**Министерство образования Республики Беларусь**

**Учреждение образования**

**«Белорусский государственный университет**

**информатики и радиоэлектроники»**

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

**Отчет**

**По лабораторной работе №6**

По дисциплине: Аппаратные основы интеллектуальных систем

На тему: Моделирование хэш-таблиц.

Выполнил:                                   Щур А. А.

Группа 821703

Проверил:                                    Захаров В.В.

**Минск 2019**

**Задача:**

Разработать и проверить программу, обеспечивающую формирование хеш-таблицы по ключевым словам и выполнение различных операций с этой таблицей – включение в таблицу новых строк, поиск информации в таблице по ключевым словам, удаление строк из таблицы.

**Вариант 28**

**Задание**

Тема таблицы – транспорт.

**Ход и результаты работы**

**Ход работы:**

Каждому ключевому слову в таблице, являющимся названием биологического вида, ставится в соответствие название царства, к которому принадлежит данный биологический вид.

Таблица содержит следующие поля:

1. id – ключевое слово
2. value – сумма номеров первых n символов
3. hash\_code – хеш-код
4. collision – флажок коллизии
5. unaval – флажок занято
6. terminal – терминальный флажок
7. delining – флажок вычеркивания
8. next – указатель на следующий элемент в области коллизии
9. data – данные

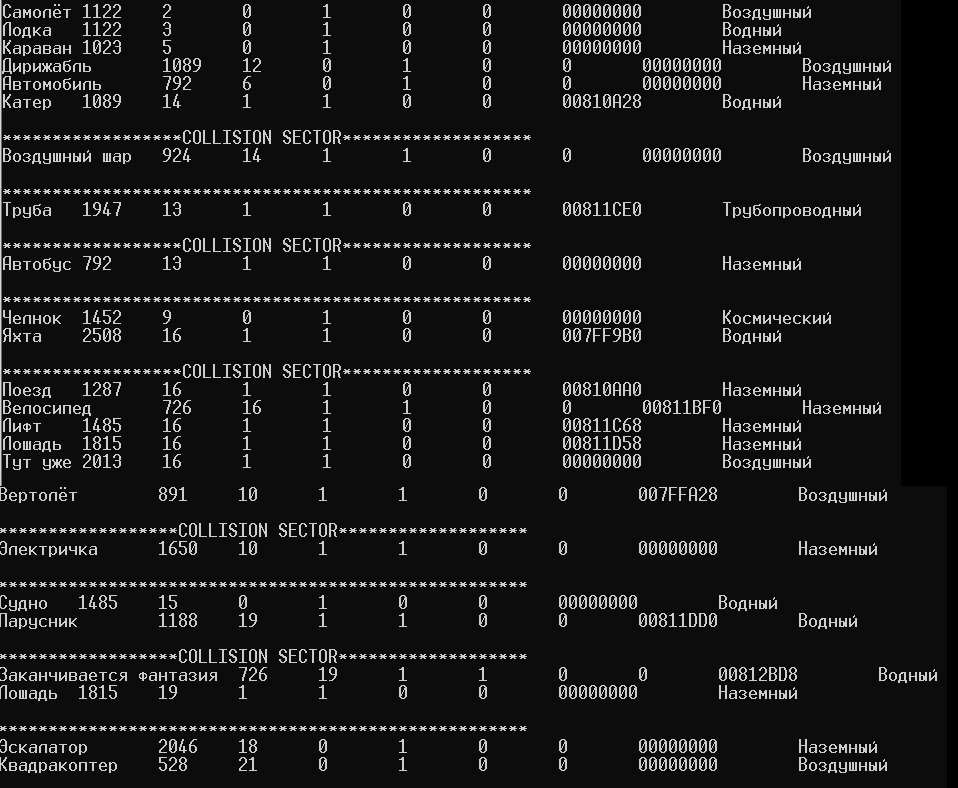
Программа содержит следующие 7 функций:

