|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА** – **Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт информационных технологий |
| Кафедра вычислительной техники |
|  |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2** | |
| **по дисциплине** | |
| *«Структуры и алгоритмы обработки данных»* | |
| Выполнил студент группы *ИКБО-02-17* | *Данилов Д.Е.* |
| Принял | *Асадова Ю.С.* |

|  |  |  |
| --- | --- | --- |
| Лабораторные работы выполнены | «\_\_»\_\_\_\_\_\_\_201\_\_ г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *Подпись студента* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_201\_\_ г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *Подпись руководителя* |

Москва 2018

|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА** – **Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт информационных технологий

Кафедра вычислительной техники

Выполнено \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Д.Е. Данилов/

Зачтено \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Ю.С. Асадова/

**Задание на лабораторную работу**

**Дисциплина**: **«**Структуры и алгоритмы обработки данных**»**

**Студент** Данилов Д.Е. **Шифр** 17и0281 **Группа** ИКБО-02-17

**1. Тема**: «Файлы»

**2. Срок сдачи студентом законченной работы:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г.

**3. Исходные данные:** индивидуальная тема, среда разработки, средства языка С++

**4. Задание:** Разработать программу, которая читает данные из одного файла и записывает их в бинарный.

**5. Содержание отчета:**

* титульный лист;
* задание;
* оглавление;
* введение;
* основные разделы отчета;
* заключение;

Руководитель работы Ю.С. Асадова \_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2018г.

подпись

Задание принял к исполнению И.О. Фамилия \_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2018г.

подпись

*Оглавление*

[Введение 4](#_Toc524637062)

[Основные разделы отчёта 5](#_Toc524637063)

Заключение …………………………………………………………………………. 9

# *Введение*

Целью данной лабораторной работы является закрепление знаний, полученных при изучении языка С++, а именно: работа по теме файлы. Лабораторная работа была выполнена в среде разработки Microsoft Visual Studio 2017 на языке C++.

1. ***Теоретическое введение***

**Файлом** называют способ хранения информации на физическом устройстве. Файл — это понятие, которое применимо ко всему — от файла на диске до терминала.

В C++ отсутствуют операторы для работы с файлами. Все необходимые действия выполняются с помощью функций, включенных в стандартную библиотеку. Они позволяют работать с различными устройствами, такими, как диски, принтер, коммуникационные каналы и т.д. Эти устройства сильно отличаются друг от друга. Однако файловая система преобразует их в *единое абстрактное логическое устройство*, называемое **потоком**.

**Текстовый поток** — это последовательность символов. При передаче символов из потока на экран, часть из них не выводится (например, символ возврата каретки, перевода строки).

**Двоичный поток** — это последовательность байтов, которые однозначно соответствуют тому, что находится на внешнем устройстве.

1. ***Постановка задачи***

Разработать программу, выполняющую следующие функции:

\* ввод данных об n объектах из текстового файла в массив структур (0<n<=50);

\* сортировку массива структур по возрастанию значений одного из полей структуры;

\* вывод данных об объектах на экран в упорядоченном по возрастанию виде;

\* поиск объекта по значению одного из полей;

\* запись упорядоченных данных об объектах в двоичный файл;

\* чтение двоичного файла.

**Дано:** файл со структурами.

**Ограничения на данные:** количество элементов не должно превышать 50

**Ограничения на решаемую задачу**: нет

1. ***Определение входных и выходных данных.***

**Входные данные** – текстовый файл.

**Результат** – новый бинарный файл и вывод на экран

1. ***Краткое описание программы.***

Из файла заполняется массив структур. Сортируется по одному из полей. Выводится на экран. Записывается в бинарный файл. Выводится на экран содержимое бинарного файла.

**V. *Разработка программы (кодирование алгоритма)***

#include "stdafx.h"

#include <iostream>

#include <fstream>

#include <vector>

#include <string>

using namespace std;

struct Date

{

string day;

string month;

string year;

};

struct PhoneBook

{

string phoneNumber;

string firstName;

string secondName;

string patronymic;

Date birthDate;

};

void fillFromFile(vector <PhoneBook> &arr)

{

cout << "Structs from file:" << endl << endl;

fstream f;

string tempString;

PhoneBook tempStruct;

f.open("Structs.txt");

while(!f.eof()) {

getline(f, tempStruct.phoneNumber);

getline(f, tempStruct.firstName);

getline(f, tempStruct.secondName);

getline(f, tempStruct.patronymic);

getline(f, tempStruct.birthDate.day);

getline(f, tempStruct.birthDate.month);

getline(f, tempStruct.birthDate.year);

arr.push\_back(tempStruct);

}

}

void sort(vector <PhoneBook> &arr)

{

for (int i = 0; i < arr.size() - 1; i++) {

for (int k = 0; k < arr.size() - i - 1; k++) {

if (arr[i].secondName == arr[i + 1].secondName) {

if (arr[i].firstName == arr[i + 1].firstName) {

if (arr[i].patronymic == arr[i + i].patronymic) {

//lol

}

else {

for (int p = 0; p < arr[i].patronymic.size() - (arr[i].patronymic.size() - arr[i + 1].patronymic.size()); p++) {

if (arr[i].patronymic[p] < arr[i + 1].patronymic[p]) {

swap(arr[i], arr[i + 1]);

}

}

}

}

else {

for (int s = 0; s < arr[i].firstName.size() - (arr[i].firstName.size() - arr[i + 1].firstName.size()); s++) {

if (arr[i].firstName[s] < arr[i + 1].firstName[s]) {

swap(arr[i], arr[i + 1]);

}

}

}

}

else {

for (int j = 0; j < arr[i].secondName.size() - (arr[i].secondName.size() - arr[i + 1].secondName.size()); j++) {

if (arr[i].secondName[j] < arr[i + 1].secondName[j]) {

swap(arr[i], arr[i + 1]);

