**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Лабораторная работа №2

по дисциплине «Программирование»

Структурированный тип

Группа: АВТ-041

Студенты: Антонов А.

Преподаватель: Балакин В.

НОВОСИБИРСК 2021

**Задача**

Регистрационный номер автомобиля, марка, пробег.

**Проектирование программы**

Идея: Структурированный принцип построения системы состоит в том, что более сложные компоненты составляются из более простых, ранее определенных.

Отдельные части программы являются относительно независимыми подзадачами и оформляются в виде модулей (функций) с последующим вызовом из главной функции.

Переменные:

reg\_num - регистрационный номер

marka - марка

probeg - пробег

mas - массив

er - вывод функционала

sw - переключатель для выбора выводить все записи или одну

k - номер структуры, кот. надо вывести

d - номер записи, которую нужно удалить

c - номер записи, которую нужно изменить

per - вывод функционала

**Текст программы с комментариями**

// Лаба2.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

struct avto //объявляем шаблон структуры

{

char reg\_num[30]; //Регистрационный номер

char marka[30]; //Марка

char probeg[30]; //Пробег

};

struct avto mas[30]; //объвляем глобальный массив структур

avto tmp; //объявляем временую переменную структурного типа

int sch = 0; //Счетчик полных записей

int er; //Переключатель

void enter\_new();

int menu();

void out();

void del();

void change();

void find();

int main()

{

setlocale(LC\_ALL, "Russian");

system("chcp 1251");

system("cls");

while (1)

{

switch (menu())

{

case 1:enter\_new(); break;

case 2:find(); break;

case 3:change(); break;

case 4:del(); break;

case 5:out(); break;

case 6: return 0;

default: printf("Не верный выбор\n");

}

}

}

void enter\_new() // ф-ция ввода новой структуры

{

if (sch < 30) //вводим новую запись только, если счетчик полных записей меньше максимального количества записей

{

int chet = sch + 1;

printf("Запись номер %d", chet); //выводим номер записи

printf("\nРегистрационный номер: \n");

scanf\_s("%s", &mas[sch].reg\_num, 30);

printf("Введите марку \n");

scanf\_s("%s", &mas[sch].marka, 30);

printf("Введите пробег\n");

scanf\_s("%s", &mas[sch].probeg, 30);

sch++; //увеличиваем счетчик полных записей на единицу

}

else printf("Введено максимальное кол-во записей ");

}

int menu()

{

int er;

printf("Введите:\n");

printf("1-для ввода новой записи\n");

printf("2-для поиска\n");

printf("3-для изменения записи\n");

printf("4-для удаления записи\n");

printf("5-для вывода записи(ей) \n");

printf("6-для выхода\n");

scanf\_s("%d", &er);

return er;

}

void out() //ф-ция вывода записей

{

int sw; // переключатель для выбора выводить все записи или одну

int k; //номер структуры, кот. надо вывести

if (sch == 0) //если счетсик количества структур равен 0, то выводим, что нет записей

printf("\nНет записей: \n");

else

{

printf("\nВведите: \n");

printf("1-если хотите вывести какую-либо запсь\n");

printf("2-если хотите вывести все записи\n");

scanf\_s("%d", &sw);

if (sw == 1)

{

printf("Введите номер записи, которую нужно вывести\n");

scanf\_s("%d", &k);

printf("\n");

printf("Регистрационный номер: %s\n", mas[k - 1].reg\_num);

printf("Марка: %s\n", mas[k - 1].marka);

printf("Пробег: %s\n", mas[k - 1].probeg);

printf("\_\_\_\_\_\_\_\_\_\_\_\_\n");

}

if (sw == 2)

{

for (int i = 0; i < sch; i++) //выводим в цикле все записи

{

printf("Регистрационный номер: %s\n", mas[k - 1].reg\_num);

printf("Марка: %s\n", mas[k - 1].marka);

printf("Пробег: %s\n", mas[k - 1].probeg);

printf("\_\_\_\_\_\_\_\_\_\_\_\_\n");

