Министерство образования Республики Беларусь

Учреждение образования БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления Кафедра Интеллектуальных информационных технологий

## ОТЧЁТ

по лабораторной работе №6

по дисциплине «Аппаратное обеспечение интеллектуальных систем» на тему:

«Моделирование хеш-таблиц»

Вариант 10

Выполнила: Д. В. Демидовец

Студент группы 221703

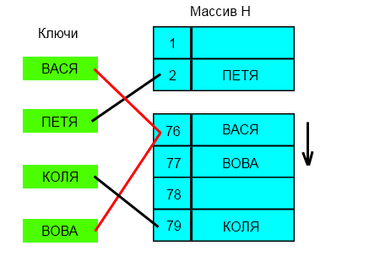
Проверил: Е. А. Казаченко

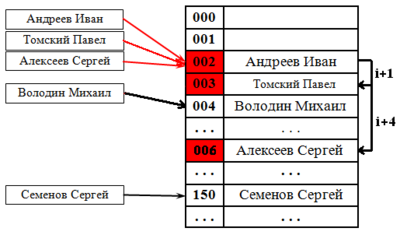
Минск 2024

**Цель**: освоение навыков построения и проверки таблиц хеширования.

**Задание:** реализовать хеш-таблицу, которая должна реализовывать CRUD операции и поддерживать решение проблемы коллизии в соответствии с вариантом.

Мой вариант - Линейное разрешение коллизий: Квадратичный поиск

*Линейное разрешение коллизий* подразумевает, что все элементы хранятся непосредственно в хеш-таблице, без использования связных списков. При использовании этого метода может возникнуть ситуация, когда хеш-таблица окажется полностью заполненной, следовательно, будет невозможно добавлять в неё новые элементы. Так что при возникновении такой ситуации решением может быть динамическое увеличение размера хеш-таблицы, с одновременной её перестройкой.



В *квадратичном поиске* мы имеем нефиксированный шаг, который изменяется квадратично: q=1,4,9,16. Соответственно, при попытке добавить элемент в занятую ячейку 𝑖 начинаем последовательно просматривать ячейки 𝑖+1,𝑖+4,𝑖+9 и так далее, пока не найдём свободную ячейку.

**Демонстрация работы программы**

Для заполнения хэш-таблицы выбрана тема литературы, соответственно *ключом* являются литературные термины, а *значением* – их определения.

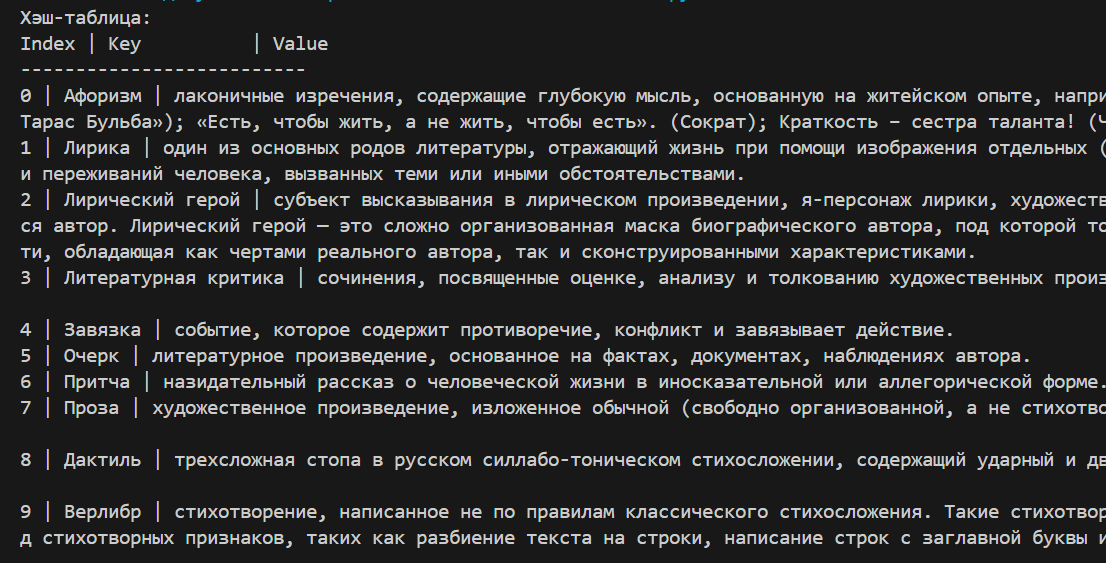
В качестве хэш-функции взято выражение:

hash = first\_char\_index \* alphabet\_size + second\_char\_index , где

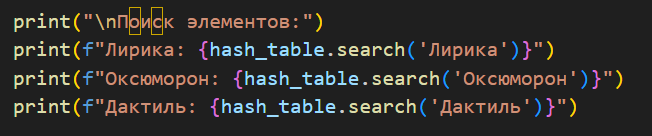
first\_char\_index – номер первой буквы ключа в алфавите,

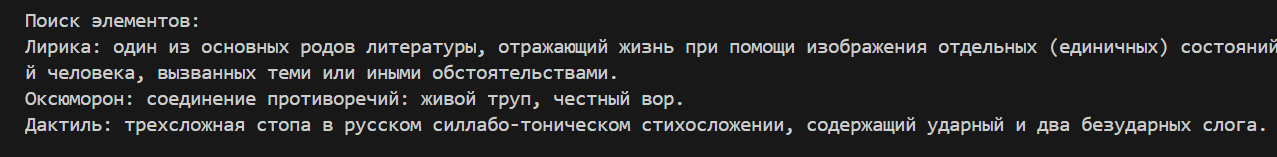
second\_char\_index – номер второй буквы ключа в алфавите.

