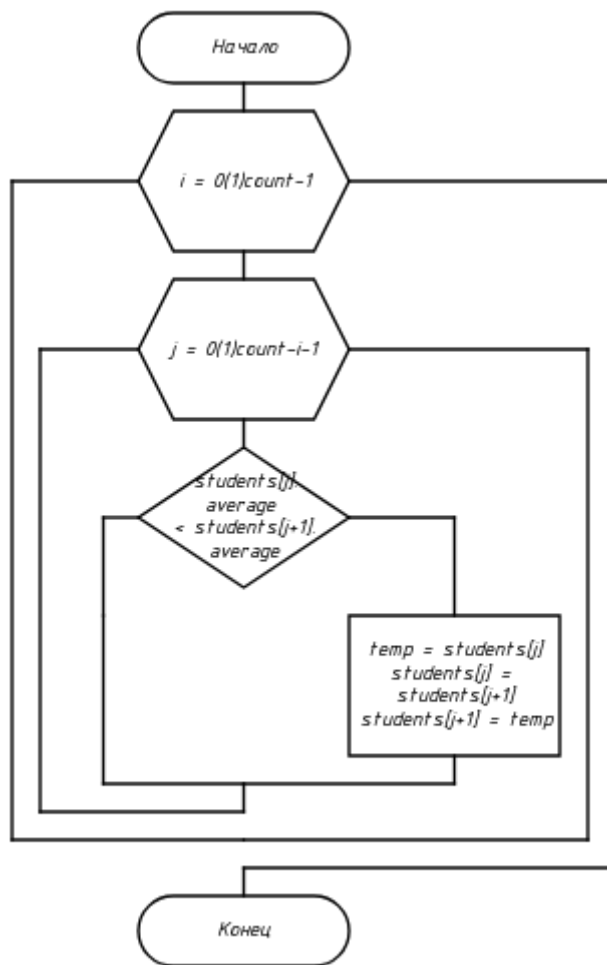




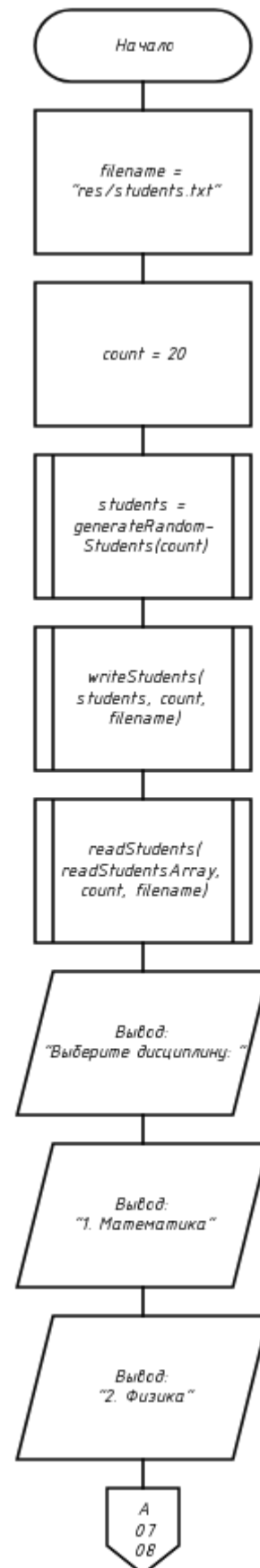
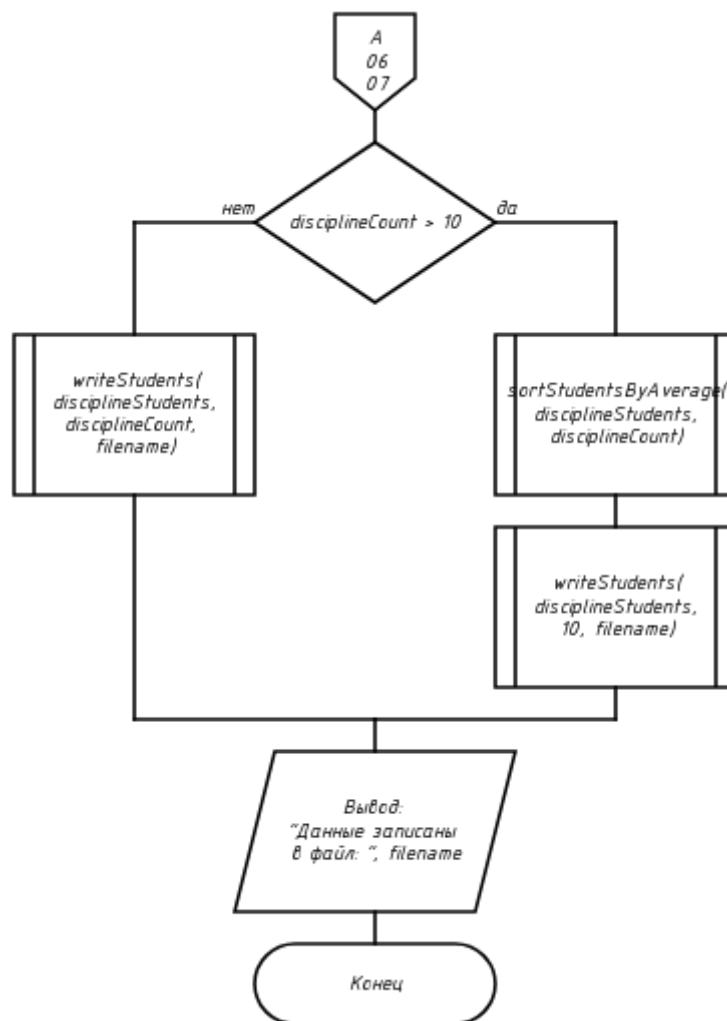
void readStudents(Student students[], int count, const char *filename)

