Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет»

Лабораторная работа №8

«Блоковый ввод-вывод»

Выполнил:

студент первого курса

ЭТФ группы РИС-23-3б

Коротаев Александр Дмитриевич

Проверила:

Доцент кафедры ИТАС О. А. Полякова

2024

**Цель**: Работа с двоичными файлами, организация ввода-вывода структурированной информации и ее хранение на внешних носителях.

**Постановка задачи**: сформировать двоичный файл из элементов, заданной в варианте структуры, распечатать его содержимое, выполнить удаление и добавление элементов в соответствии со своим вариантом, используя для поиска удаляемых или добавляемых элементов функцию. Формирование, печать, добавление и удаление элементов оформить в виде функций.

**Задача**:

Структура "Студент":

- фамилия, имя, отчество;

- дата рождения;

- домашний адрес;

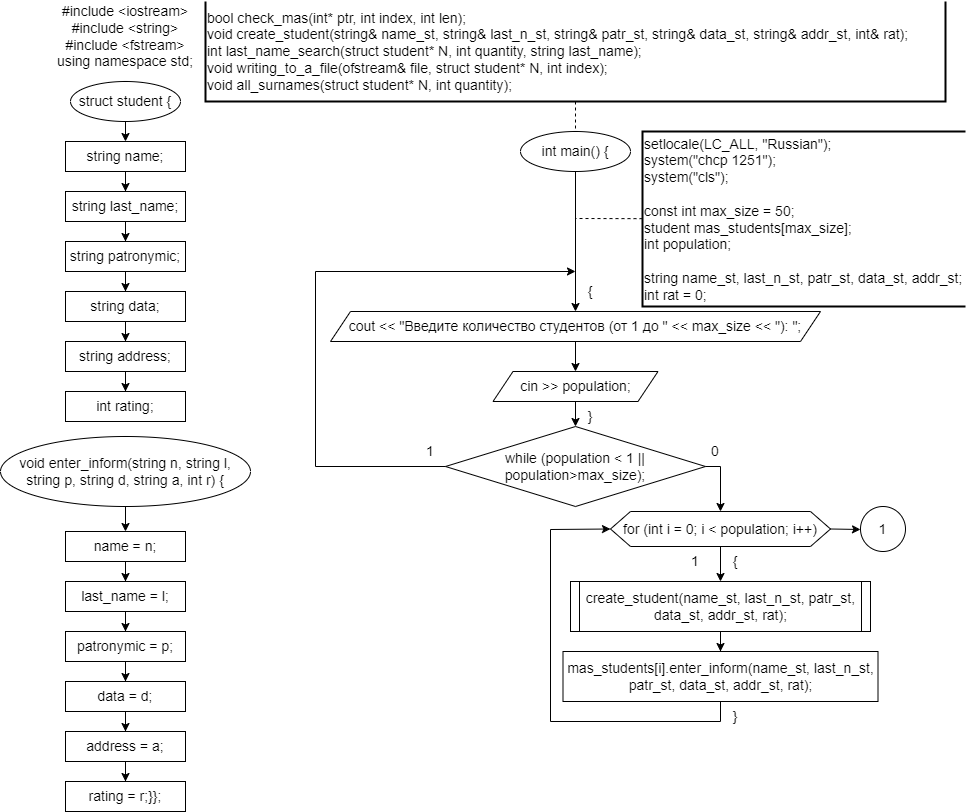
- рейтинг.

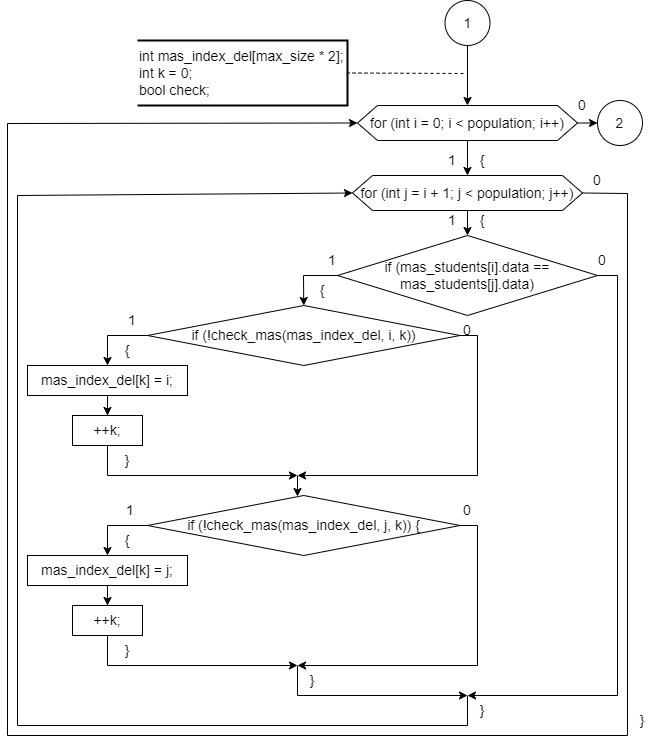
Удалить элементы, у которых даты рождения совпадают, добавить элемент перед элементом с заданной фамилией.

