**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра информационных систем**

отчет

**по практической работе №1**

**по дисциплине «Программирование»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 2373 |  | Карпачёв А. С. |
| Преподаватель |  | Глущенко А. Г. |

Санкт-Петербург

2023

**Цель работы.**

Изучение структур в c++, массивов структур, создание прототипа баз данный в c++ при помощи структур, добавление функционала.

**Основные теоретические положения.**

В языке c++, структура — композитный тип данных, инкапсулирующий без сокрытия набор значений различных типов. Порядок размещения значений в памяти задаётся при определении типа и сохраняется на протяжении времени жизни объектов, что даёт возможность косвенного доступа. Пример объявления структуры: struct str\_name

Структуры можно использовать в качестве параметров функций, как и обычные переменные. Для структур поддерживаются все три механизма передачи данных: по значению, через указатели и по ссылке.

Объединения также хранят набор элементов, но в отличие от структуры все элементы объединения имеют нулевое смещение. А это значит, что разные элементы занимают в памяти один и тот же участок.

Для определения объединений применяется ключевое слово union и следующий формальный синтаксис:

union имя\_объединения

{

    определения\_элементов

};

**Постановка задачи**

Необходимо создать программу, позволяющую:

1.   Создание новой записи о студенте.

2.   Внесение изменений в уже имеющуюся запись.

3.   Вывод всех данных о студентах.

4.   Вывод информации обо всех студентах группы *N*. *N* – инициализируется пользователем.

5.   Вывод топа самых успешных студентов с наивысшим по рейтингу средним баллом за прошедшую сессию.

6.   Вывод количества студентов мужского и женского пола.

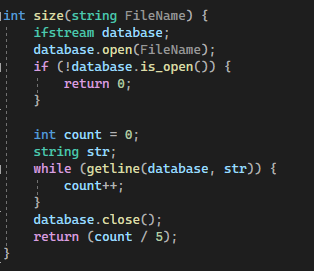
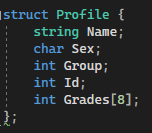
7.   Вывод данных о студентах, которые не получают стипендию; учатся только на «хорошо» и «отлично»; учатся только на «отлично»;

8.   Вывод данных о студентах, имеющих номер в списке – *k*.

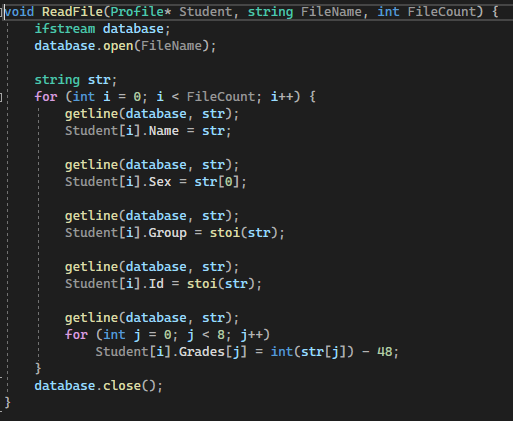
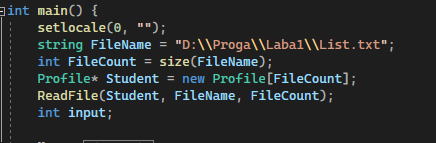
**Выполнение работы.**

Код программы представлен в приложении А.

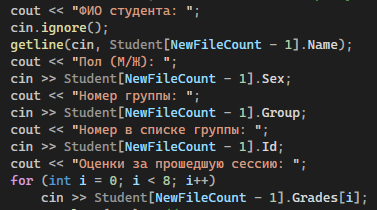
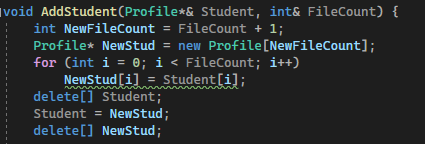
**Блок описания кода и использованных алгоритмов:**

Создаём динамический массив структур, размер которого узнаём считывая количество строк в изначальном файле List.txt.

