Учреждение образования Федерации профсоюзов Беларуси

«Международный университет «МИТСО»

Рег. № \_\_\_\_\_\_\_\_\_\_ Кафедра информационных систем и технологий

Дата \_\_\_\_\_\_\_\_\_2020

**КУРСОВАЯ РАБОТА**

на тему \_ Учет клиентов музыкальной студии

по дисциплине \_Название\_\_Основы конструирования программ\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Основные замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отметка о допуске курсовой работы к  защите:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_  Подпись научного руководителя:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Студент: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  (Ф.И.О. – полностью)  Курс \_\_1\_\_, группа \_\_\_2020\_\_\_\_  Факультет \_\_экономический\_\_\_  Специальность \_\_Информационные системы и технологии\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Научный руководитель:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (должность, ученая степень, ученое звание)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О. – полностью) |

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc72055182)

[1.ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 4](#_Toc72055183)

[2.ТРЕБОВАНИЕ К ПРОГРАММЕ 5](#_Toc72055184)

[3.КОНСТРУИРОВАНИЕ ПРОГРАММЫ 6](#_Toc72055185)

[4.ОПИСАНИЕ ИНТЕРФЕЙСА 7](#_Toc72055186)

[5.ВЫБОР СПОСОБА ОРГАНИЗАЦИИ ДАННЫХ 9](#_Toc72055187)

[6.РАЗРАБОТКА ПЕРЕЧНЯ ПОЛЬЗОВАТЕЛЬСКИХ ФУНКЦИЙ ПРОГРАММЫ 10](#_Toc72055188)

[7.ПРОГРАММНАЯ ЧАСТЬ 16](#_Toc72055191)

[ЗАКЛЮЧЕНИЕ 17](#_Toc72055192)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 17](#_Toc72055193)

[ПРИЛОЖЕНИЕ 19](#_Toc72055194)

# **ВВЕДЕНИЕ**

Была поставлена задача разработать программный модуль «Учёт успеваемости студентов», предназначенный для использования деканата. В базе содержатся сведения о студентах (фамилия, имя, отчество, номер группы, оценки по предметам). При возникновения нового студента его данные заноситься в базу.

Программа создана для систематизации студентов, быстрого просмотра/изменения и добавления новых студентов.

Цель разработки обеспечить работы деканата быстро и максимально эффективно, удовлетворять запросы работников деканата, а также должен быть возможность пополнять базу, для оптимизации процессов университета. И неотъемлемой частью данной программы должна высокая оптимизация, чтобы пользователь мог запустить даже на самом слабом устройстве.

Задачи данного программного продукта: быстро и гибко редактироваться со стороны владельцев музыкальной студии. С точки зрения использования пользователя программа должна иметь красивый, понятный и интуитивный интерфейс чтобы было владельцу приятно пользоваться программой. Так же требуется предусмотреть возможность добавления новых функций в программе.

Для разработки программы был использован текстовый файл и язык программирования C++ разработанный в среде Visual Studio компаний Microsoft.

Visual Studio-линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой Windows Forms, а также веб-сайты, веб-приложения, веб-службы, как в родном, так и в управляемом кодах для всех платформ, поддерживающих Windows, Windows Mobile, Windows Ce, .Net Framework, Xbox, Windows Phone .Net Compact Framework и Silverling.

# **1.ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ**

Была поставлена задача разработать программный модуль «Учёт успеваемости студентов», предназначенный для использования деканата. В базе содержатся сведения о студентах (фамилия, имя, отчество, номер группы, оценки по предметам). При возникновения нового студента его данные заноситься в базу.

Спрос на высоко квалифицированных кадров привело к росту учебных заведений и как следствие студентов. Из-за этого и возникла потребность в данном программном продукте. Поэтому разрабатываемая программа должна выдержать весьма серьёзную конкуренцию.

Функции программного продукта: содержатся сведения о студентах (фамилия, имя, отчество, номер группы, оценки по предметам) должны иметь понятные данные и возможность обрабатывать их (добавлять, просматривать, изменять).

Данный программный продукт выгодно отличается от других подобных тем что, пользуясь им, пользователь может не только хранить данные студентов, но и вычислять средний бал студентов и сортировать по среднему балу.