}

}

}

}

void del() //ф-ция удаления записи

{

int d; //номер записи, которую нужно удалить

printf("\nВведите номер записи, которую необходимо удалить\n");

printf("Если необходимо удалить все записи, введите '99'\n");

scanf\_s("%d", &d);

if (d != 99)

{

for (int i = (d - 1); i < sch; i++) //цикл для удаления заданной записи, начинаем цикл с удаляемой записи

mas[i] = mas[i + 1]; //замещаем текущую запись следующей за ней

sch = sch - 1; //уменяьшаем счетчик полных записей на 1

}

if (d == 99)

{

for (int i = 0; i < 30; i++)//цикл по все записям от первой до 30-ой

mas[i] = tmp; //замещаем каждую структуру в массиве пустой структурой

sch = 0; //счетчик структур обнуляем, т.к. все записи удалены

}

}

void change() //функция для изменения записи

{

int c; //номер записи, которую нужно изменить

int per;

printf("\nВведите номер записи\n");

scanf\_s("%d", &c);

do

{

printf("Введите: \n");

printf("1-для изменения регистрационного номера\n");

printf("2-для изменения марку\n");

printf("3-для изменения пробега\n");

printf("6-для прекращения\n");

scanf\_s("%d", &per);

switch (per)

{

case 1: printf("\nВведите новый регистрационный номер\n");

scanf\_s("%s", &mas[c - 1].reg\_num);

break;

case 2:

printf("Введите новую марку \n");

scanf\_s("%s", &mas[c - 1].marka);

break;

case 3:

printf("Введите новый пробег\n");

scanf\_s("%s", &mas[c - 1].probeg);

break;

case 6: return;

}

} while (1);

}

void find() //ф-ция поиска записей

{

int sw; // переключатель

if (sch == 0)

printf("\nНет записей: \n");

else

{

printf("\nВведите: \n");

printf("1-один из всех\n");

printf("2-все работы\n");

scanf\_s("%d", &sw);

for (int i = 0; i < sch; i++) //в цикле просматриваем все структуры из массива структур

if (i == sw)

{

if (i == 0)

printf("Ремонт\n");

else

printf("Регистрационный номер: %s\n", mas[i].reg\_num);

printf("Марка: %s\n", mas[i].marka);

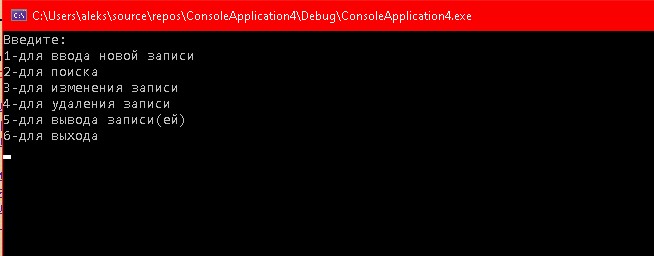
printf("Пробег %s\n", mas[i].probeg);

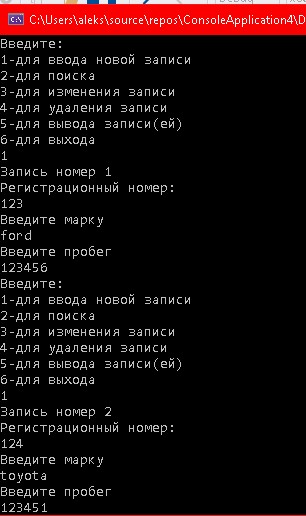
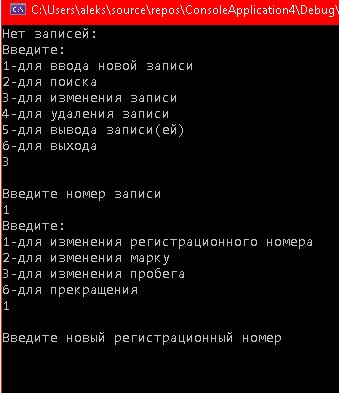
printf("\_\_\_\_\_\_\_\_\_\_\_\_\n");

}

}

}

**Пример работы программы**



**Вывод:**  
Функция была реализована, программа работает корректно.