Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №11

По дисциплине: «Основы алгоритмизации и программирования»

**Тема:** **«Бинарные и текстовые файлы»**

Вариант №11

Выполнил:

Студент 1-го курса

Группы ПО-7

Лобан К.Ю.

Проверила:

Войцехович О. Ю.

Брест 2021

**Цель работы:** Изучить принципы программирования с использованием бинарных файлов. Ознакомиться с основными функциями в Си для работы с бинарными файлами.

**Текст программы:**

#include <iostream>

#include <string>

#include <Windows.h>

#include <fstream>

using namespace std;

struct Enrollee

{

string name;

string city;

enum

{

Math,

English,

Physic

} subjects;

float marks[3];

};

int main()

{

setlocale(LC\_ALL, "Russian");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int choice, n, kol = 0;

cout << "Введите количество абитуриентов: ";

cin >> n;

Enrollee\* en = new Enrollee[n];

string name;

char city[50];

float math, eng, phys, sum = 0;

bool found = false;

do

{

cout << endl << "Меню: " << endl;

cout << "1-Ввод данных;" << endl;

cout << "2-Вывод данных;" << endl;

cout << "3-Сортировка по ФИО;" << endl;

cout << "4-Удаление по заданному значению поля;" << endl;

cout << "5-Средний балл абитуриентов из Минска;" << endl;

cout << "6-Выход." << endl;

cout << endl << "Ваш выбор: ";

cin >> choice;

switch (choice)

{

case 1:

{

char c;

cout << "Какой файл нужно создать? (b-бинарный, t-текстовый): ";

do

{

cin >> c;

} while (c != 'b' && c != 't');

if (c == 'b')

{

ofstream out;

out.open("Enrollee.txt", ios::binary);

if (!out.is\_open())

{

cout << "Не удалось открыть файл" << endl;

}

else

{

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

{

cin.ignore(cin.rdbuf()->in\_avail());

cout << "Введите ФИО: ";

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

cout << "Введите город: ";

cin.ignore(cin.rdbuf()->in\_avail());

getline(cin, en[i].city);

cout << "Введите оценки по предметам: " << endl;

cout << "Математика: ";

cin >> en[i].marks[en->Math];

cout << "Английский: ";

cin >> en[i].marks[en->English];

cout << "Физика: ";

cin >> en[i].marks[en->Physic];

cin.ignore(cin.rdbuf()->in\_avail());

}

out.write((char\*)&n, sizeof(int));

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

{

out.write((char\*)&(en[i]), sizeof(Enrollee));

}

}

out.close();

}

else

{

ofstream out;

out.open("Enrollee.txt", ios::binary);

if (!out.is\_open())

{

cout << "Не удалось открыть файл" << endl;

}

else

{

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

{

cin.ignore(cin.rdbuf()->in\_avail());

cout << "Введите ФИО: ";

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

cout << "Введите город: ";

cin.ignore(cin.rdbuf()->in\_avail());

getline(cin, en[i].city);

cout << "Введите оценки по предметам: " << endl;

cout << "Математика: ";

cin >> en[i].marks[en->Math];

cout << "Английский: ";

cin >> en[i].marks[en->English];

cout << "Физика: ";

cin >> en[i].marks[en->Physic];

cin.ignore(cin.rdbuf()->in\_avail());

}

out.write((char\*)&n, sizeof(int));

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

{

out.write((char\*)&(en[i]), sizeof(Enrollee));

}

}

out.close();

}

}

break;

case 2:

{

/\*for (int i = 0; i < n; i++)

{

cout << i + 1 << ". ФИО: " << en[i].name << endl;

cout << "Город: " << en[i].city << endl;

cout << "Оценки: " << endl;

cout << "Математика - " << en[i].marks[en->Math];

cout << ", Английский - " << en[i].marks[en->English];

cout << ", Физика - " << en[i].marks[en->Physic] << endl << endl;

}\*/

int n2;

Enrollee \*en2 = new Enrollee[n];

ifstream in("Enrollee.txt", ios::in | ios::binary);

if (!in.is\_open())

{

cout << "Файл отсутствует" << endl;

}

else

{

in.read((char\*)&n2, sizeof(int));

for (int i = 0; i < n2; i++)

{

in.read((char\*)&(en2[i]), sizeof(Enrollee));

}

in.close();

cout << "Содержимое файла: " << endl;

for (int i = 0; i < n2; i++)

{

cout << i + 1 << ". ФИО: " << en2[i].name << endl;

cout << "Город: " << en2[i].city << endl;

cout << "Оценки: " << endl;

cout << "Математика - " << en2[i].marks[en2->Math];

cout << ", Английский - " << en2[i].marks[en2->English];

cout << ", Физика - " << en2[i].marks[en2->Physic] << endl << endl;

}

}

}

break;

case 3:

{

for (int i = n - 1; i >= 0; i--)

{

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

{

if (en[j].name > en[j + 1].name)

{

Enrollee temp = en[j];

\*(en + j) = \*(en + j + 1);

\*(en + j + 1) = temp;

}

}

}

}

break;

case 4:

