ФГБОУ ВО

Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет) «МГТУ им. Н.Э. Баумана»

Факультет ИУ «Информатика и системы управления»

Кафедра ИУ-3 «Информационные системы и телекоммуникации»

Отчет по лабораторной работе 21

**«Обработка исключительных ситуаций»**

По курсу «Информатика»

Выполнил: Еньшин Б. А.

Группа ИУ3-12

Вариант 9

Проверил: Видьманов Д. А.

Москва 2019

**Цель работы:**

Ознакомиться с определением исключительной ситуации и научиться обрабатывать исключительные ситуации. Закрепить полученные знания с в ходе выполнения поставленной задачи на языке программирования C++, попутно обрабатывая исключительные ситуации.

**Постановка задания:**

(Задач всего шесть. Девятому варианту соответствует задача №3)

№3

Вручную сформировать текстовый файл, содержащий фамилии студентов, количество экзаменов, которые сдавал каждый студент, и оценки, полученные на экзаменах. Программно создать второй файл для записи фамилий студентов, закончивших сессию без двоек, средний балл которых превышает средний балл по всей группе. При подсчете среднего балла, двойки во внимание не принимать.

**Ниже приведен код программы lab4\_source.cpp**

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

void write\_file(string path, char[10], short, short\*);

class My\_Exception

{public :

My\_Exception(string message)

{

cerr << message;

}

};

int main()

{

try {

char surname[10];

short student\_count;

string path = "students-info.txt";

string path2 = "students-best.txt";

bool flag;

cout << "Add info? "; cin >> flag;

if (flag) {

cout << "Enter number of students >> "; cin >> student\_count;

if (student\_count < 0) { throw My\_Exception("number cannot be negative"); }

short exams, \* res;

for (int i = 0; i < student\_count; i++) {

cin >> surname >> exams;

res = new short[exams];

for (int j = 0; j < exams; ++j) cin >> res[j];

write\_file(path, surname, exams, res);

delete[] res;

}

}

double average = 0;

int count = 0;

short count\_marks = 0;

fstream file;

file.open(path, ios::in);

if (file.fail()) { throw My\_Exception("Reading file error"); }

while (true)

{

file >> surname;

if (file.eof()) break;

file >> count;

for (int j = 0; j < count; j++)

{

short mark;

file >> mark;

if (mark > 2) { average += mark; count\_marks++; }

}

}

average /= count\_marks;

file.clear();

file.seekg(0);

bool f\_flag;

fstream resfile;

resfile.open(path2, ios::out);

if (resfile.fail()) { throw My\_Exception("Opening file for writing error"); }

resfile << "List of students above average. AVERAGE = " << average << endl;

while (true)

{

double individual = 0;

count\_marks = 0;

f\_flag = false;

file >> surname;

if (file.eof()) break;

file >> count;

for (int j = 0; j < count; j++)

{

short mark;

file >> mark;

if (mark > 2) { individual += mark; count\_marks++; }

else f\_flag = true;

}

if ((individual / count\_marks > average) && (!f\_flag)) { resfile << surname << endl; }

}

file.close();

resfile.close();

}

catch (My\_Exception & ex) { cout << endl << "Program ended with an exception"; }

return 0;

}

void write\_file(string path, char name[10], short exams, short\* res) {

fstream file;

file.open(path, ios::app);

if (file.fail()) { throw My\_Exception("Opening file for writing error"); }

file << name <<" "<< exams;

for (int i = 0; i < exams;++i)file <<" "<< res[i] ;

file << endl;