# **2.ТРЕБОВАНИЕ К ПРОГРАММЕ**

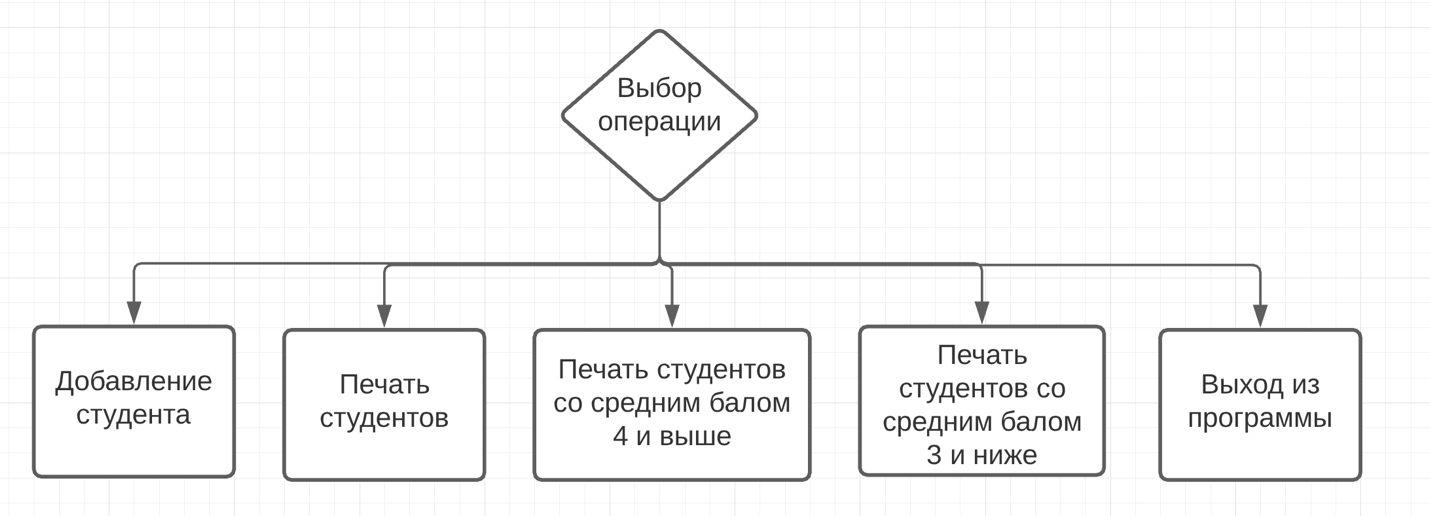
Исходя из задач, описанных выше главными требованиями к языку, на котором будет написана программы — это тонкая настройка и возможность использованием минимум вычислительных ресурсов добиваться максимум результата.

Для выполнения данных задач очень подходит такой язык программирования как с++. Так как его из-за того, что он является весьма низкоуровневым языком программирования, а значит он больше других приближен к машинному коду, из-за этого с++ дает возможность оптимизировать проект так что даже на очень старых устройствах он будет работать.

Windows 10 – операционная система для персональных компьютеров и рабочих станций, разработанная корпорацией Microsoft в рамках семейства Windows NT. После Windows 8.1 система получила номер 10, минуя 9. Серверный аналог Windows 10 – Windows Server 2016. Система призвана стать единой для разных устройств, таких как персональные компьютеры, планшеты, смартфоны, консоли Xbox One и пр. Доступна единая платформа разработки и единый магазин универсальных приложений, совместимых со всеми поддерживаемыми устройствами. Windows 10 поставляется в качестве услуги с выпуском обновлений на протяжении всего цикла поддержки. В течение первого года после выхода системы пользователи могли бесплатно обновиться до Windows 10 на устройствах под управлением лицензионных копий Windows 7, Windows 8.1 и Windows Phone 8.1. Среди значимых нововведений – голосовая помощница Кортана, возможность создания и переключения нескольких рабочих столов и др. Windows 10 – последняя «коробочная» версия Windows, все последующие версии будут распространяться исключительно в цифровом виде.

Среда разработки Microsoft Visual Studio. Вид приложения – консольное. Программа построена на базе объектно-ориентированного программирования. Способ организации данных: структуры. Каждая логическая завершенная данных подзадача программы реализована в виде методов. К защите курсовой предоставляется: консольное приложение и пояснительная записка.

# **3.КОНСТРУИРОВАНИЕ ПРОГРАММЫ**



В текущей главе представлено описание алгоритма, который имеет начало алгоритма и конец, внутри алгоритма имеются циклы и иные операций.

Первым пунктом идёт выбор операции, за счет введение цифры от 1 до 6 и последующей за вводом проверки.

Следующий шаг — это работа непосредственно с программой в зависимости от выбранных данных. Добавление данных, вывод данных,изменение данных.

# **4.ОПИСАНИЕ ИНТЕРФЕЙСА**

При старте программы происходит выбор желаемой операции (рис.4.1)

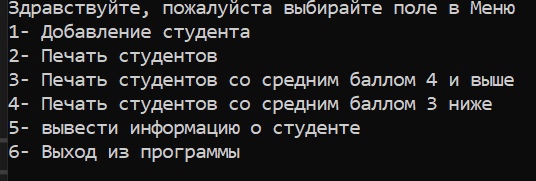


Рис. 4.1

При выборе первого пункта в консоль необходимо ввести данные студента, последовательно (рис. 4.2).

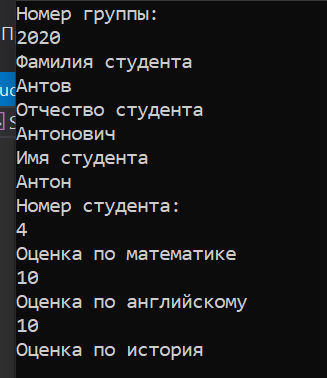


Рис.4.2

При выборе второго пункта происходит вывод всех студентов на консоль (рис 4.3).