}

}

}

}

}

}

void display(vector <PhoneBook> arr)

{

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

cout << arr[i].phoneNumber << '\t';

cout << arr[i].secondName << '\t';

cout << arr[i].firstName << '\t';

cout << arr[i].patronymic << '\t';

cout << arr[i].birthDate.day << "." << arr[i].birthDate.month << "." << arr[i].birthDate.year << endl << endl;

}

}

void displayOne(vector <PhoneBook> arr, int index)

{

cout << endl << "Struct was found" << endl << endl;

cout << arr[index].phoneNumber << '\t';

cout << arr[index].secondName << '\t';

cout << arr[index].firstName << '\t';

cout << arr[index].patronymic << '\t';

cout << arr[index].birthDate.day << "." << arr[index].birthDate.month << "." << arr[index].birthDate.year << endl << endl;

}

void NOTbinarySearch(vector <PhoneBook> arr)

{

string key;

cout << "Enter key: ";

cin >> key;

int count = 0;

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

if (arr[i].firstName == key || arr[i].secondName == key || arr[i].patronymic == key) {

displayOne(arr, i);

count++;

}

}

if (count == 0) {

cout << "Struct was not found" << endl;

}

}

void createBin(vector <PhoneBook> arr)

{

ofstream out("BinaryStructs.bin", std::ios::binary);

if (out.is\_open())

{

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

for (int q = 0; q < arr[i].phoneNumber.size(); q++) {

out.write((char \*)&arr[i].phoneNumber[q], sizeof(char));

}

for (int w = 0; w < arr[i].secondName.size(); w++) {

out.write((char \*)&arr[i].secondName[w], sizeof(char));

}

for (int e = 0; e < arr[i].firstName.size(); e++) {

out.write((char \*)&arr[i].firstName[e], sizeof(char));

}

for (int r = 0; r < arr[i].patronymic.size(); r++) {

out.write((char \*)&arr[i].patronymic[r], sizeof(char));

}

for (int t = 0; t < arr[i].birthDate.day.size(); t++) {

out.write((char \*)&arr[i].birthDate.day[t], sizeof(char));

}

for (int y = 0; y < arr[i].birthDate.month.size(); y++) {

out.write((char \*)&arr[i].birthDate.month[y], sizeof(char));

}

for (int u = 0; u < arr[i].birthDate.year.size(); u++) {

out.write((char \*)&arr[i].birthDate.year[u], sizeof(char));

}

}

out.close();

}

}

void readBin()

{

cout << "Bin File Open" << endl;

ifstream instrm("BinaryStructs.bin", std::ios::binary);

char a;

while (instrm.read((char \*)&a, sizeof(char)))

cout << a << ' ';

cout << endl;

}

int main()

{

setlocale(LC\_ALL, "rus");

cout << "Лабораторная работа 11, Вариант 3, Данилов Дмитрий, ИКБО-02-17" << endl;

cout << "Задание : 'Работа с файлами'" << endl << endl;

vector <PhoneBook> arr;

fillFromFile(arr);

sort(arr);

display(arr);

NOTbinarySearch(arr);

createBin(arr);

readBin();

system("pause");

return 0;

}

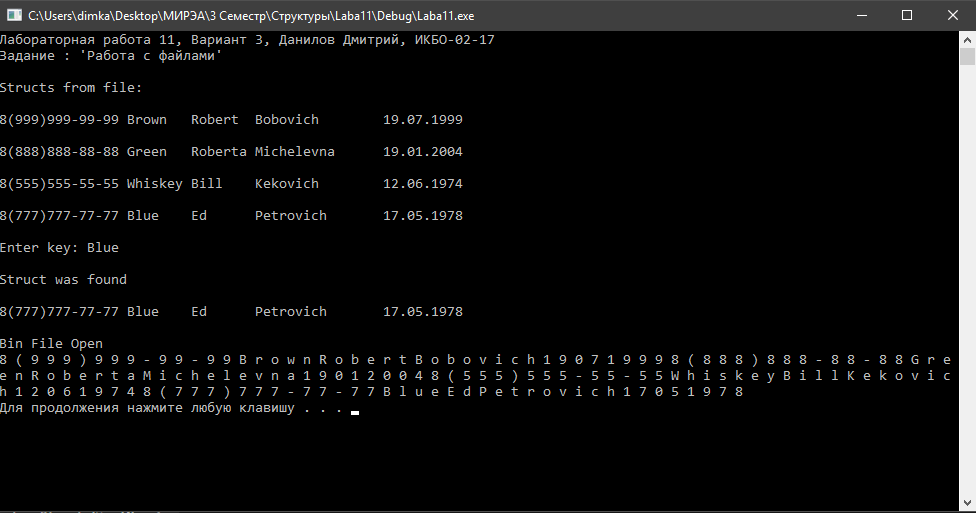


Рис.1 Результат выполнения теста 1

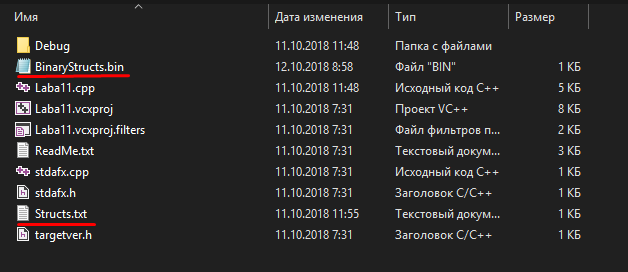


Рис.2 Полученные файлы

***Заключение***

В ходе выполнения работы были получены практические навыки при работе по теме файлы на языке С++.