|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | | |
|  | | | |
| Кафедра теоретической и прикладной информатики | | | |
|  | | | |
| Лабораторная работа № 4 | | | |
| по дисциплине «Структуры данных и алгоритмы» | | | |
| **Управление таблицами** | | | |
|  | | | |
|  | Группа | ПМИ-03 |
| Бригада | 9 |
|  | Малыгин Сергей |
|  | Сидоров Даниил |
|  |  |
| Преподаватели | Еланцева Ирина леонидовна  Тракимус Юрий Викторович |
| Дата | 31.05.2021 |
|  | | | |
| Новосибирск | | | |

1. **Условие задачи**

В файле “owners.txt” дана информация об автомобилях: <номер а/м>, <марка а/м>, <ФИО владельца>. Создать таблицу “owners”, с элементами такого же вида. Таблица должна быть:

б) Упорядочена по возрастанию <ФИО владельца>.

е) Создать таблицу предполагаемых владельцев угнанной машины, о которой известна только марка. Элемент таблицы: <номер>, <ФИО владельца>.

1. **Анализ задачи**

**Входные данные:** Данные из файла“owners” и “stolen”.

**Выходные данные:** Упорядоченная таблица автовладельцев и таблица предполагаемых владельцев угнанной машины.

**Решение:** Создадим таблицу“owners”, заполняя её данными из файла owners.txt., с информацией о номере и марке машины, а также фамилией владельца. Во время заполнения будем проверять, где должна стоять новая строка (Сравниваем по ФИО). Сместим таблицу на одну строку вниз, чтобы освободить место для новой строки и записать её. Если новая строка не должна стоять выше других, то просто запишем ее в конец таблицы. Возьмем марку угнанной машины из файла stolen.txt. Если марка машины совпадает с маркой угнанной машины, то запишем её номер и фамилию владельца в таблицу предполагаемых владельцев угнанной машины. Выведем обе таблицы на консоль.

1. **Структура входных и выходных данных**

**Внешнее представление входных данных:**

Данные из файла owners.txt и stolen.txt.

**Внутреннее представление входных данных:**

Таблица в статической памяти для хранения данных автовладельцев и переменная brand для хранения марки машины.

struct owners

{

UINT number;

char name[30];

char brand[10];

};

struct ownerstable

{

struct owners S[N];

UINT n;

}T;

**Внешнее представление выходных данных:**

Упорядоченная таблица автовладельцев и созданная таблица предполагаемых владельцев угнанной машины. Обе выведены на консоль.

**Внутреннее представление выходных данных:**

struct owners

{

UINT number;

char name[30];

char brand[10];

};

struct ownerstable

{

struct owners S[N];

UINT n;

}T;

struct stolencar

{

UINT number;

char name[30];

};

struct stolencartable

{

struct stolencar Q[M];

UINT m;

}P;

1. **Алгоритм**

константа N = 20;

константа M = N;

константа K = 10;

константа L = 30;

структура owners

{

целая number;

символьная name[L];

символьная brand[K];

};

структура ownerstable

{

структура owners S[N];

целая n;

}T;

структура stolencar

{

целая number;

символьная name[L];

};

структура stolencartable

{

структура stolencar Q[M];

целая m;

}P;

Главная программа

{

Подключение русского языка;

целая number;

символьная name[L], brand[K];

открыть файл("owners.txt");

ввод >> T.S[0].number >> T.S[0].brand >> T.S[0].name;

T.n = 1;

логическая insert = false, limit = true;

пока (пока файл не пуст && limit)

{

если (T.n < N)

{

ввод >> number >> brand >> name;

insert = false;

for (целая i = 0; i < T.n && !insert; i++)

{

если (strcmp(T.S[i].name, name) > 0)

{

for (целая j = T.n-1; j >= i; j--)

{

strcpy\_s(T.S[j+1].name, T.S[j].name);

strcpy\_s(T.S[j+1].brand, T.S[j].brand);

T.S[j+1].number = T.S[j].number;

}

strcpy\_s(T.S[i].name, name);

strcpy\_s(T.S[i].brand, brand);

T.S[i].number = number;

insert = true;