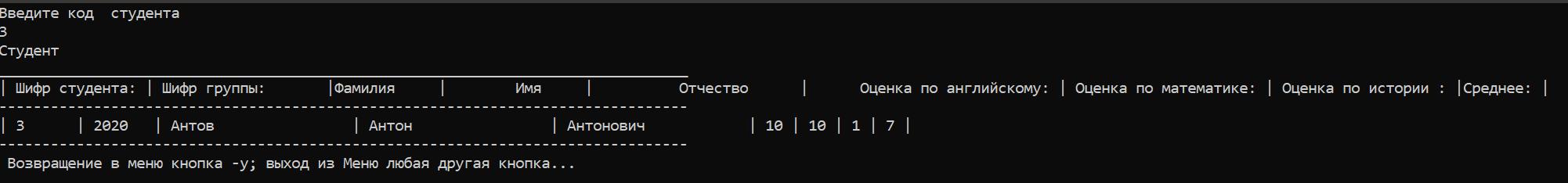
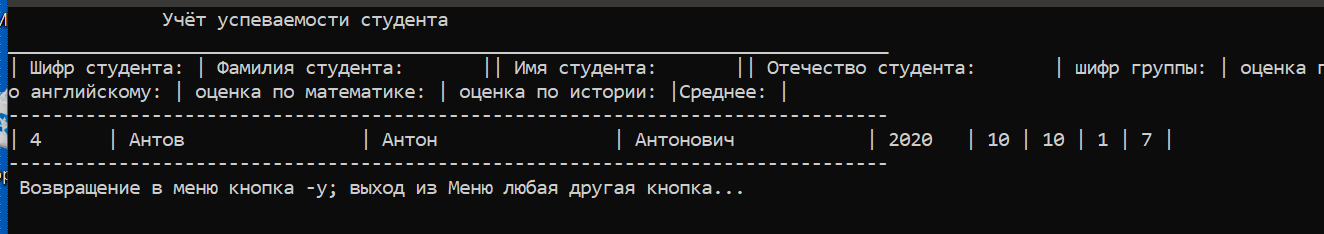
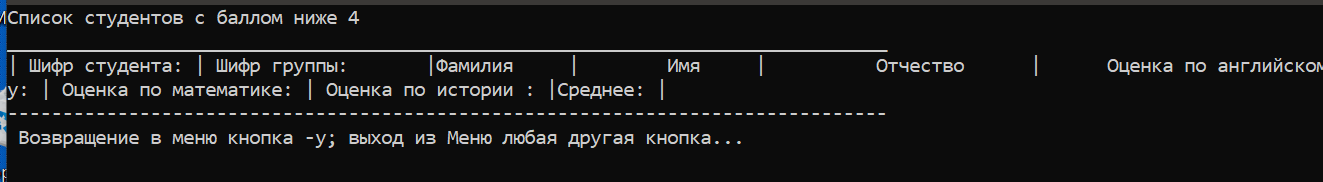


Рис. 4.3

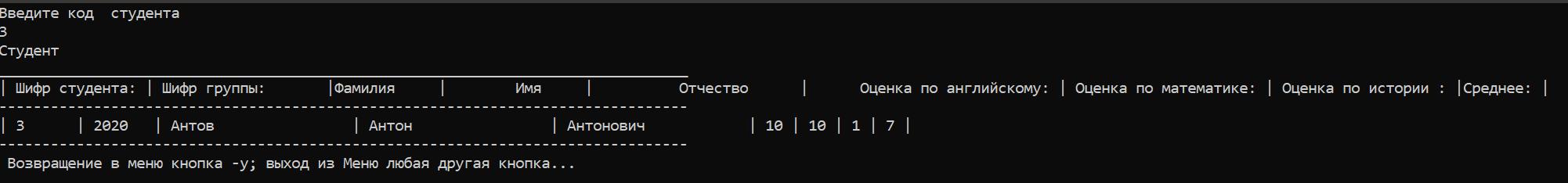
При выборе третьего пункта происходит вывод всех студентов со средним балом 4 и выше на консоль (рис 4.4).

Рис.4.4

При выборе третьего пункта происходит вывод всех студентов со средним балом 3 и ниже на консоль (рис 4.5).

Рис 4.5

При выборе пятого пункта на консоль выводиться данные определённого студента. Путём ввода номера студента на консоль и последующего вывода студента на консоль (рис 4.6).

Рис 4.6

При выборе шестого пункта происходит завершение программы.

# **5.ВЫБОР СПОСОБА ОРГАНИЗАЦИИ ДАННЫХ**

Структура **OT** предназначена для хранения данных студента, его ФИО и отметок. Данная структура состоит из 10 переменных:

int shifr\_studentа – номер группы.

char imja[20 + 1] – имя студента.

char familia[20 + 1] – фамилия студента.

char otchestvo[20 + 1] – отчество студента.

char facultet[10 + 1] – факультет студента.

int shifr\_gruppi - номер группы.

int matematika – оценка по математике.

int angl – оценка по английскому.

int history – оценка по истории.

int count – количество студентов.

# **6.РАЗРАБОТКА ПЕРЕЧНЯ ПОЛЬЗОВАТЕЛЬСКИХ ФУНКЦИЙ ПРОГРАММЫ**

bool **add**(int& count, OT\* ot); - предназначена для добавление студента в программу (рис 6.1).