Вывод построенной хэш-таблицы:

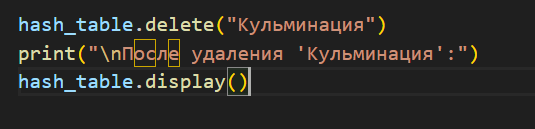


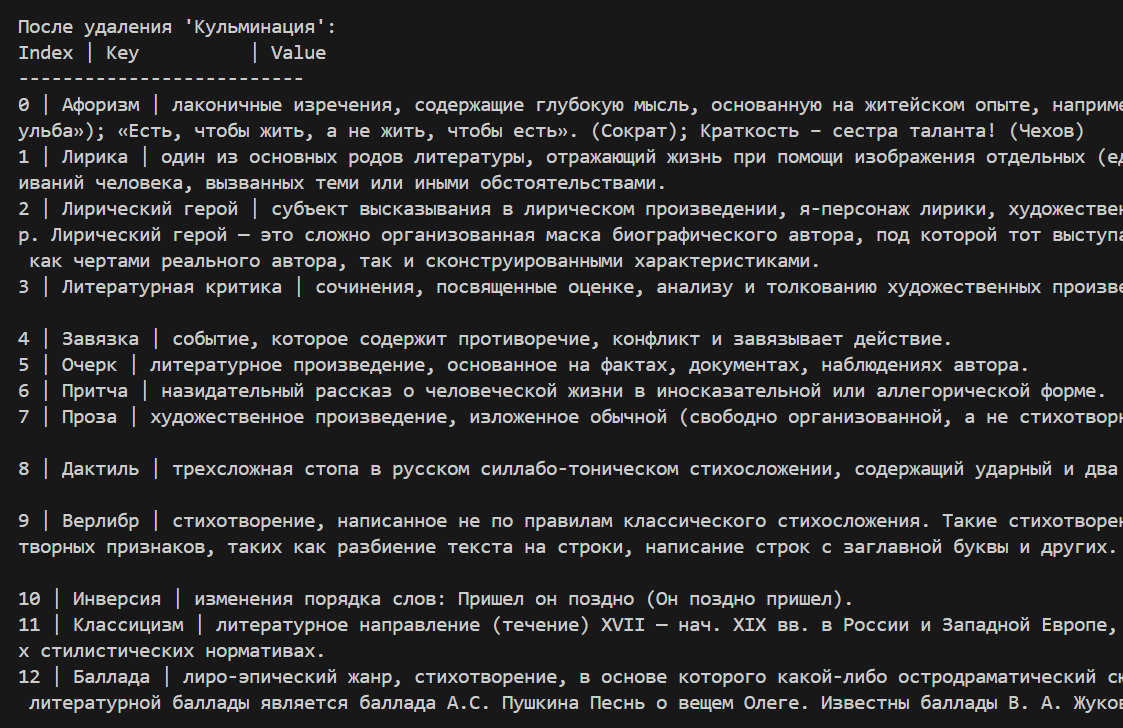
Результаты поиска элементов хэш-таблицы по ключу:

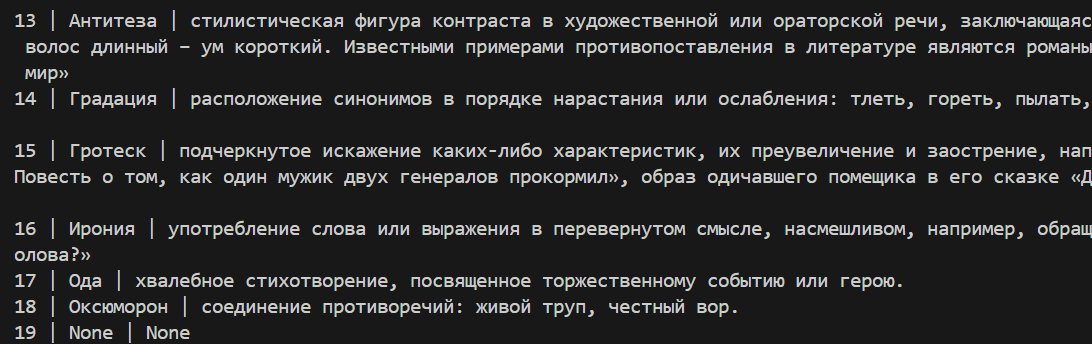




Результат удаления ячейки в хэш-таблице:







**Вывод:** в ходе выполнения лабораторной работы была разработана структура данных хэш-таблица, позволяющая эффективно хранить и обрабатывать большие объемы данных, также учтено выполнение всех CRUD операций, таких как создание, чтение, обновление и удаление элементов в хэш-таблице, для решения коллизий был реализован метод линейного разрешения коллизий. Приобретенные навыки и основы принципа работы с хэш-таблицами могут быть полезны в дальнейших проектах, где требуется эффективное хранение и обработка данных.