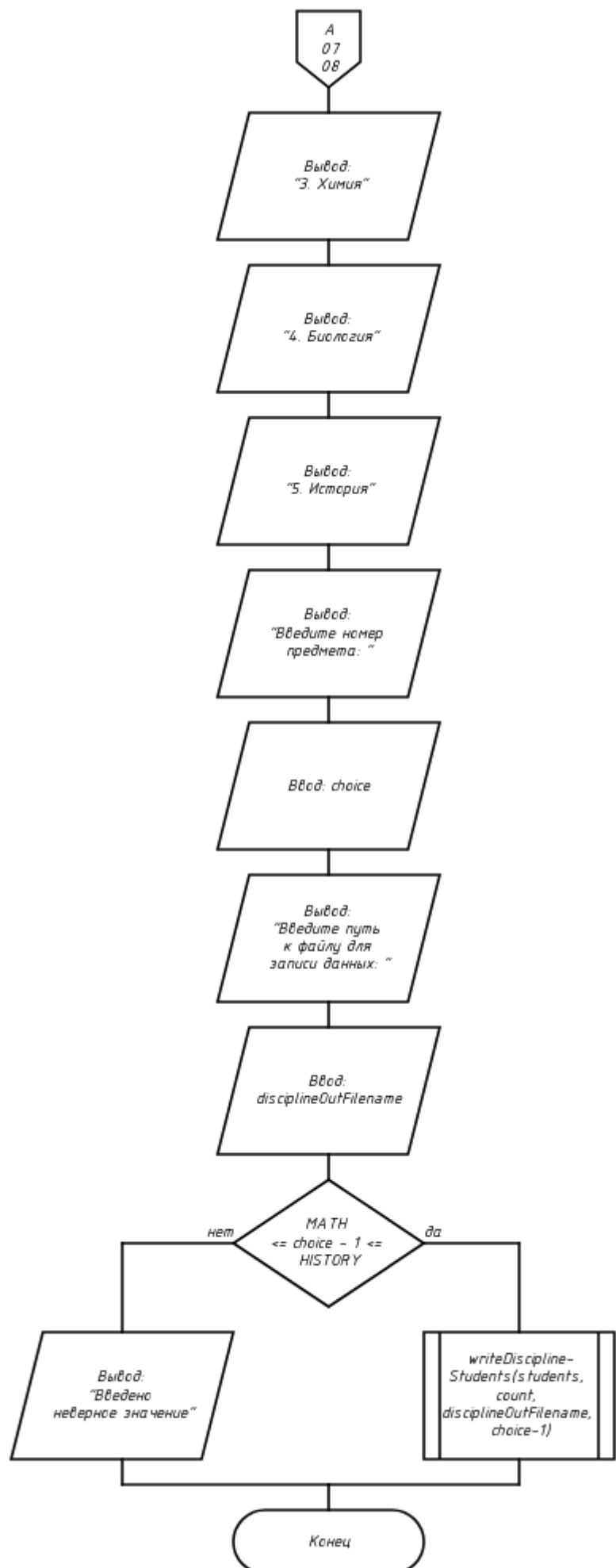
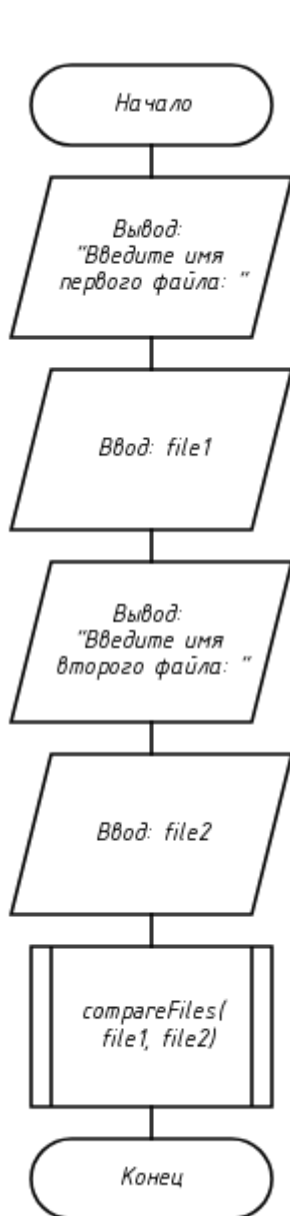