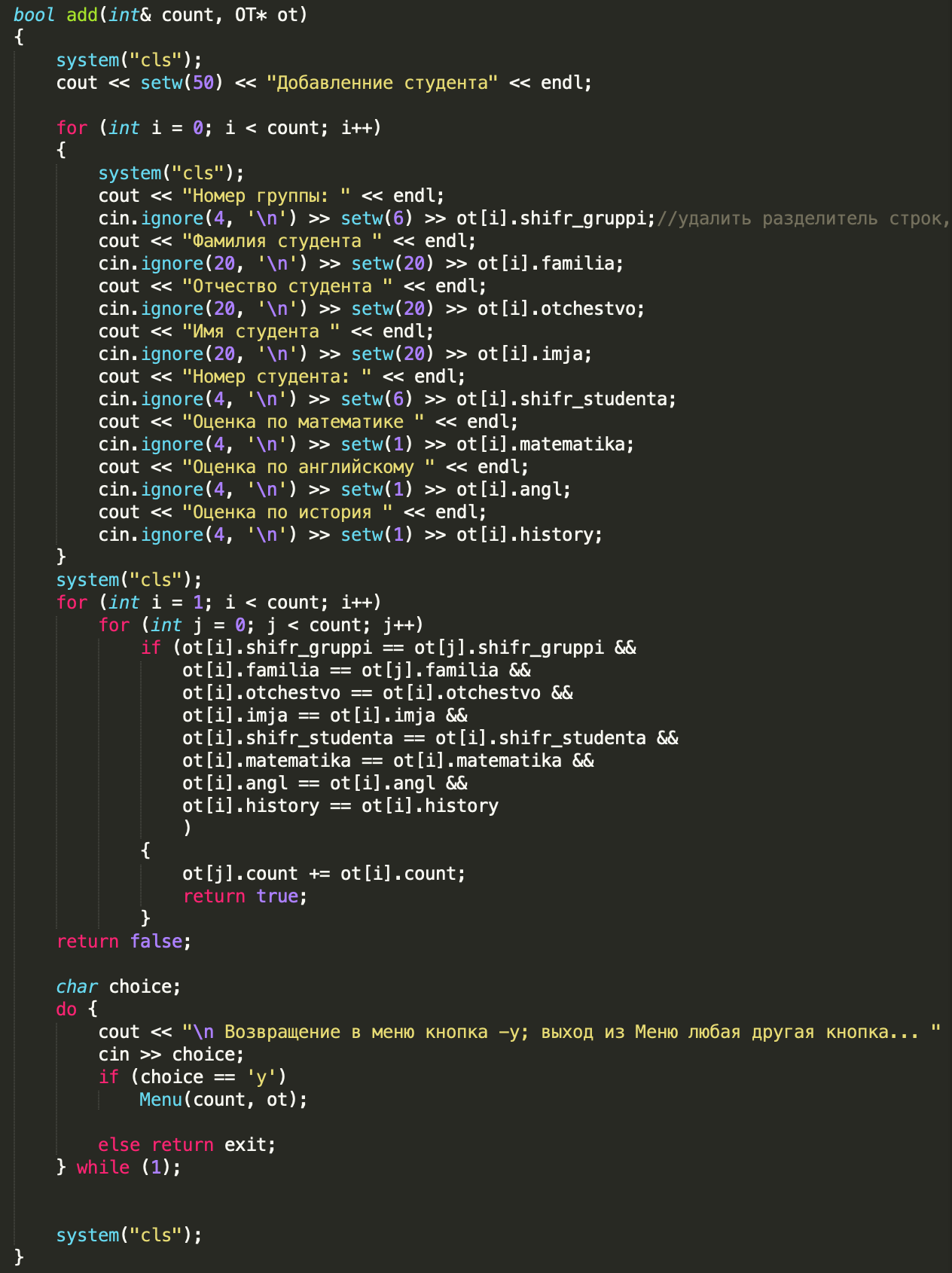


рис 6.1

Функции void **drawLineW**() и void **drawLineN**() – предназначены для вывода меню (рис 6.2)

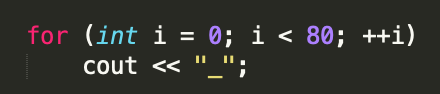


рис 6.2

void **print**(int count, OT\* ot); - предназначена для информирование данных студентов на консоль (рис 6.3).



Рис 6.3

void **Menu** (int count, OT\* ot); - основная функция в данном приложение. С её помощью происходит вывод всех возможных операций на консоль. И при помощи условного оператора Switch происходит выбор необходимой операции (рис 6.4).

# 

# Рис 6.4

void **horosho**(int count, OT\* ot) – данная функция выводит студентов, у которых средний бал 4 и выше (рис 6.5).



Рис 6.5

void **ploho**(int count, OT\* ot) - данная функция выводит студентов, у которых средний бал 3 и ниже (рис 6.6).