{

char c;

cout << "По какому полю произвести удаление? (n-ФИО, c-город, m-оценка по математике, e-оценка по английскому, p-оценка по физике): ";

cin >> c;

switch (c)

{

case 'n':

{

cin.ignore(cin.rdbuf()->in\_avail());

cout << "Введите ФИО: ";

getline(cin, name);

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

{

if (name == en[i].name)

{

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

{

en[j] = en[j + 1];

}

n--;

}

}

}

break;

case 'c':

{

cin.ignore(cin.rdbuf()->in\_avail());

cout << "Введите город: ";

cin.getline(city, 50);

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

{

if (city == en[i].city)

{

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

{

en[j] = en[j + 1];

}

n--;

}

}

}

break;

case 'm':

{

cout << "Введите оценку по математике: ";

cin >> math;

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

{

if (math == en[i].marks[en->Math])

{

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

{

en[j] = en[j + 1];

}

n--;

}

}

}

break;

case 'e':

{

cout << "Введите оценку по английскому: ";

cin >> eng;

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

{

if (eng == en[i].marks[en->English])

{

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

{

en[j] = en[j + 1];

}

n--;

}

}

}

break;

case 'p':

{

cout << "Введите оценку по физике: ";

cin >> phys;

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

{

if (phys == en[i].marks[en->Physic])

{

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

{

en[j] = en[j + 1];

}

n--;

}

}

}

break;

}

}

break;

case 5:

{

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

{

if (en[i].city == "Минск")

{

found = true;

kol++;

sum += en[i].marks[en->Math] + en[i].marks[en->English] + en[i].marks[en->Physic];

}

}

if (found)

{

cout << "Средний балл минских абитуриентов: " << sum / (3 \* kol) << endl;

}

else

{

cout << "Ничего не найдено!" << endl;

}

}

break;

case 6:

{

cout << "Выход из программы..." << endl;

}

break;

default:

{

cout << "Неверный ввод!" << endl;

}

}

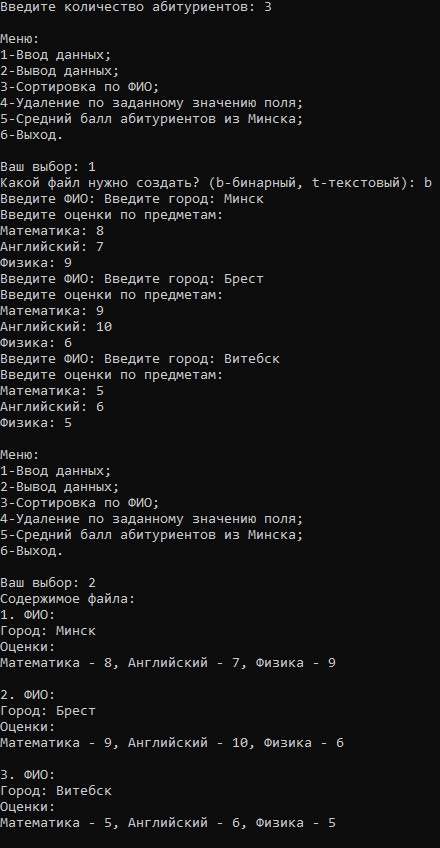
} while (choice != 6);

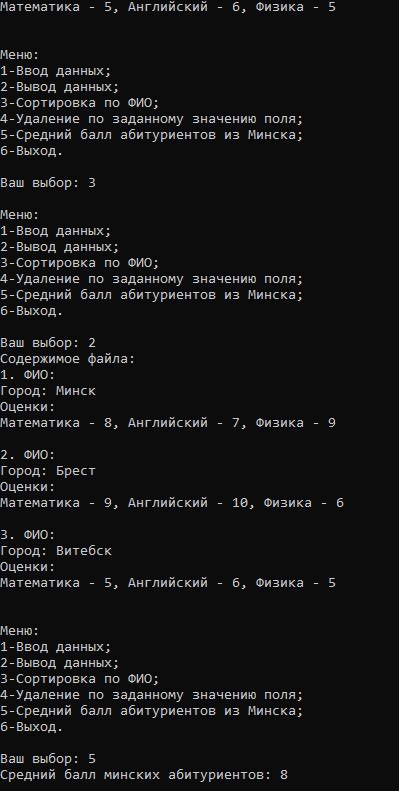
delete[]en;

return 0;

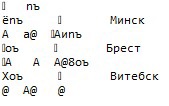
}

**Результаты:**

****

****

В бинарный файл записалось как:



Двоичные файлы сохраняют информацию аналогично представлении информации в памяти компьютера во время работы программы, что позволяет не выполняется никаких преобразований, ускоряя тем самым процесс чтения. Кроме преимущества, двоичные файлы имеют огромный недостаток, связанный с их переносимостью. Поэтому двоичный файл, может по-разному прочитаться на разных операционных системах.

Текстовые файлы являются универсальным средством представления информации в связи с их переносимостью. То есть, не зависимо от разных операционных систем, информация будет прочитана правильно, за исключением проблем с кодированием. Также преимуществом текстовых файлов является независимость от порядка байт в слове.

**Вывод:** я изучилпринципы программирования с использованием бинарных файлов. Ознакомился с основными функциями в Си для работы с бинарными файлами.