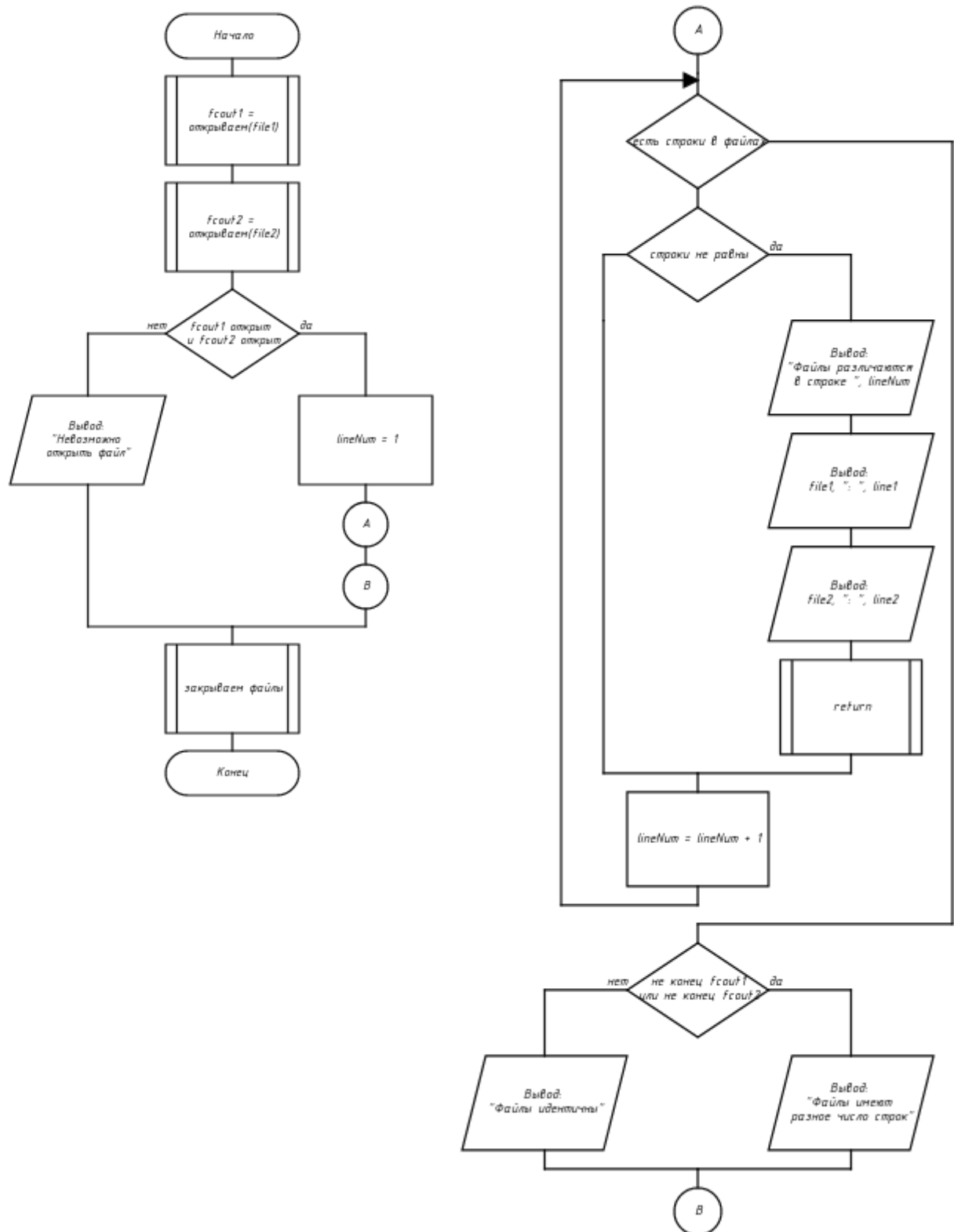


void writeDisciplineStudents(Student students[], int count, const char *filename, Discipline discipline)