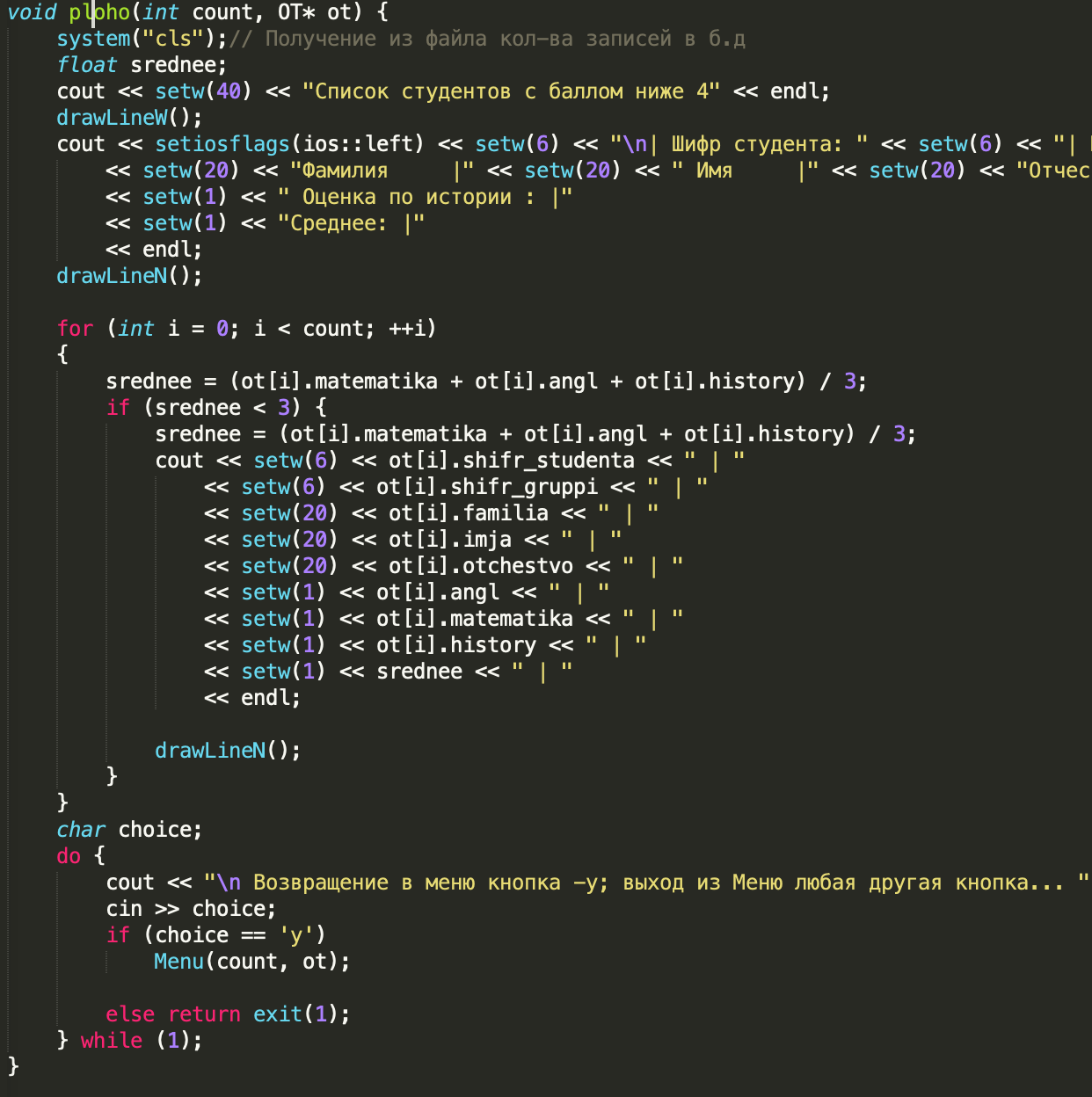


Рис 6.6

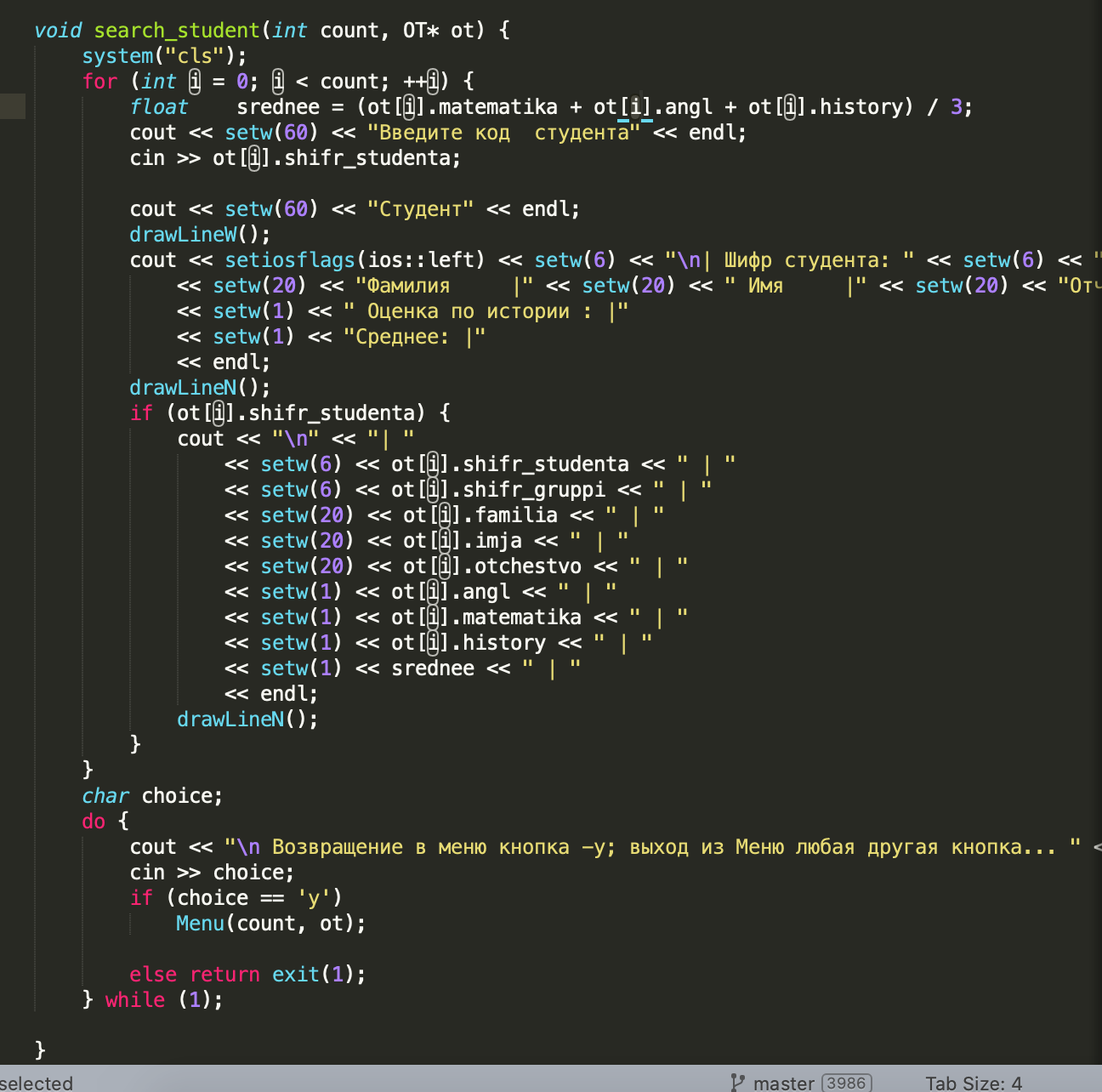
void **search\_student** (int count, OT\* ot) – данная функция выводит номер определённого студента по коду студента (рис 6.7).