T.n++;

}

}

если (!insert)

{

T.S[T.n].number = number;

strcpy\_s(T.S[T.n].name, name);

strcpy\_s(T.S[T.n].brand, brand);

T.n++;

}

}

Иначе

{

вывод << "В таблице больше " << N << " элементов, взяты первые " << N << " строк!" << endl;

limit = false;

}

}

Закрыть файл();

вывод << "Упорядоченная таблица по ФИО:" << endl;

for (целая i = 0; i < T.n; i++)

{

вывод << T.S[i].number << " " << T.S[i].brand << " " << T.S[i].name << endl;

}

Открыть файл("stolen.txt");

ввод >> brand;

Закрыть файл();

P.m = 0;

for (целая i = 0; i < T.n; i++)

{

если (strcmp(T.S[i].brand, brand) == 0)

{

P.Q[P.m].number = T.S[i].number;

strcpy\_s(P.Q[P.m].name, T.S[i].name);

P.m++;

}

}

вывод << "Предполагаемые владельцы угнанной машины ";

вывод << brand << ": " << endl;

for (целая i = 0; i < P.m; i++)

{

вывод << P.Q[i].number << " " << P.Q[i].name << endl;

}

}

1. **Текст программы**

#include <fstream>

#include <iostream>

#include <windows.h>

using namespace std;

const size\_t N = 20;

const size\_t M = N;

const size\_t K = 10;

const size\_t L = 30;

struct owners

{

UINT number;

char name[L];

char brand[K];

};

struct ownerstable

{

struct owners S[N];

UINT n;

}T;

struct stolencar

{

UINT number;

char name[L];

};

struct stolencartable

{

struct stolencar Q[M];

UINT m;

}P;

int main()

{

setlocale(LC\_ALL, "rus");

ifstream fin;

UINT number;

char name[L], brand[K];

fin.open("owners.txt");

fin >> T.S[0].number >> T.S[0].brand >> T.S[0].name;

T.n = 1;

bool insert = false, limit = true;

while (!fin.eof() && limit)

{

if (T.n < N)

{

fin >> number >> brand >> name;

insert = false;

for (int i = 0; i < T.n && !insert; i++)

{

if (strcmp(T.S[i].name, name) > 0)

{

for (int j = T.n-1; j >= i; j--)

{

strcpy\_s(T.S[j+1].name, T.S[j].name);

strcpy\_s(T.S[j+1].brand, T.S[j].brand);

T.S[j+1].number = T.S[j].number;

}

strcpy\_s(T.S[i].name, name);

strcpy\_s(T.S[i].brand, brand);

T.S[i].number = number;

insert = true;

T.n++;

}

}

if (!insert)

{

T.S[T.n].number = number;

strcpy\_s(T.S[T.n].name, name);

strcpy\_s(T.S[T.n].brand, brand);

T.n++;

}

}

else {

cout << "В таблице больше " << N << " элементов, взяты первые " << N << " строк!" << endl;

limit = false;

}

}

fin.close();

cout << "Упорядоченная таблица по ФИО:" << endl;

for (UINT i = 0; i < T.n; i++)

{

cout << T.S[i].number << " " << T.S[i].brand << " " << T.S[i].name << endl;

}

fin.open("stolen.txt");

fin >> brand;

fin.close();

P.m = 0;

for (UINT i = 0; i < T.n; i++)

{

if (strcmp(T.S[i].brand, brand) == 0)

{

P.Q[P.m].number = T.S[i].number;

strcpy\_s(P.Q[P.m].name, T.S[i].name);

P.m++;

}

}

cout << "Предполагаемые владельцы угнанной машины ";

cout << brand << ": " << endl;

for (UINT i = 0; i < P.m; i++)

{

cout << P.Q[i].number << " " << P.Q[i].name << endl;