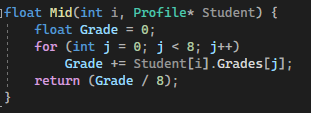
Далее просто считываем файл и полученные данные добавляем в массив структур.

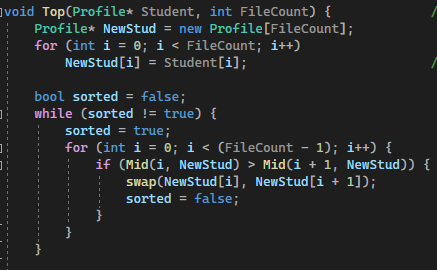


1. Сначала массив расширяется на единицу, а затем в появившуюся ячейку записываются данные о новом студенте.



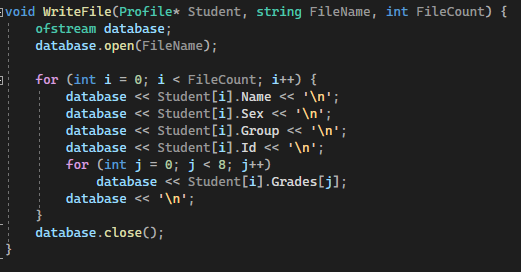
1. Пользователь вводит номер студента, данные которого хочет изменить, а затем меняет их.
2. Циклом for по очереди выводим всех студентов в списке.
3. Пользователь вводит номер группы и перебором выводятся студенты с соответствующим номером группы.
4. Сначала создаём функцию для высчитывания среднего значения оценок обучающегося.

Далее создаём копию исходного массива и сортируем его алгоритмом Bubble sort.



1. Перебором проходимся по всему массиву и увеличиваем счётчики обеих полов.
2. Делаем проверку оценок студентов. Если находим соответствие, то выводим в консоль.
3. Пользователь вводит номер и перебором выводятся студенты с соответствующим номеров в группе.

По завершению работы программы записываем получившиеся данные обратно в файл.



**Выводы.**

Мы научились использовать структуры c++, написали прототип баз данный c++, научились использовать функционал структур.

Приложение А

рабочий код

#include <iostream>

#include <string>

#include <fstream>

#include <Windows.h>

using namespace std;

string way = "C:\\Users\\artem\\source\\repos\\лаба сем 2\\лаба сем 2\\laba222.txt";

struct Book {

int id;

string name;

string autor;

};

struct Students

{

string name; // Фамилия Имя Отчество

string sex; // Пол: МУЖ или ЖЕН

int group; // Номер группы

int id; // Номер студента в списке

int grades[8]; // Оценки за экзамены и диф. зачёты

};

//Не относится к лабе-----------------------------------------------Тесты (начало)

void Refactfile(){

setlocale(LC\_ALL, "russian");

;

std::ofstream out("laba222.txt", std::ios::app);

if (out.is\_open())

{

cout << "Что вы хотите записать (текст) ?"<<"\n";

string save1;

cin >> save1;

std::cout << "В вайл записывается : " << save1 << std::endl;

out << save1 << std::endl;

}

out.close();

}

void Reedfile() {

setlocale(LC\_ALL, "russian");

std::string line;

std::ifstream in("laba222.txt"); // окрываем файл для чтения

if (in.is\_open())

{

while (getline(in, line))

{

std::cout << line << std::endl;

}

}

in.close(); // закрываем файл

}

void Cleanfile() {

cout << "Вы точно хотите удалить все записи в файле ?" << "\n";

cout << " Да(1) или Нет(0)" << "\n";

bool clenevar;

cin >> clenevar;

switch (clenevar)

{

case 1:

{

cout << "Файл очищен" << "\n";

ofstream file("laba222.txt");

break;

}

case 0:

{

cout << "Файл не будет очищен" << "\n";

break;

}

}

}

//Не относится к лабе-----------------------------------------------Тесты (конец)

void NewStudent() {

Students profile;