Рис 6.7

# **7.ПРОГРАММНАЯ ЧАСТЬ**

Вначале происходит выбор операции:

1. Добавление студента
2. Печать студентов
3. Печать студентов со средним баллом 4 и выше
4. Печать студентов со средним баллом 3 ниже
5. Вывести информацию о студенте
6. Выход из программы

Далее при помощи конструкции switch происходит переход к выбранной операции.

1: Происходит вызов функции add. Вначале происходит в ввод данных структуры из консоли. А затем присвоение введённых данных структуры на консоль.

2: Происходит вызов функции print. Происходит перебор массива структур и вывод их на консоль.

3: Происходит вызов функции horosho. Вначале происходит проверка каждого студента (является ли средний бал студента 4 и выше). В случае если проверка приходит успешно, то выводиться данные студента на консоль.

4: Происходит вызов функции ploho. Вначале происходит проверка каждого студента (является ли средний бал студента 3 и ниже). В случае если проверка приходит успешно, то выводиться данные студента на консоль.

5: Происходит вызов функции search\_student. Вначале приходит ввод пользователем кода студента, а далее вывод указанного студента.

6: Происходит окончание работы программы.

# **ЗАКЛЮЧЕНИЕ**

Исходя из поставленных целей и задач было сконструировано, разработана и протестирован программный модуль «Учёт успеваемости студентов», предназначенный для использования деканата. В базе содержатся сведения о студентах (фамилия, имя, отчество, номер группы, оценки по предметам). При возникновения нового студента его данные заноситься в базу.

1)Быстро и гибко редактироваться со стороны сотрудников деканата.

2)Программный продукт имеет красивый, понятный и интуитивный интерфейс чтобы пользователю было приятно пользоваться программой.

3)Так же предусмотрено возможность добавления новых функций в программе.

Цель была выполнены, картотека имеет следующие преимущества:

1)Работа деканата происходит быстро и максимально эффективно.

2)Удовлетворяет запросы деканата.

2)Также реализована возможность пополнять базу, для оптимизации процессов университета.

3)Реализована высокая оптимизация, что-то привело к возможности пользователем запускать приложение даже на самом слабом устройстве.

Данный программный продукт выгодно отличается от других подобных тем что, пользуясь им, пользователь может не только хранить данные студентов, но и производить выборку студентов по среднему балу.

Таким образом программа работает без сбоев и все задачи были выполнены.

# **СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирование / Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. – СПб: Питер, 2015. – 368с.
2. Коплиен Дж. Мультипарадигменное проектирование для С++ / Коплиен Дж. – Питер, 2005.
3. Обзор обновлений и новых функций Windows 10 [Электронный ресурс]. – Microsoft, 2020. – Режим доступа: https://www.microsoft.com/ru-ru/windows/features. – Дата доступа: 22.05.2020.
4. Объектно-ориентированный анализ и проектирование с примерами приложений / Гради Буч [и др.]. – 3-е изд. – М.: ООО «И.Д. Вильямс», 2008. – 720 с.
5. Стивен Прата Язык программирования C++ / Стивен Прата – Вильямс, 2012.
6. Страуструп Б. Программирование. Принципы и практика с использованием C++ / Страуструп Б. – 2-е изд. – Вильямс, 2016.
7. Model-View-Presenter [Электронный ресурс]. – Википедия, 2020. – Режим доступа: https://ru.wikipedia.org/wiki/Model-View-Presenter. – Дата доступа: 22.04.2020.
8. SFML [Электронный ресурс]. – SFML, 2020. – Режим доступа: https://www.sfml-dev.org. – Дата доступа: 22.04.2020.
9. SFML [Электронный ресурс]. – Википедия, 2020. – Режим доступа: https://ru.wikipedia.org/wiki/SFML. – Дата доступа: 22.04.2020.

