Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет»

ПНИПУ

**Лабораторная работа**

**«Структуры в C++»**

Выполнил:

студент группы РИС-23-2б

Кобзев Станислав Игоревич  
Проверила:

доцент кафедры ИТАС

О.А. Полякова

Пермь 2024 г.

**Структуры в C++**

**Постановка задачи:**

1. Сформировать двоичный файл из элементов, заданной в варианте структуры, распечатать его содержимое, выполнить удаление и добавление элементов.

Структура "Автомобиль":

* марка;
* год выпуска;
* цена;
* цвет.

Удалить все элементы, у которых год выпуска меньше заданного, добавить элемент в начало файла.

**Анализ задачи:**

1. Для работы с данными автомобилей создаётся структура car, поля которой описывают марку, год выпуска, цену и цвет. Структура содержит метод для инициализации полей элемента.
2. Внутри функции main создаётся массив элементов структуры car, обозначается количество машин. С помощью цикла их данные вводятся и с клавиатуры
3. Для хранения индексов элементов, которые нужно удалить, создаётся отдельный массив. Если год выпуска каких-либо машин меньше введенного, их индексы помещаются в этот массив.
4. В файл записываются только те автомобили, индексы в массиве у которых не совпадают с запрещёнными.

**Блок схема:**

**Код программы:**

#include <iostream>

#include <Windows.h>

#include <string>

#include <fstream>

using namespace std;

struct car

{

string mark;

int year;

int price;

string color;

void enterInfo(string m, int y, int p, string c)

{

mark = m;

year = y;

price = p;

color = c;

}

void print() {

cout << "mark\t" << mark << "\nyear\t" << year << "\nprice\t" << price << "\ncolor\t" << color << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

car array[10]; // Массив всех машин

cout << "Enter the number of car: ";

int car\_count; // Количество машин

cin >> car\_count;

for (int i = 0; i < car\_count; i++)

{

string m, c;

int y, p;

cout << "Enter mark, year, price, color: ";

cin >> m >> y >> p >> c;

array[i].enterInfo(m, y, p, c); // Использование метода для заполнения данных об автомобиле

array[i].print(); // Вывод введенных данных

}

int inputYear; // В файл будут записываться только те машины, год которых выше этого

cout << "Enter the year: ";

cin >> inputYear;

int zarray[50]; // Массив, хранящий индексы элементов, которые не нужно добавлять в файл

int k = 0;

for (int i = 0; i < car\_count - 1; i++)

{

for (int j = i + 1; j < car\_count; j++)

{

if (array[i].year < inputYear)

{

zarray[k] = i;

k += 1;

// Цикл сравнивает года выпуска автомобилей и добавляет в массив zarray

// индексы авто у которых год выпуска меньше чем inputYear

}

}

}

int zMark; // Хранит индекс элемента с указанной в inputMark маркой,

string inputMark; // Автомобиль который нужно поставить в начало файла

cout << "Enter the mark: ";

cin >> inputMark;

for (int i = 0; i < car\_count; i++) {

if (array[i].mark == inputMark) {

zMark = i; // В цикле перебираются все элементы для поиска указанной в inputMark марки

}

}

string path = "file.txt"; // Включение файла для записи

ofstream file;

file.open(path);

if (!file.is\_open())

{

cout<<"Error opening the file"<<endl;

}

else

{

// Запись в файл первой машины с маркой указанной в inputMark

file << array[zMark].mark << " " << array[zMark].year << " " << array[zMark].price << " " << array[zMark].color << endl;

for (int i = 0; i < car\_count; i++)

{

bool flag = true;

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

{

if (i == zarray[j] || i == zMark)

{

flag = false;

// Если индекс элемента находится в zarray, флаг переворачивается, чтобы не записывать элемент в файл

}

}

if (flag)

{

file << array[i].mark << " " << array[i].year << " " << array[i].price << " " << array[i].color << endl;

// Если флаг не перевернулся, элемент записывается в файл

}

}

}

file.close();

// Вывод в консоль содержимого файла

ifstream readFile;

readFile.open(path);

if (!readFile.is\_open())

{

cout << "Error opening the file" << endl;

}

else

{

string str;

while (!readFile.eof())

{

str = "";

getline(readFile, str);

cout << str << endl;

}

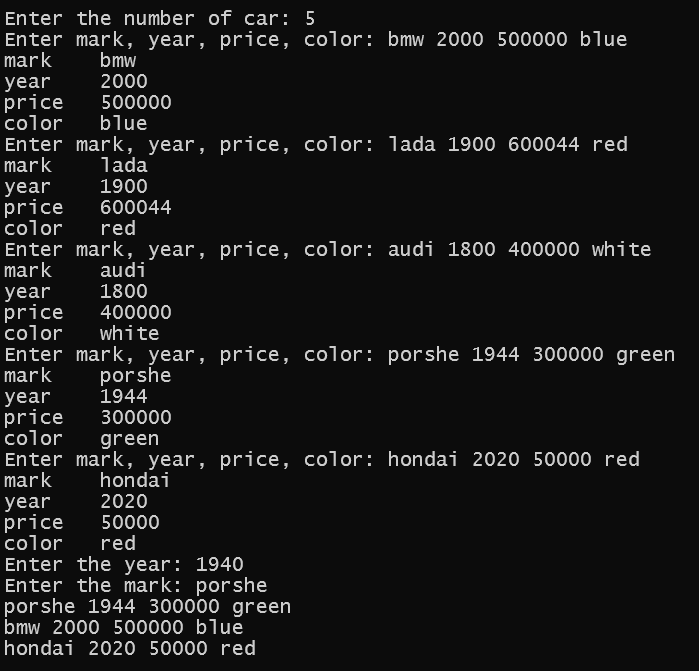
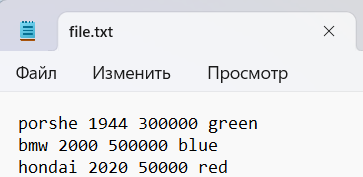
}

readFile.close();

return 0;

}

**Результат работы алгоритма:**

** **

**Скриншот с Github**