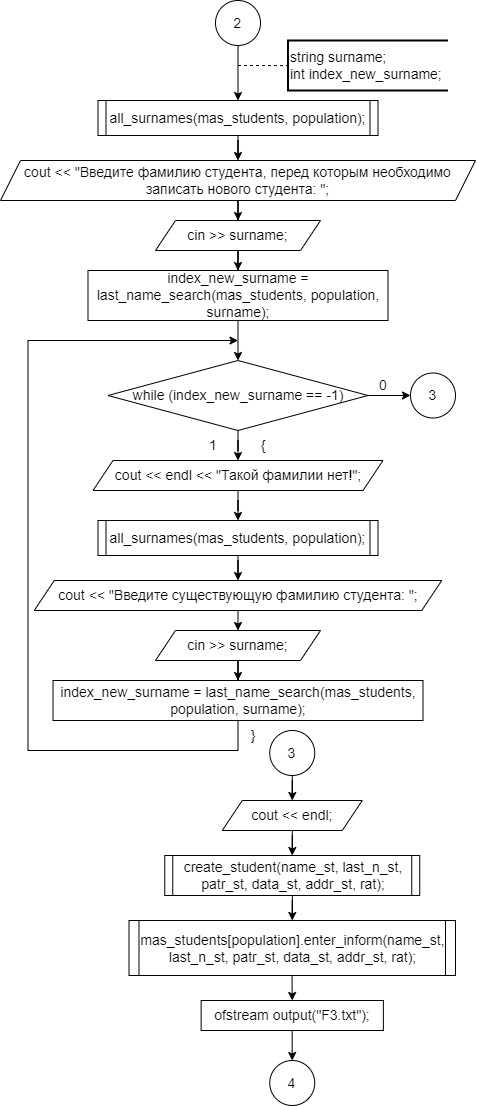
## Анализ задачи

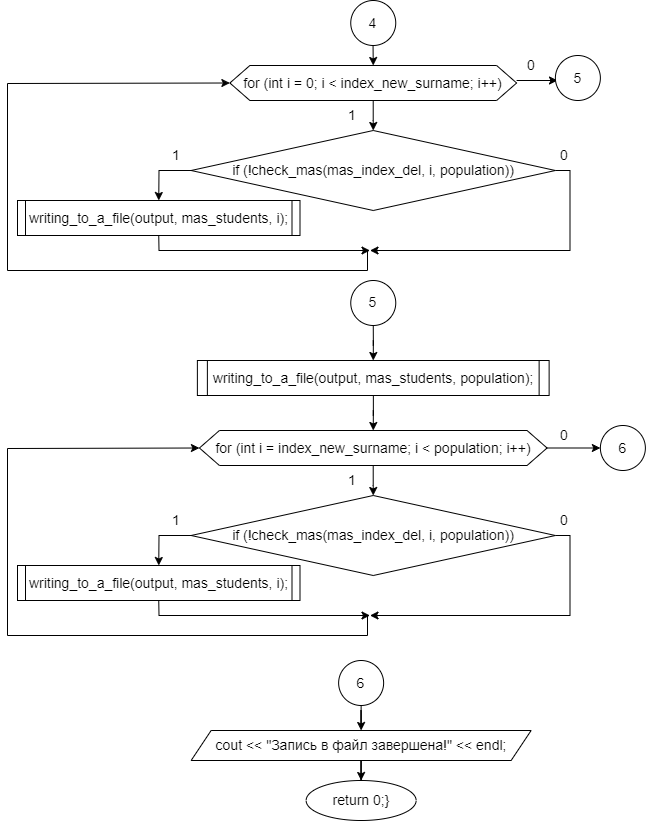
1. Для работы с данными студентов создаётся структура student, поля которой описывают ФИО, дату рождения, адрес и рейтинг студента.
2. Структура содержит метод для инициализации полей элемента.
3. Внутри функции main создаётся массив элементов структуры student, обозначается количество студентов. С помощью цикла их данные вводятся и с клавиатуры, используя отдельно написанную функцию.
4. Для хранения индексов элементов, которые нужно удалить из-за совпадения даты рождения, создаётся отдельный массив. Если даты рождения каких-либо студентов совпадают, их индексы помещаются в этот массив. В массиве нет одинаковых индексов. С помощью арифметического цикла в отдельной функции проверяется есть ли уже такой индекс в массиве. При необходимости индекс помещается в массив.
5. Для поиска студентов с одинаковыми датами рождения используются вложенные арифметические циклы. Внешний цикл – первый студент, внутренний цикл – второй студент.
6. В файл записываются только те студенты, индексы в массиве у которых не совпадают с запрещёнными, т.е. их даты рождения уникальны.
7. В файл, с помощью арифметического, цикла записываются сначала студенты, которые идут до новой фамилии. Далее добавляется в файл новый студент. Затем с помощью арифметического цикла записываются оставшиеся студенты.
8. С помощью функции в файл записываются только те студенты, индексы в массиве у которых не совпадают с запрещёнными, т.е. их даты рождения уникальны.
9. Функция сheck\_mas с помощью арифметического цикла проверяет наличие студента в массиве, где хранятся индексы студентов, которые не подходят по условию. Если студент найден в массиве, то функция возвращает значение true, иначе false.