# **ПРИЛОЖЕНИЕ**

#include<iostream>

#include<string>

#include<conio.h>

#include<fstream>

#include<iomanip>//Для пробелов между словами и символами

#include<Windows.h>

#define N 1

using namespace std;

struct OT

{

int shifr\_studenta;

char imja[20 + 1];

char familia[20 + 1];

char otchestvo[20 + 1];

char facultet[10 + 1];

int shifr\_gruppi;

int matematika;

int angl;

int history;

int count;

};

bool add(int& count, OT\* ot);

void drawLineW();

void drawLineN();

void print(int count, OT\* ot);

void Menu(int count, OT\* ot);

void horosho(int count, OT\* ot);

void ploho(int count, OT\* ot);

void search\_student(int count, OT\* ot);

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("cls");

int count = N;

OT\* ot = new OT[count];

Menu(count, ot);

\_getch();

return 0;

}

void Menu(int count, OT\* ot)

{

system("cls");

int numberSearch;

do

{

cout << "Здравствуйте, пожалуйста выбирайте поле в Меню" << endl;

cout << "1- Добавление студента " << endl;

cout << "2- Печать студентов" << endl;

cout << "3- Печать студентов со средним баллом 4 и выше" << endl;

cout << "4- Печать студентов со средним баллом 3 ниже " << endl;

cout << "5- вывести информацию о студенте" << endl;

cout << "6- Выход из программы" << endl;

cin >> numberSearch;

switch (numberSearch)

{

case 1: add(count, ot); break;

case 2: print(count, ot); break;

case 3: horosho(count, ot); break;

case 4: ploho(count, ot); break;

case 5: search\_student(count, ot); break;

case 6: exit(0); break;

}

} while (numberSearch != 6);

system("cls");

}

bool add(int& count, OT\* ot)

{

system("cls");

cout << setw(50) << "Добавленние студента" << endl;

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

{

system("cls");

cout << "Номер группы: " << endl;

cin.ignore(4, '\n') >> setw(6) >> ot[i].shifr\_gruppi;//удалить разделитель строк,// пропускает указанное число элементов от текущей позиции чтения

cout << "Фамилия студента " << endl;

cin.ignore(20, '\n') >> setw(20) >> ot[i].familia;

cout << "Отчество студента " << endl;

cin.ignore(20, '\n') >> setw(20) >> ot[i].otchestvo;

cout << "Имя студента " << endl;

cin.ignore(20, '\n') >> setw(20) >> ot[i].imja;

cout << "Номер студента: " << endl;

cin.ignore(4, '\n') >> setw(6) >> ot[i].shifr\_studenta;

cout << "Оценка по математике " << endl;

cin.ignore(4, '\n') >> setw(1) >> ot[i].matematika;

cout << "Оценка по английскому " << endl;

cin.ignore(4, '\n') >> setw(1) >> ot[i].angl;

cout << "Оценка по история " << endl;

cin.ignore(4, '\n') >> setw(1) >> ot[i].history;

}

system("cls");

for (int i = 1; i < count; i++)

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

if (ot[i].shifr\_gruppi == ot[j].shifr\_gruppi &&

ot[i].familia == ot[j].familia &&

ot[i].otchestvo == ot[i].otchestvo &&

ot[i].imja == ot[i].imja &&

ot[i].shifr\_studenta == ot[i].shifr\_studenta &&

ot[i].matematika == ot[i].matematika &&

ot[i].angl == ot[i].angl &&

ot[i].history == ot[i].history

)

{

ot[j].count += ot[i].count;

return true;

}

return false;

char choice;

do {

cout << "\n Возвращение в меню кнопка -y; выход из Меню любая другая кнопка... " << endl;

cin >> choice;

if (choice == 'y')

Menu(count, ot);

else return exit;

} while (1);

system("cls");

}

void drawLineW()

{

for (int i = 0; i < 80; ++i)

cout << "\_";

}

void drawLineN()

{

for (int i = 0; i < 80; ++i)

cout << "-";

}

void print(int count, OT\* ot)

{

system("cls");

setlocale(LC\_ALL, "Russian");

cout << setw(40) << "Учёт успеваемости студента" << endl;

drawLineW();

cout << setiosflags(ios::left) << setw(6) << "\n| Шифр студента: " << setw(20) << "| Фамилия студента: |"

<< setw(20) << "| Имя студента: |"

<< setw(20) << "| Отечество студента: |" << setw(6) << " шифр группы: |" << setw(1) << " оценка по английскому: |"

<< setw(1) << " оценка по математике: |" << setw(1) << " оценка по истории: |"

<< setw(1) << "Среднее: |" << endl;

drawLineN();

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

{

float srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

cout << setiosflags(std::ios::left) << "\n" << "| "

<< setw(6) << ot[i].shifr\_studenta << " | "

<< setw(20) << ot[i].familia << " | "