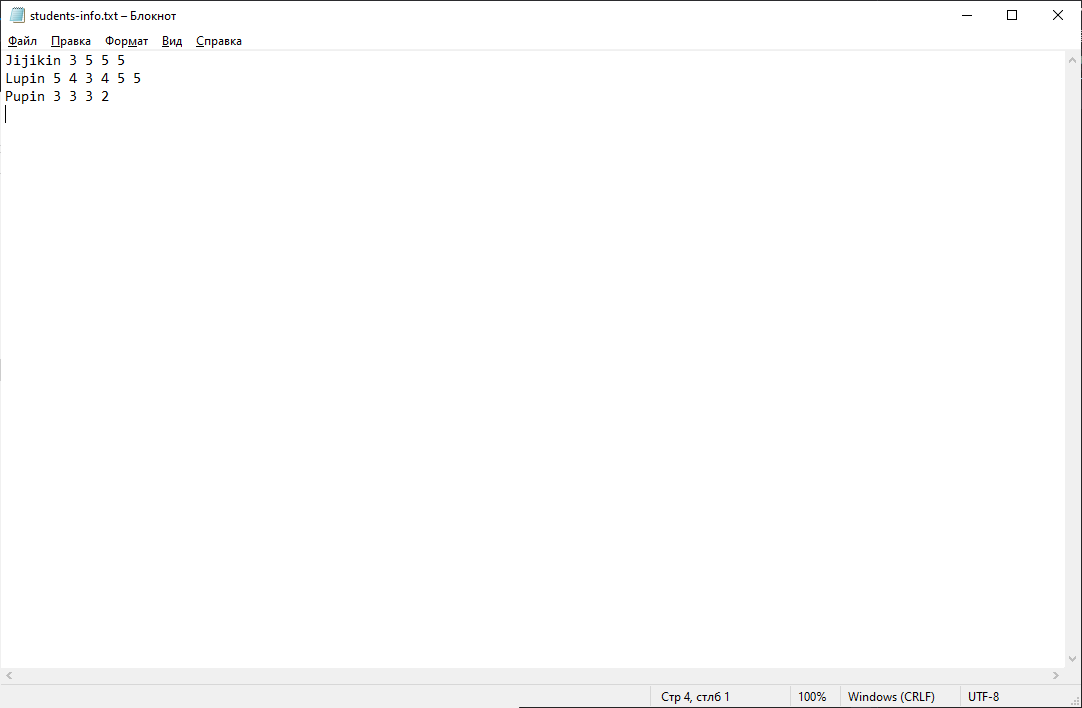
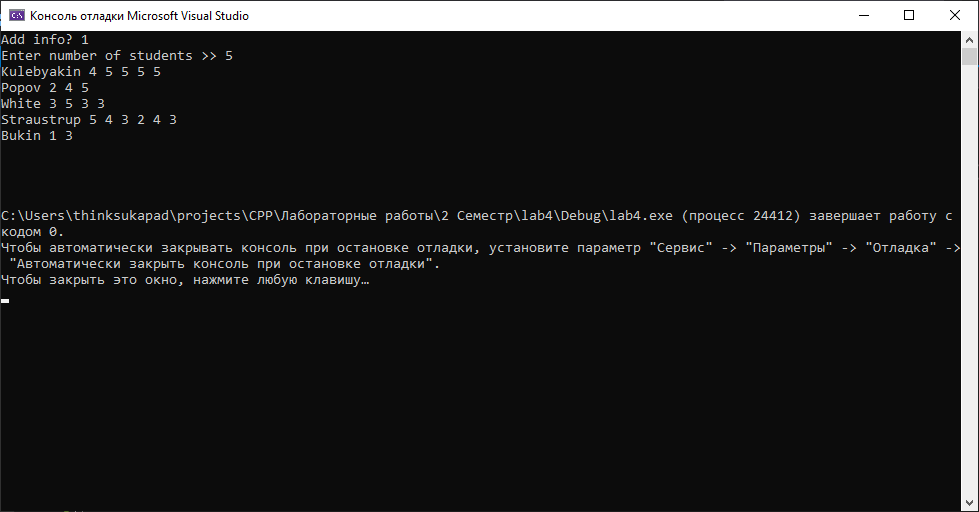
file.close();