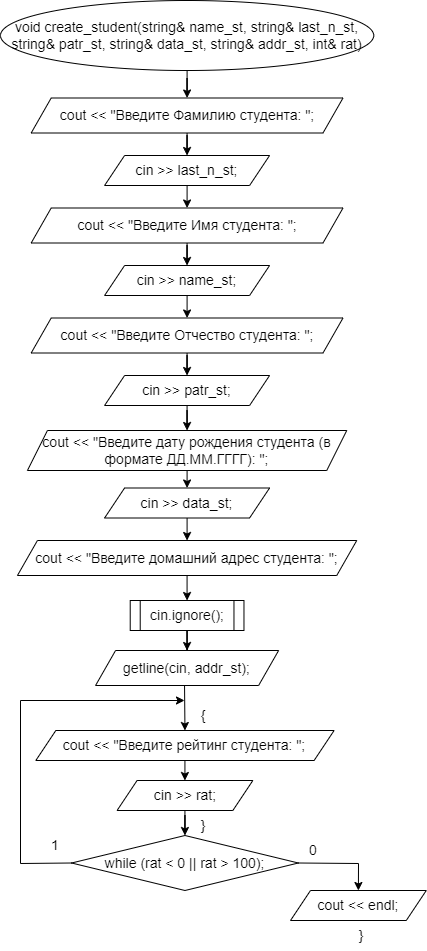
## Блок схема

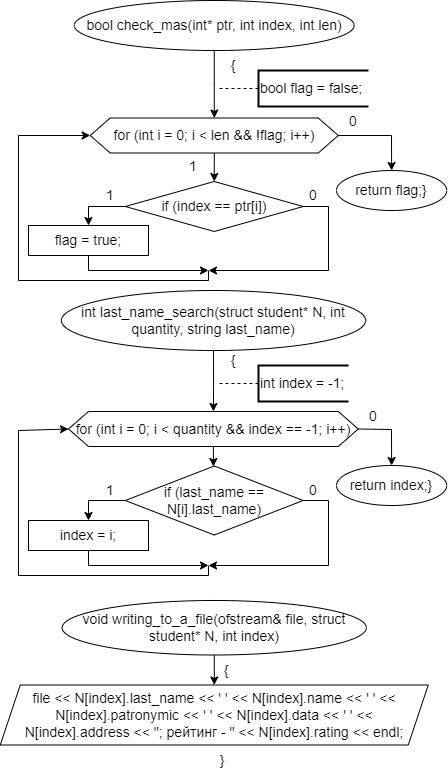


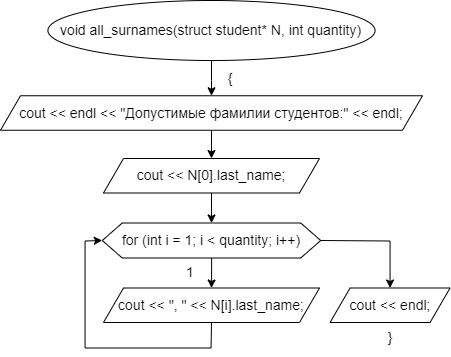












## Код программы

#include <iostream>

#include <string>

#include <vector>

#include <fstream>

using namespace std;

struct Student

{

string name, surname, patronymic, date, home;

int rate;

};

int n;

void info(vector <Student>& a, int i); //для заполнения студентами

void del\_student(vector <Student>& a, int i); //удаление студента

bool is\_date(Student s1, Student s2); //совпадают ли даты

void insert\_stud(vector <Student>& a, string surname); //вставить студента

void print\_stud(vector <Student>& a); //вывод в файл

int main()

{

system("chcp 1251");

system("cls");

vector <Student> students;

do

{

cout << "Введите кол-во студентов: "; cin >> n;