<< setw(20) << ot[i].imja << " | "

<< setw(20) << ot[i].otchestvo << " | "

<< setw(6) << ot[i].shifr\_gruppi << " | "

<< setw(1) << ot[i].angl << " | "

<< setw(1) << ot[i].matematika << " | "

<< setw(1) << ot[i].history << " | "

<< setw(1) << srednee << " | "

<< endl;

drawLineN();

}

char choice;

do {

cout << "\n Возвращение в меню кнопка -y; выход из Меню любая другая кнопка... " << endl;

cin >> choice;

if (choice == 'y')

Menu(count, ot);

else return exit(1);

} while (1);

}

void horosho(int count, OT\* ot)

{

system("cls");// Получение из файла кол-ва записей в б.д.

float srednee;

cout << setw(40) << "Список студентов с баллом 4 и выше" << endl;

drawLineW();

cout << setiosflags(ios::left) << setw(6) << "\n| Шифр студента: " << setw(6) << "| Шифр группы: |"

<< setw(20) << "Фамилия |" << setw(20) << " Имя |" << setw(20) << "Отчество |" << setw(1) << " Оценка по английскому: |" << setw(1) << " Оценка по математике: |"

<< setw(1) << " Оценка по истории : |"

<< setw(1) << "Среднее: |"

<< endl;

drawLineN();

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

{

srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

if (srednee >= 4) {

srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

cout << setw(6) << ot[i].shifr\_studenta << " | "

<< setw(6) << ot[i].shifr\_gruppi << " | "

<< setw(20) << ot[i].familia << " | "

<< setw(20) << ot[i].imja << " | "

<< setw(20) << ot[i].otchestvo << " | "

<< setw(1) << ot[i].angl << " | "

<< setw(1) << ot[i].matematika << " | "

<< setw(1) << ot[i].history << " | "

<< setw(1) << srednee << " | "

<< endl;

drawLineN();

}

}

char choice;

do {

cout << "\n Возвращение в меню кнопка -y; выход из Меню любая другая кнопка... " << endl;

cin >> choice;

if (choice == 'y')

Menu(count, ot);

else return exit(1);

} while (1);

}

void ploho(int count, OT\* ot) {

system("cls");// Получение из файла кол-ва записей в б.д

float srednee;

cout << setw(40) << "Список студентов с баллом ниже 4" << endl;

drawLineW();

cout << setiosflags(ios::left) << setw(6) << "\n| Шифр студента: " << setw(6) << "| Шифр группы: |"

<< setw(20) << "Фамилия |" << setw(20) << " Имя |" << setw(20) << "Отчество |" << setw(1) << " Оценка по английскому: |" << setw(1) << " Оценка по математике: |"

<< setw(1) << " Оценка по истории : |"

<< setw(1) << "Среднее: |"

<< endl;

drawLineN();

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

{

srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

if (srednee < 3) {

srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

cout << setw(6) << ot[i].shifr\_studenta << " | "

<< setw(6) << ot[i].shifr\_gruppi << " | "

<< setw(20) << ot[i].familia << " | "

<< setw(20) << ot[i].imja << " | "

<< setw(20) << ot[i].otchestvo << " | "

<< setw(1) << ot[i].angl << " | "

<< setw(1) << ot[i].matematika << " | "

<< setw(1) << ot[i].history << " | "

<< setw(1) << srednee << " | "

<< endl;

drawLineN();

}

}

char choice;

do {

cout << "\n Возвращение в меню кнопка -y; выход из Меню любая другая кнопка... " << endl;

cin >> choice;

if (choice == 'y')

Menu(count, ot);

else return exit(1);

} while (1);

}

void search\_student(int count, OT\* ot) {

system("cls");

for (int i = 0; i < count; ++i) {

float srednee = (ot[i].matematika + ot[i].angl + ot[i].history) / 3;

cout << setw(60) << "Введите код студента" << endl;

cin >> ot[i].shifr\_studenta;

cout << setw(60) << "Студент" << endl;

drawLineW();

cout << setiosflags(ios::left) << setw(6) << "\n| Шифр студента: " << setw(6) << "| Шифр группы: |"

<< setw(20) << "Фамилия |" << setw(20) << " Имя |" << setw(20) << "Отчество |" << setw(1) << " Оценка по английскому: |" << setw(1) << " Оценка по математике: |"

<< setw(1) << " Оценка по истории : |"

<< setw(1) << "Среднее: |"

<< endl;

drawLineN();

if (ot[i].shifr\_studenta) {

cout << "\n" << "| "

<< setw(6) << ot[i].shifr\_studenta << " | "

<< setw(6) << ot[i].shifr\_gruppi << " | "

<< setw(20) << ot[i].familia << " | "

<< setw(20) << ot[i].imja << " | "

<< setw(20) << ot[i].otchestvo << " | "

<< setw(1) << ot[i].angl << " | "

<< setw(1) << ot[i].matematika << " | "

<< setw(1) << ot[i].history << " | "

<< setw(1) << srednee << " | "

<< endl;

drawLineN();

}

}

char choice;

do {

cout << "\n Возвращение в меню кнопка -y; выход из Меню любая другая кнопка... " << endl;

cin >> choice;

if (choice == 'y')

Menu(count, ot);

else return exit(1);

} while (1);

}