1. alpha() – функция возвращает порядковый номер в алфавите символа, переданного ей
2. get\_value() – функция возвращает целочисленное число value – сумма 3 чисел, соответствующих номерам первых трех символов строки-ключа в алфавите. Если в строке меньше 3 чисел, берутся значения всех символов строки.
3. get\_hash() – функция возвращает числовое значение val, вычисленное по формуле val = V % 10 + B, где V – значение, вычисленное функцией get\_value(), В – размер хеш-таблицы. Значение val используется в качестве хеш-ключа
4. add\_el() – функция, осуществляющая вставку в таблицу пары элемент-значение. По результатам работы функции get\_hаsh() проводится анализ наличия коллизии: если коллизии не наблюдается – пара вставляется в следующую ячейку хеш-таблицы, иначе находится последний элемент с таким же хещ-ключом и данная пара заносится в ячейку next последнего элемента с таким же хеш-ключом. Все соответствующие данной паре флажки и значения также заносятся в таблицу согласно наличию/отсутствию коллизии.
5. del\_el() – функция, осуществляющая удаление элемента из хеш-таблицы по переданному в данную функцию ключу. Отыскивается ячейка таблицы с переданным ключом и производится операция удаления данной ячейки. Если элемент находится в области коллизии, вся цепочка “разрывается” по данному элементу, элемент удаляется из цепочки и цепочка снова склеивается. Иначе ячейка просто извлекается из таблицы.
6. print\_search\_el() – функция, выводящая на экран некоторые данные, соответствующие ключевому слову, переданному в функцию. Вычисляется хеш-код данного ключевого слова и по хеш-коду осуществляется поиск и вывод на экран соответствующих данных.
7. print\_table() – функция, осуществляющая вывод на экран всей хеш-таблицы. Область коллизии выделяется следующей рамкой:

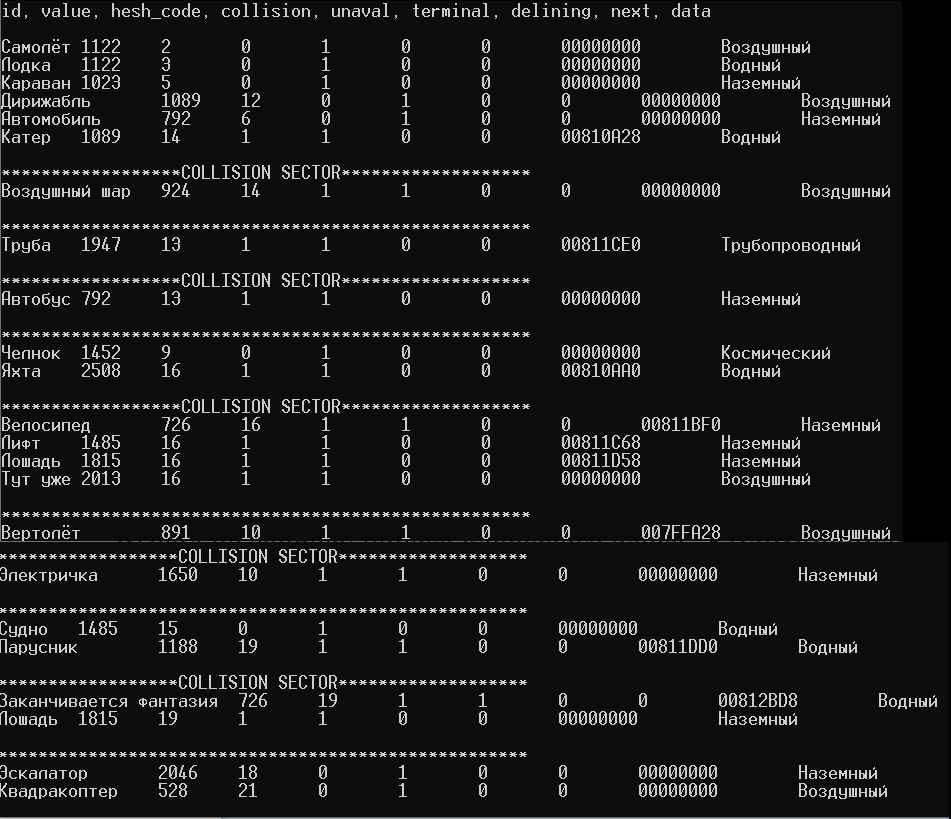
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*COLLISION SECTOR\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Вид первоначальной таблицы, содержащей 20 элементов:



Вид таблицы после удаления ячейки с ключевым словом “Поезд”



Вывод по запросу поиска ключевого слова «Самолёт»:



**Выводы:**

Хеш-таблица – это структура данных, позволяющая хранить пары (ключ - значение) и выполнять операции добавления новой пары, поиска и удаления пары по ключу. Хеш таблицы используются, например, для хранения истории посещения сайтов в браузере, для хеширования и хранения данных пользователей на различных ресурсах и сайтах, в аппаратных маршрутизаторах и т.д.

Хеширование может осуществляться различными способами. Если хеш-ключи известны заранее, то можно избежать коллизий, найдя для них совершенную хеш-функцию.

Ситуация, когда для различных ключей получается одно и то же хеш-значение, называется коллизией. В этом случае для размещения второй и последующих записей необходимо использовать резервные ячейки памяти, которые размещаются либо в самой таблице хеширования (внутренняя адресация), либо в специальной области памяти (области переполнения).

Для поиска свободных резервных ячеек при внутренней адресации применяется процедура пробинга (линейного, квадратичного или случайного).