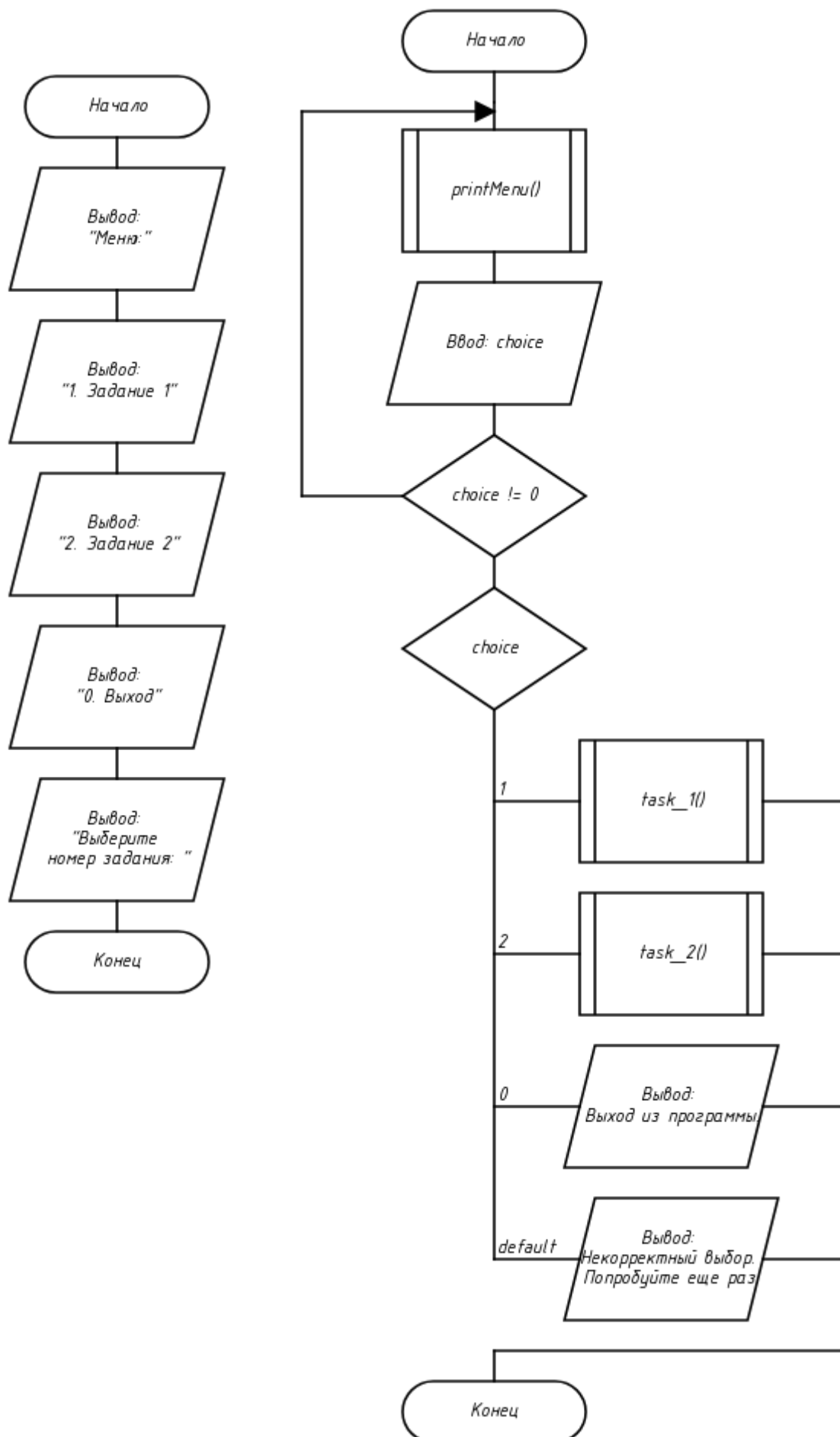





```
void compareFiles(const char *file1, const char *file2)
```