}

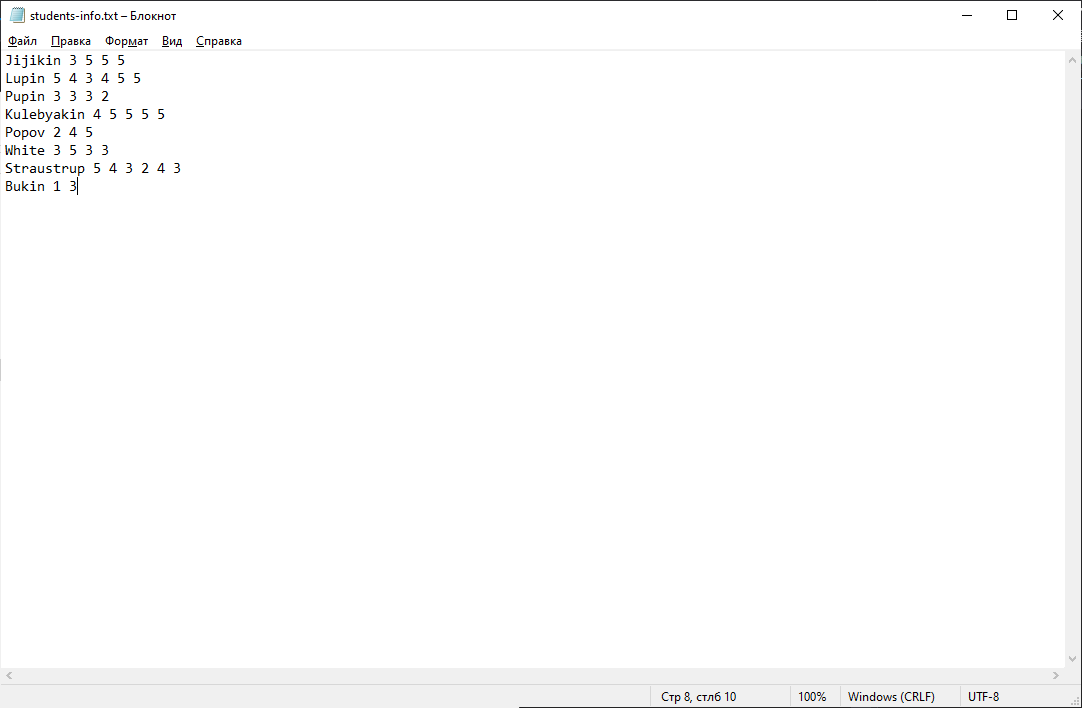
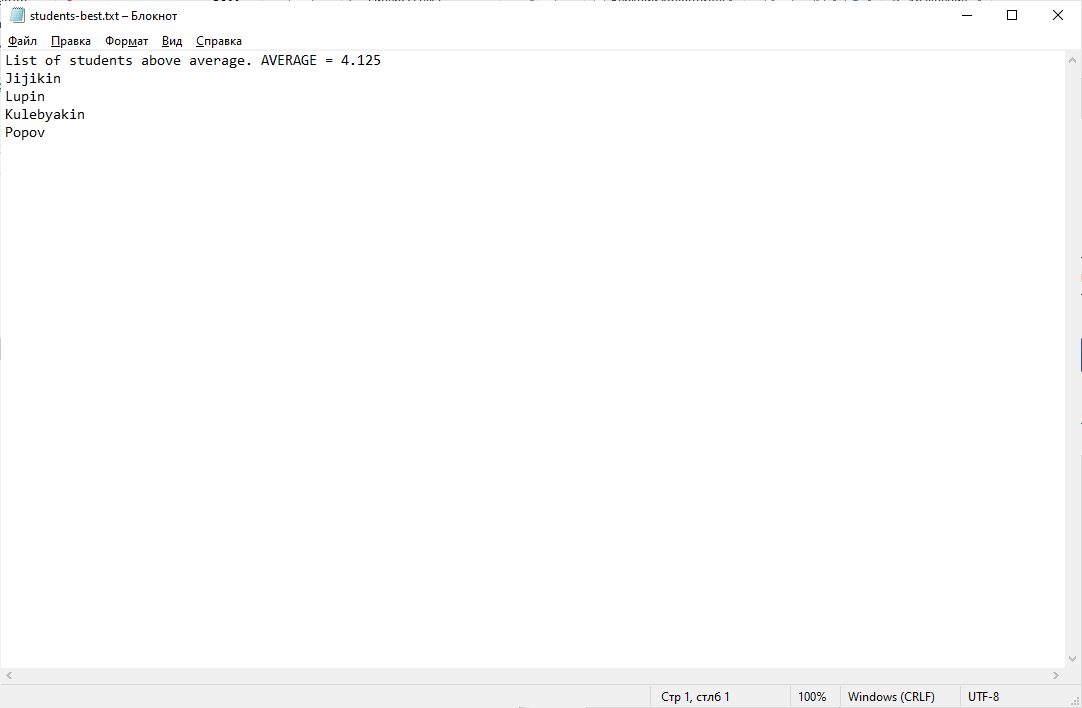
**Скриншот консоли вывода программы**