SetConsoleCP(1251); //Для коректной записи кириллицы в файл надо сменить кодировку консоли

cin.ignore();

cout << "ФИО: ";

getline(cin, profile.name);

cout << "Пол (МУЖ / ЖЕН): ";

cin >> profile.sex;

cout << "Номер группы: ";

cin >> profile.group;

cout << "Номер в списке группы: ";

cin >> profile.id;

cout << "Оценки за семестр:\n";

for (int i = 0; i < 8; i++) {

cin >> profile.grades[i];

Mcr: if (profile.grades[i] > 5 || profile.grades[i] < 1) {

cout << "Введена неверная оценка. Введите ещё раз: ";

cin >> profile.grades[i];

goto Mcr;

}

}

int temp = 0;

for (int i = 0; i < 8; i++) {

if (profile.grades[i] == 2)

temp++;

}

if (temp == 0) {

ofstream database;

database.open(way, ofstream::app);

if (!database.is\_open())

cout << "Ошибка сохранения данных!\n";

else {

database << profile.name << "\n";

database << profile.sex << "\n";

database << profile.group << "\n";

database << profile.id << "\n";

for (int i = 0; i < 8; i++) {

if (i == 7)

database << profile.grades[i];

else

database << profile.grades[i] << " ";

}

database << "\n";

cout << "Запись о студенте успешно сохранена!\n";

}

database.close();

}

else

cout << "Студент должен быть исключён. Запись не будет сохранена\n";

SetConsoleCP(866); //Возвращение старой кодировки

}

int CountStudents() {

ifstream database;

database.open(way);

if (database.is\_open())

{

int temp = 0;

string data;

while (!database.eof())

{

getline(database, data);

temp++;

}

database.close();

int n;

n = temp / 5;

return n;

}

else return 0;

}

void CreateMassStudents(Students\* student, int size) {

char trash;

ifstream database;

database.open(way);

if (!database.is\_open())

cout << "Ошибка открытия файла!\n";

else {

if (size == 0)

cout << "Файл пуст!";

else {

for (Students\* p = student; p < student + size; p++) {

getline(database, p->name);

getline(database, p->sex);

database >> p->group >> p->id;

for (int i = 0; i < 8; i++)

database >> p->grades[i];

database.get(trash);

}

}

}

}

void ChangeStudent() {

int size;

size = CountStudents();

Students\* student = new Students[size];

CreateMassStudents(student, size);

if (size == 0) {

delete[] student;

system("pause");

return;

}

string chName;

int change;

cin.ignore();

SetConsoleCP(1251);

while (true) {

cout << "Введите полное имя студента, или введите 'выход' : \n";

getline(cin, chName);

if (chName == "выход")

break;

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

if (student[i].name == chName) {

cout << "Что вы ходите изменить?\n" << "1 - ФИО.\n" << "2 - Пол.\n" << "3 - Номер группы.\n" << "5 - Оценки за семестр.\n" << "6 - Отменить изменение.\n" << "0 - Выход" << "Ваш выбор: ";

cin >> change;

switch (change) {

case 1:

cout << "Введите новые данные: ";

cin.ignore();

getline(cin, student[i].name);

break;

case 2:

cout << "Введите новые данные: ";

cin >> student[i].sex;

cin.ignore();

break;

case 3:

cout << "Введите новые данные: ";

cin >> student[i].group;

cin.ignore();

break;

case 4:

cout << "Введите новые данные: ";

cin >> student[i].id;

cin.ignore();

break;

case 5:

cout << "Введите новые данные:\n";

for (int j = 0; j < 8; j++) {

cin >> student[i].grades[j];

Mch: if (student[i].grades[j] > 5 || student[i].grades[j] < 1) {

cout << "Введена неверная оценка. Введите ещё раз: ";

cin >> student[i].grades[j];

goto Mch;

}

}

cin.ignore();

break;

case 6:

cin.ignore();

break;

case 0:

chName = "выход";

}

break;

}

if (i == (size - 1))

cout << "Данный студент не найден!\n";

}

}

ofstream database;

database.open(way);

if (!database.is\_open()) {

cout << "Ошибка открытия файла!\n";

}

else {

for (Students\* p = student; p < student + size; p++) {

database << p->name << "\n" << p->sex << "\n" << p->group << "\n" << p->id << "\n";

for (int i = 0; i < 8; i++) {

if (i == 7)

database << p->grades[i];

else

database << p->grades[i] << " ";