Блок схема основной программы:



Листинг программы:

Файл task_1.h:

```
1  #ifndef TASK_1_H
2  #define TASK_1_H
3
4  enum Discipline
5  {
6      MATH,
7      PHYSICS,
8      CHEMISTRY,
9      BIOLOGY,
10     HISTORY
11 };
12
13 struct Student
14 {
15     char surname[20];
16     int group;
17     float average;
18     bool disciplines[5];
19 };
20
21 Student *generateRandomStudents(int count);
22 void writeStudents(Student students[], int count, const char *filename);
23 void readStudents(Student students[], int count, const char *filename);
24 void sortStudentsByAverage(Student students[], int count);
25 void writeDisciplineStudents(Student students[], int count, const char
*filename, Discipline discipline);
26 void task_1();
27
28 #endif // TASK_1_H
```

Файл task_1.cpp:

```
1  #include <iostream>
2  #include <fstream>
3  #include <cstring>
4  #include <cstdlib>
5  #include <ctime>
6
7  #include "task_1.h"
8
9  using namespace std;
10
11 Student *generateRandomStudents(int count)
12 {
13     srand(time(nullptr));
14
15     Student *students = new Student[count];
16
17     for (int i = 0; i < count; i++)
18     {
19         for (int j = 0; j < 19; j++)
20         {
21             students[i].surname[j] = 'A' + rand() % 26;
22         }
23     }
24 }
```

```

23 students[i].surname[19] = '\0';
24
25 students[i].group = rand() % 10 + 1;
26
27 students[i].average = static_cast<float>(rand() % 51) / 10;
28
29 for (int j = 0; j < 5; j++)
30 {
31     students[i].disciplines[j] = rand() % 2 == 0;
32 }
33 }
34
35 return students;
36 }
37
38 void writeStudents(Student students[], int count, const char *filename)
39 {
40     ofstream file(filename);
41     if (file.is_open())
42     {
43         for (int i = 0; i < count; i++)
44         {
45             file << students[i].surname << " ";
46             file << students[i].group << " ";
47             file << students[i].average << " ";
48             for (int j = 0; j < 5; j++)
49             {
50                 if (students[i].disciplines[j])
51                 {
52                     file << "*";
53                 }
54                 else
55                 {
56                     file << "-";
57                 }
58             }
59             file << endl;
60         }
61         file.close();
62         cout << "Данные записаны в файл: " << filename << endl;
63     }
64     else
65     {
66         cout << "Невозможно открыть файл: " << filename << endl;
67     }
68 }
69
70 void readStudents(Student students[], int count, const char *filename)
71 {
72     ifstream file(filename);
73     if (file.is_open())
74     {
75         for (int i = 0; i < count; i++)
76         {
77             char line[256];
78             file.getline(line, 256);
79
80             char *token = strtok(line, " ");
81             strcpy(students[i].surname, token);
82
83             token = strtok(nullptr, " ");
84             students[i].group = atoi(token);
85
86             token = strtok(nullptr, " ");