} while (n < 1);

for (int i = 0; i < n; i++) info(students, i);

for (int i = 0; i < n - 1; i++)

{

for (int j = i+1; j < n; j++)

{

if (is\_date(students[i], students[j]))

{

del\_student(students, i);

del\_student(students, j);

n -= 2;

}

}

}

string surname;

cout << "Введите фамилию, перед которой нужно вставить студента: "; cin >> surname;

insert\_stud(students, surname);

print\_stud(students);

return 0;

}

void info(vector <Student>& a, int i)

{

string name, surname, patronymic, date, home;

int rate;

if (i == -1) cout << "Информация о новом студенте." << endl;

else cout << "Информация о " << i + 1 << "-ом студенте." << endl;

cout << "Имя студента: "; cin.seekg(cin.eof()); getline(cin, name);

cout << "Фамилия студента: "; cin.seekg(cin.eof()); getline(cin, surname);

cout << "Отчество студента: "; cin.seekg(cin.eof()); getline(cin, patronymic);

cout << "Дата рождения студента (ДД.ММ.ГГГГ): "; cin.seekg(cin.eof()); getline(cin, date);

cout << "Адрес студента: "; cin.seekg(cin.eof()); getline(cin, home);

cout << "Рейтинг студента: "; cin >> rate;

cout << endl;

a.push\_back(Student{ name, surname, patronymic, date, home, rate });

}

void del\_student(vector <Student>& a, int i)

{

for (int j = i; j < n - 1; j++) a[j] = a[j + 1];

a.pop\_back();

}

bool is\_date(Student s1, Student s2)

{

if (s1.date == s2.date) return 1;

else return 0;

}

void insert\_stud(vector <Student>& a, string surname)

{

bool flag = false;

for (int i(0); i < n+1 && !flag; i++)

{

if (i == n) cout << "Фамилия не найдена";

else

{

if (a[i].surname == surname)

{

info(a, -1);

n++;

for (int j = n - 1; j != i; j--)

{

Student tmp = a.back();

a[j] = a[j - 1];

a[i] = tmp;

}

flag = true;

}

}

}

}

void print\_stud(vector <Student>& a)

{

ofstream output("F.txt");

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

{

output << "Информация о " << i + 1 << "-ом студенте." << endl;

output << "Имя студента: " << a[i].name << endl;

output << "Фамилия студента: " << a[i].surname << endl;

output << "Отчество студента: " << a[i].patronymic << endl;

output << "Дата рождения студента (ДД.ММ.ГГГГ): " << a[i].date << endl;

output << "Адрес студента: " << a[i].home << endl;

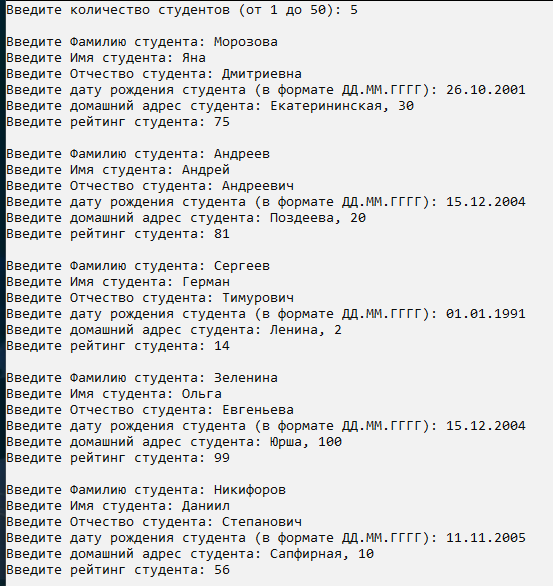
output << "Рейтинг студента: " << a[i].rate << endl << endl;

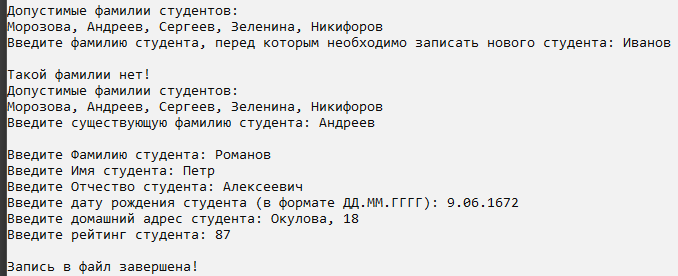
}

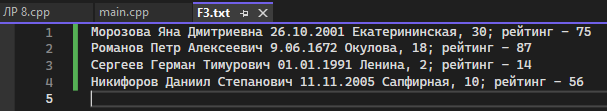
output.close();

}

## Результат работы программы







## GitHub

<https://github.com/Korovay4ik/Laboratory-works>