}

database << "\n";

}

}

database.close();

cout << "Сохранение успешно завершено!\n";

SetConsoleCP(866);

delete[] student;

}

void showGroup()

{

int n;

cout << "Введите номер группы: ";

cin >> n;

int size;

size = CountStudents();

Students\* student = new Students[size];

CreateMassStudents(student, size);

if (size == 0) {

delete[] student;

return;

}

int temp = 0;

for (Students\* p = student; p < student + size; p++)

if (p->group == n) {

temp++;

cout << "ФИО: " << p->name << "\nПол: " << p->sex << "\nгруппа: " << p->group << "\nНомер в группе: " << p->id << "\nОценки за семестр: ";

for (int i = 0; i < 8; i++)

cout << p->grades[i] << " ";

cout << "\n\n";

}

delete[] student;

if (temp == 0)

cout << "Студентов данной группы не найдено!\n";

}

void showsex()

{

string n;

cout << "Введите пол: ";

setlocale(LC\_ALL, "Rus"); // Подключение русского языка

SetConsoleCP(1251); //Для коректной записи кириллицы в файл надо сменить кодировку консоли

cin >> n;

SetConsoleCP(866);

//cout << n;

int size;

size = CountStudents();

Students\* student = new Students[size];

CreateMassStudents(student, size);

if (size == 0) {

delete[] student;

return;

}

int temp = 0;

int men = 0, women = 0;

if (size == 0) {

delete[] student;

return;

}

for (Students\* p = student; p < student + size; p++) {

if (p->sex == "МУЖ")

men++;

if (p->sex == "ЖЕН")

women++;

}

cout << "Найдено:\n"<< "Парней : "<<men<<"\n"<<"Девушек : " << women << "\n";

cout <<"\n"<<"Список найденых стдудентов :" << "\n";

for (Students\* p = student; p < student + size; p++)

if (p->sex == n) {

temp++;

cout << "ФИО: " << p->name << "\nПол: " << p->sex << "\nгруппа: " << p->group << "\nНомер в группе: " << p->id << "\nОценки за семестр: ";

for (int i = 0; i < 8; i++)

cout << p->grades[i] << " ";

cout << "\n\n";

}

delete[] student;

if (temp == 0)

cout << "Студентов данной группы не найдено!\n";

}

void showid()

{

int n;

cout << "Введите ID: ";

cin >> n;

int size;

size = CountStudents();

Students\* student = new Students[size];

CreateMassStudents(student, size);

if (size == 0) {

delete[] student;

return;

}

int temp = 0;

for (Students\* p = student; p < student + size; p++)

if (p->id == n) {

temp++;

cout << "ФИО: " << p->name << "\nПол: " << p->sex << "\nгруппа: " << p->group << "\nНомер в группе: " << p->id << "\nОценки за семестр: ";

for (int i = 0; i < 8; i++)

cout << p->grades[i] << " ";

cout << "\n\n";

}

delete[] student;

if (temp == 0)

cout << "Студентов данной группы не найдено!\n";

}

int main()

{

setlocale(LC\_ALL, "Rus"); // Подключение русского языка

{

while (true) {

system("cls");

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << "\n";

cout << "| Меню : |" << "\n";

cout << "| Тестовые пункты: |" << "\n";

cout << "| 1) 1 Сделать запись в файле(просто добовление текста в блокнот). |" << "\n";

cout << "| 2) 2 Прочитать записи в файле(посмотреть всех студентов) . |" << "\n";

cout << "| 3) 3 Удалить все записи в файле. |" << "\n";