```

```

87  students[i].average = atof(token);
88
89  token = strtok(nullptr, " ");
90  for (int j = 0; j < 5; j++)
91  {
92      if (token[j] == '*')
93      {
94          students[i].disciplines[j] = true;
95      }
96      else
97      {
98          students[i].disciplines[j] = false;
99      }
100 }
101 }
102 file.close();
103 }
104 else
105 {
106     cout << "Невозможно открыть файл: " << filename << endl;
107 }
108 }
109
110 void sortStudentsByAverage(Student students[], int count)
111 {
112     for (int i = 0; i < count - 1; i++)
113     {
114         for (int j = 0; j < count - i - 1; j++)
115         {
116             if (students[j].average < students[j + 1].average)
117             {
118                 Student temp = students[j];
119                 students[j] = students[j + 1];
120                 students[j + 1] = temp;
121             }
122         }
123     }
124 }
125
126 void writeDisciplineStudents(Student students[], int count, char
*filename, Discipline discipline)
127 {
128     int disciplineCount = 0;
129     for (int i = 0; i < count; i++)
130     {
131         if (students[i].disciplines[discipline])
132         {
133             disciplineCount++;
134         }
135     }
136
137     Student disciplineStudents[disciplineCount];
138     int j = 0;
139     for (int i = 0; i < count; i++)
140     {
141         if (students[i].disciplines[discipline])
142         {
143             disciplineStudents[j] = students[i];
144             j++;
145         }
146     }
147
148     if (disciplineCount > 10)
149     {

```

```

150 sortStudentsByAverage(disciplineStudents, disciplineCount);
151 writeStudents(disciplineStudents, 10, filename);
152 }
153 else
154 {
155 writeStudents(disciplineStudents, disciplineCount, filename);
156 }
157 }
158
159 void task_1()
160 {
161 const char *filename = "res/students.txt";
162
163 int count{20};
164
165 Student *students = generateRandomStudents(count);
166
167 writeStudents(students, count, filename);
168
169 Student readStudentsArray[count];
170 readStudents(readStudentsArray, count, filename);
171
172 int choice;
173 cout << "Выберите дисциплину: " << endl;
174 cout << "1. Математика" << endl;
175 cout << "2. Физика" << endl;
176 cout << "3. Химия" << endl;
177 cout << "4. Биология" << endl;
178 cout << "5. История" << endl;
179 cout << "Введите номер предмета: ";
180 cin >> choice;
181
182 char disciplineOutFilename[256];
183 cout << "Введите путь к файлу для записи данных: ";
184 cin >> disciplineOutFilename;
185
186 if (choice - 1 <= HISTORY && choice - 1 >= MATH)
187 {
188 writeDisciplineStudents(students, count, disciplineOutFilename,
(Discipline)(choice - 1));
189 }
190 else
191 {
192 cout << "Введено неверное значение" << endl;
193 }
194 }
195

```

Файл task_2.h:

```

1 #ifndef TASK_2_H
2 #define TASK_2_H
3
4 void compareFiles(const char *file1, const char *file2);
5 void task_2();
6
7 #endif // TASK_2_H

```

Файл task_2.cpp:

```
1  #include <iostream>
2  #include <fstream>
3  #include <cstring>
4
5  #include "task_2.h"
6
7  using namespace std;
8
9  void compareFiles(const char *file1, const char *file2)
10 {
11     ifstream fcout1(file1);
12     ifstream fcout2(file2);
13
14     if (fcout1.is_open() && fcout2.is_open())
15     {
16         char line1[256], line2[256];
17         int lineNum = 1;
18
19         while (fcout1.getline(line1, 256) && fcout2.getline(line2, 256))
20         {
21             if (strcmp(line1, line2) != 0)
22             {
23                 cout << "Файлы различаются в строке " << lineNum << ":" << endl;
24                 cout << file1 << ": " << line1 << endl;
25                 cout << file2 << ": " << line2 << endl;
26                 return;
27             }
28             lineNum++;
29         }
30
31         if (!fcout1.eof() || !fcout2.eof())
32         {
33             cout << "Файлы имеют разное число строк" << endl;
34         }
35         else
36         {
37             cout << "Файлы идентичны" << endl;
38         }
39     }
40     else
41     {
42         cout << "Невозможно открыть файл" << endl;
43     }
44
45     fcout1.close();
46     fcout2.close();
47 }
48
49 void task_2()
50 {
51     char file1[256];
52     char file2[256];
53
54     cout << "Введите имя первого файла: ";
55     cin >> file1;
56
57     cout << "Введите имя второго файла: ";
58     cin >> file2;
59
60     compareFiles(file1, file2);
61 }
```

