ОДЕССКИЙ НАЦИОНАЛЬНЫЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ИНСТИТУТ КОМПЬЮТЕРНЫХ СИСТЕМ

Кафедра информационных систем и технологий

**Лабораторная работа №10**

по дисциплине: «Алгоритмизация и программирование»

Вариант № 1

**Выполнил:**

АД-192

Березовский В. А.

**Проверили:**

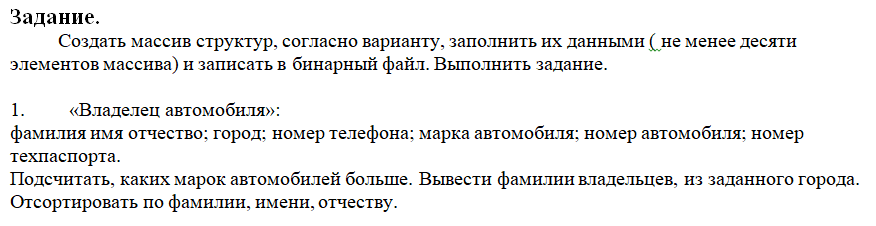
Одесса 2020

**Тема:** работа с бинарными файлами.

**Цель:** научиться правильно, познать материал, выполнять все задания, работать с бинарными файлами.

**Ход работы**

**Задание:**

****

**Листинг:**

#include <iostream>

#include <fstream> // подключаем данную библиотеку, для работы с потоками

#include <string> // подключаем данную библиотеку, для работы со строками

using namespace std;

struct Struct { // создаем структуру с массивами для заполнения данных

char Surname[20]; // фамилия

char Name[15]; // имя

char Patronimic[20]; // отчество

char City[15]; // город

char Phone[15]; // номер телефона

char Car\_model[15]; // марка машины

char Car\_number[10]; // номер машины

char Certificate\_number[15]; // номер техпаспорта

};

void surname(Struct Fs[5]); // создание прототипов функций

void models(char fs[10], char sf[10]); // создание прототипов функций

void found(int\* ps, int ms); // создание прототипов функций

int main() {

setlocale(LC\_ALL, "en"); // подключаем английский язык

Struct Human[5]; // создаем массив который будет содержать все массивы в структуре

fstream file("Car owner.txt", ios::binary | ios::in | ios::out); // открываем поток для работы с бинарным файлом

if (!file.is\_open()) { // проверка на открытие

cout << "Error!" << endl << endl; // если файл не открылся

}

else { // если файл открылся

int count = 5; // кол-во человек

Human[0] = { "Berezovskiy","Vlad","Alexsandrovich","Odessa","0966192686","BMW","3686","4789" }; // данные первого человека

Human[1] = { "Tsapenko","Ludmila","Efimova","Odessa","0674934511","Mercedez","8201","7301" }; // данные второго человека

Human[2] = { "Chechelnitskiy","Evgeny","Igorevich","Kiev","0973430765","BMW","8483","2791" }; // данные третьего человека

Human[3] = { "Arman","Maksim","Nikolaevich","Donetsk","0935712635","Mazda","0999","6281" }; // данные четвертого человека

Human[4] = { "Gasimov","Daniil","Vladimirovich","Nikolaev","0957216755","Lada","3781","0001" }; // данные пятого человека

file.write((char\*)&Human[0], sizeof(Human[0])); // записываем в файл данные 1 человека

file.write((char\*)&Human[1], sizeof(Human[1])); // записываем в файл данные 2 человека

file.write((char\*)&Human[2], sizeof(Human[2])); // записываем в файл данные 3 человека

file.write((char\*)&Human[3], sizeof(Human[3])); // записываем в файл данные 4 человека

file.write((char\*)&Human[4], sizeof(Human[4])); // записываем в файл данные 5 человека

file.close(); // закрываем файл после работы с записью

file.read((char\*)&Human[0], sizeof(Human[0])); // прочитываем в файле данные 1 человека

file.read((char\*)&Human[1], sizeof(Human[1])); // прочитываем в файле данные 2 человека

file.read((char\*)&Human[2], sizeof(Human[2])); // прочитываем в файле данные 3 человека

file.read((char\*)&Human[3], sizeof(Human[3])); // прочитываем в файле данные 4 человека

file.read((char\*)&Human[4], sizeof(Human[4])); // прочитываем в файле данные 5 человека

string key; // контейнер для ввода города

cout << "Enter city: ";

cin >> key;

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

if (Human[i].City == key) { // проверка на соотв. город

cout << Human[i].Surname << "\t" << Human[i].Name << "\t" << Human[i].Patronimic << endl;

}

}

char cont[5][10]; // создание массив символов

int sum[5] = { 0,0,0,0,0 }; // счетчик для кол-во марок машин

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

models(cont[i], Human[i].Car\_model); // копирование из структуры в массив

}

for (int i = 0; i < 5; i++) { // подсчет количества каждой марки машины

if (Human[i].Car\_model[0] == cont[0][0]) {

sum[0]++;

}

else if (Human[i].Car\_model[0] == cont[1][0]) {

sum[1]++;

}

else if (Human[i].Car\_model[0] == cont[2][0]) {

sum[2]++;

}

else if (Human[i].Car\_model[0] == cont[3][0]) {

sum[3]++;

}

else if (Human[i].Car\_model[0] == cont[4][0]) {

sum[4]++;

}

}

int max = 0;

found(sum, max); // поиск номера элемента которого больше всего

cout << "\nMore " << cont[max] << " models!" << endl << endl;

surname(Human);

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

cout << Human[i].Surname << "\t" << Human[i].Name << "\t" << Human[i].Patronimic << "\n";

}

}

}

void models(char fs[10], char sf[10]) { // ф-ция по присвоению в отдельный массив всех марок машин по отдельности

for (int i = 0; i < 10; i++) {

fs[i] = sf[i];

}

}

void surname(Struct Fs[5]) { // ф-ция по сортировке фамилий

for (int i = 0; i < 5; i++) {

for (int j = 0; j < 4; j++) {

if ((int)Fs[j].Surname[0] > (int)Fs[j + 1].Surname[0]) {

swap(Fs[j], Fs[j + 1]);

}

}

}

}

void found(int\* ps, int ms) { // ф-ция по поиску номера

for (int i = 0; i < 5; i++) {

for (int j = 0; j < 4; j++) {

if (ps[j] > ps[j + 1]) {

ms = j;

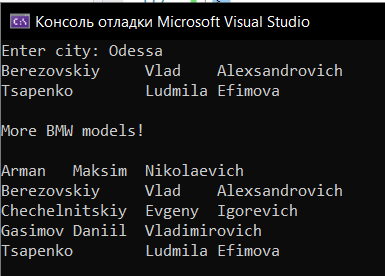
}

}

}

}

**Тестирование программы:**

****

**Вывод:** янаучился правильно, познавать материал, выполнять все задания, работать с бинарными файлами. Все задания были успешно выполнены и готовы к рассмотрению.