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

cout << "| Начало заданий по лабораторной работы |" << "\n";

cout << "| 4) 4 Запись о новом студенте |" << "\n";

cout << "| 5) 5 Изменение записи о студенте |" << "\n";

cout << "| 6) 6 Вывод информации обо всех студентах определенной группы |" << "\n";

cout << "| 7) 7 Вывод определленного пола студентов |" << "\n";

cout << "| 8) 8 Вывод определленного ID |" << "\n";

cout << "| 0) 0 Выход из программы |" << "\n";

cout << "|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|" << "\n";

cout << "Выберите пункт : " << "\n";

cout << endl;

int choice;

cin >> choice ;

switch (choice) {

case 0:

system("cls");

cout << " Спасибо за просмотр" << "\n";

cout << " попугай вторник" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$\_ \_\_$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$\_\_\_\_\_( &)\_$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$\_\_\_\_\_\_\_\_ \_\_\_\_$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$$$\_\_\_\_§ §§§\_\_\_\_\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$$$$$$§§ ?§\_§§§\_\_\_\_\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$$$$$$$$$§§ §§§\_§\_\_\_\_" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$$$$$$\_$$$$\_\_\_§ §§\_\_§\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$\_$$$$$$$$\_$\_\_\_\_\_\_ \_\_§" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$$$$$$$$$$$$$$$$\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$$\_$$$$$$$$$$$$$$$$\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$$\_§$$$$$$$$$§$$$$$$$$\_$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$$\_$$§§§$$$§$$§§$$$$§§$\_ $$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_\_$$$$$$\_§§§$$§§$$§§§$$§§$$§ ?$\_$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_\_$$$$$\_$§§§$$$$§§§$$$§§§§§$$\_ $$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_\_$$$$$\_§§§§§§$$§§§$$§§§§§§ §\_$$$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_\_$$$$\_§§0§§§§§§§§§§§§§§§ ?\_$$$$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_$$$\_000§§§00§§§§§000§00§\_$$$$$$" << "\n";

cout << "\_\_\_\_\_\_\_\_$$\_0000§§0000§§§00§0§\_$$$$$$" << "\n";

cout << "\_\_\_\_\_\_\_$\_0000§§00000§0000\_$$$$$$$" << "\n";

cout << "\_\_\_\_\_§§00000§000§0000\_$$$$$$$$" << "\n";

cout << "\_\_\_§00000§0000000\_$$$$$$$$$$" << "\n";

cout << "\_\_\_000§00000§\_§§§§$$$$$$$$$$$$$\_\_\_......" << "\n";

cout << "\_\_\_000§00000§\_§§§§$$$$$$$$$$$$$\_\_\_......" << "\n";

cout << "\_000§0\_§§§§§§$$$$$$$\_§000\_$$$§§§\_000000 0000000000000" << "\n";

cout << "00\_\_\_\_§§§§§$$$$$$$\_§§00000\_$§§§\_00000000 000000000000" << "\n";

cout << "\_\_\_\_\_\_§§§§$$$$$$$\_§§§\_’’\_\_\_\_\_\_§§§\_\_???’" << "\n";

cout << "\_\_\_\_\_§§§$$$$$$" << "\n";

cout << "\_\_\_\_§§$$$$$$$" << "\n";

cout << "\_\_\_$$$$$$$$" << "\n";

cout << "\_\_$$$$$$$$" << "\n";

cout << "\_$$$$$$$$" << "\n";

cout << "$$$$$$$$" << "\n";

cout << "$$$$$$$" << "\n";

cout << "$$$$$$" << "\n";

cout << "$$$$" << "\n";

return 0;

case 1:

system("cls");

Refactfile();

Reedfile();

system("pause");

break;

case 2:

system("cls");

Reedfile();

system("pause");

break;

case 3:

system("cls");

Cleanfile();

Reedfile();

system("pause");

break;

case 4:

system("cls");

NewStudent();

system("pause");

break;

case 5:

system("cls");

ChangeStudent();

system("pause");

break;

case 6:

system("cls");

showGroup();

system("pause");

break;

case 7:

system("cls");

showsex();

system("pause");

break;

case 8:

system("cls");

showid();

system("pause");

break;

case 9:

system("cls");

cout << "|" << "Students =" << " " << sizeof(Students) << "byte" << "|" << endl;

system("pause");

break;

default:

cout << "\n не найден данный символ " << choice << " d" << endl;

system("pause");

break;

}

}

}

return 0;

}