Файл main.cpp:

```
1  #include <iostream>
2  #include <fstream>
3  #include <cstring>
4
5  #include "task_1.h"
6  #include "task_2.h"
7
8  // cd "/home/syricoff/bmstu/ВП/Суриков_НС_ИУК4-11Б_2023_ЛР7_ВП/" && g++
main.cpp task_1.cpp task_2.cpp -o main &&
"/home/syricoff/bmstu/ВП/Суриков_НС_ИУК4-11Б_2023_ЛР7_ВП/"main
9
10 using namespace std;
11
12 void printMenu();
13
14 int main()
15 {
16     int choice;
17
18     do
19     {
20         printMenu();
21         cin >> choice;
22
23         switch (choice)
24         {
25             case 1:
26                 task_1();
27                 break;
28             case 2:
29                 task_2();
30                 break;
31             case 0:
32                 cout << "Выход из программы." << endl;
33                 break;
34             default:
35                 cout << "Некорректный выбор. Попробуйте еще раз." << endl;
36                 break;
37         }
38
39         cout << endl;
40
41     } while (choice != 0);
42 }
43
44 void printMenu()
45 {
46     cout << "Меню:" << endl;
47     cout << "1. Задание 1" << endl;
48     cout << "2. Задание 2" << endl;
49     cout << "0. Выход" << std::endl;
50     cout << "Выберите номер задания: ";
51 }
```


Результаты работы:

Первое задание:

```
Меню:
1. Задание 1
2. Задание 2
0. Выход
Выберите номер задания: 1
Данные записаны в файл: res/students.txt
Выберите дисциплину:
1. Математика
2. Физика
3. Химия
4. Биология
5. История
Введите номер предмета: 1
Введите путь к файлу для записи данных: res/math.txt
Данные записаны в файл: res/math.txt
```

students.txt

```
Ivanov 1 4.5 *- *- *
Petrov 2 3.8 - *- -
Sidorov 1 4.2 *****
Smith 3 3.5 - - - -
Johnson 2 4 ** - - *
Brown 1 3.9 - * - -
Lee 3 4.1 *- *- *
Kim 2 3.7 - *- -
Garcia 1 4.4 *****
Martinez 2 3.6 - - - -
Lopez 1 4.3 ** - - *
Liu 3 3.9 - * - -
Wang 2 4.2 *- *- *
Gonzalez 1 3.8 - *- -
Hernandez 3 4 *****
Taylor 2 3.5 - - - -
Anderson 1 4.1 ** - - *
Thomas 3 3.7 - *- -
Jackson 1 4.4 *****
White 2 3.6 - - - -
```

math.txt

```
Ivanov 1 4.5 *- *- *
Sidorov 1 4.2 *****
Johnson 2 4 ** - - *
Lee 3 4.1 *- *- *
Garcia 1 4.4 *****
Lopez 1 4.3 ** - - *
Wang 2 4.2 *- *- *
Hernandez 3 4 *****
Anderson 1 4.1 ** - - *
Jackson 1 4.4 *****
```

Второе задание:

```
Меню:
1. Задание 1
2. Задание 2
0. Выход
Выберите номер задания: 2
Введите имя первого файла: res/file1.txt
Введите имя второго файла: res/file2.txt
Файлы различаются в строке 11:
res/file1.txt: qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsfsghtrtyugfdfsghvjgdfg
res/file2.txt: qwertyjuytressdfghjhsddfgjkhkjgfdghjgfdsfsghtrtyugfdfsghvjgdfg
```

file1.txt

file2.txt

qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg
adsas	adsasd
qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf	qwertyuytressdfghjhsddfgjkhkjgfdghjgfdsf
sghtrtyugfdfsghvjgdfg	fsghtrtyugfdfsghvjgdfg

Вывод:

В ходе данной лабораторной работы я приобрёл практические навыки процедурного программирования средствами языка C++, познакомился с организацией файлов последовательного доступа, научился создавать объекты файловых потоков для связи с текстовым файлом на внешнем носителе, изучил основные программные средства для последовательной обработки файлов указанного типа, а также научился создавать пользовательское меню для демонстрации заданий лабораторной работы.

Литература

1. Курс лекций доцента кафедры ФН1-КФ Пчелинцевой Н.И.
2. Зырянов, К. И. Программирование на С++: учебное пособие / К. И. Зырянов, Н. П. Кисленко. — Новосибирск: Новосибирский государственный архитектурно-строительный университет (Сибстрин), ЭБС АСВ, 2017. — 129 с. — ISBN 978-5-7795-0817- 9. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <https://www.iprbookshop.ru/85873.html>
3. Фридман, А. Л. Язык программирования С++: учебное пособие / А. Л. Фридман. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 217 с. — ISBN 978-5-4497-0920-2. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <https://www.iprbookshop.ru/102076.html>