1. Успешное выполнение программы

Стартовые данные Ввод в консоль

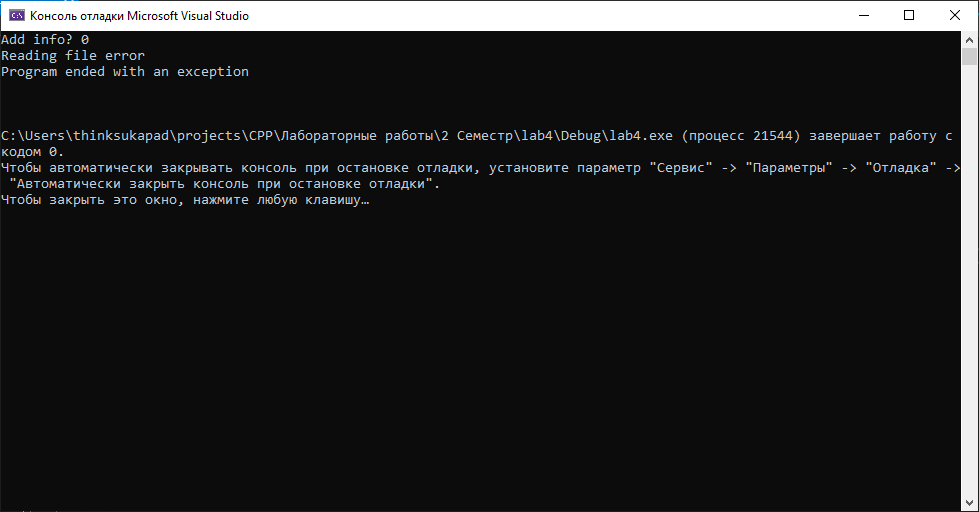
 

Список после выполнения программы Список успешных студентов

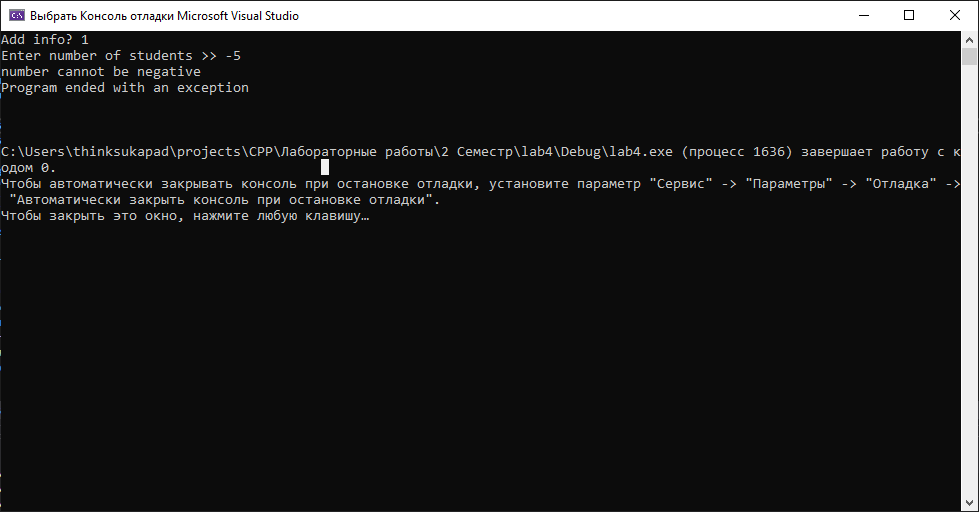
 

1. Обработка исключений

Ошибка чтения файла (например, неверно указанный путь)



Выход за пределы значений



**Вывод:**

В ходе данной лабораторной работы я освоил основную информацию о исключительных ситуациях. Я узнал о том, как выявлять и генерировать исключительные ситуации с помощью операторов try/catch, throw и научился создавать свой класс исключительных ситуаций. Полученные знания я применил для написания программы записи в файл и чтения из файла с выявлением исключительных ситуаций на языке программирования C++.