}

}

**6. Набор тестов**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *№* | *Входные данные* | | *Выходные данные* | *Примечание* |
| *Файл owners.txt* | *Файл stolen.txt* |
| *1* | 123 KIA Sidorov  345 BMW Ivanov  404 FORD Petrov  256 GAZ Acinov  300 Mercedez Boosin  204 Toyota Corov  250 Honda Davidov  799 Nissan Jokov  686 Hondai Worotov  999 Ferrari Yaniv  101 Ferrari Kokarev  121 Ferrari Maneskin  412 Toyota Washington  888 GAZ Lobanov  777 BMW Ugodov  666 BMW Akarapev | BMW | *Сортировка по ФИО:*  *256 GAZ Acinov*  *666 BMW Akarapev*  *300 Mercedez Boosin*  *204 Toyota Corov*  *250 Honda Davidov*  *345 BMW Ivanov*  *799 Nissan Jokov*  *101 Ferrari Kokarev*  *888 GAZ Lobanov*  *121 Ferrari Maneskin*  *404 FORD Petrov*  *123 KIA Sidorov*  *777 BMW Ugodov*  *412 Toyota Washington*  *686 Hondai Worotov*  *999 Ferrari Yaniv*  *Предполагаемые владельцы угнанной машины BMW:*  *666 Akarapev*  *345 Ivanov*  *777 Ugodov* | *Программа работает верно.* |
| *2* | 123 KIA Sidorov  345 BMW Ivanov  404 FORD Petrov  256 GAZ Acinov  300 Mercedez Boosin |  | *Сортировка по ФИО:*  *256 GAZ Acinov*  *300 Mercedez Boosin*  *345 BMW Ivanov*  *404 FORD Petrov*  *123 KIA Sidorov*  *Предполагаемые владельцы угнанной машины :* | *Нет марки угнанной машины, соответственно вторая таблица пуста.* |
| *3* | 204 Toyota Corov  250 Honda Davidov  799 Nissan Jokov  686 Hondai Worotov  999 Ferrari Yaniv  101 Ferrari Kokarev | Ferrari  Honda | *Упорядоченная таблица по ФИО:*  *204 Toyota Corov*  *250 Honda Davidov*  *799 Nissan Jokov*  *101 Ferrari Kokarev*  *686 Hondai Worotov*  *999 Ferrari Yaniv*  *Предполагаемые владельцы угнанной машины Ferrari:*  *101 Kokarev*  *999 Yaniv* | *Даны две марки угнанной машины. Учитывается только первая.* |
| *4* | 123 KIA Sidorov  345 BMW Ivanov  404 FORD Petrov  256 GAZ Acinov  300 Mercedez Boosin  204 Toyota Corov  250 Honda Davidov  799 Nissan Jokov  686 Hondai Worotov  999 Ferrari Yaniv  101 Ferrari Kokarev  121 Ferrari Maneskin  412 Toyota Washington  888 GAZ Lobanov  777 BMW Ugodov  666 BMW Akarapev  436 Ferrari Konovalov  467 KIA Horonov  869 Toyota Jovanov  256 FORD Imanbek  763 Nissan Valeev  367 Honda Borisenko | Ferrari | *В таблице больше 20 элементов, взяты первые 20 строк!*  *Упорядоченная таблица по ФИО:*  *256 GAZ Acinov*  *666 BMW Akarapev*  *300 Mercedez Boosin*  *204 Toyota Corov*  *250 Honda Davidov*  *467 KIA Horonov*  *256 FORD Imanbek*  *345 BMW Ivanov*  *799 Nissan Jokov*  *869 Toyota Jovanov*  *101 Ferrari Kokarev*  *436 Ferrari Konovalov*  *888 GAZ Lobanov*  *121 Ferrari Maneskin*  *404 FORD Petrov*  *123 KIA Sidorov*  *777 BMW Ugodov*  *412 Toyota Washington*  *686 Hondai Worotov*  *999 Ferrari Yaniv*  *Предполагаемые владельцы угнанной машины Ferrari:*  *101 Kokarev*  *436 Konovalov*  *121 Maneskin*  *999 Yaniv* | *В таблице больше 20 элементов, взяты первые 20 строк!* |
| *5* |  | Ferrari | *Упорядоченная таблица по ФИО:*  *0*  *Предполагаемые владельцы угнанной машины Ferrari:* | *Файл с автовладельцами пуст* |

1. **Результаты работы программы**

Программа работает правильно, что подтверждают тесты.