Лабораторная работа №2

**Построение словарей на основе метода открытого хеширования данных**

Цель работы: научиться строить словари на базе линейных списков и открытого хеширования данных.

Порядок выполнения работы

1. Ознакомиться с теоретической частью лабораторной работы.

2. Реализовать практическое задание.

3. Оформить отчет по лабораторной работе.

*Задание.*

На основе динамических списков необходимо реализовать словарь.

Основные операции, выполняемые над данными словаря:

- поиск,

- вставка;

- удаление.

В сочетании со списками для построения словарей удобно использовать *открытое хеширование данных*, позволяющее фиксировать время выполнения операторов над словарем, а также сделать *потенциально бесконечным пространство* для хранения данных.

На рис. 1 показана базовая структура данных при открытом хешировании. Основная идея метода заключается в том, что множество данных (возможно, очень большое) разбивается на конечное число классов. Для В классов, пронумерованных от 0 до В-1, строится хеш-функция h такая, что для любого элемента x исходного множества функция h(x) принимает целочисленное значение из интервала 0, …, В-1, которое соответствует классу, которому принадлежит элемент x. Элемент x называют ключом, h(x) – хеш-значением х, а классы – сегментами. Массив (таблица сегментов), проиндексированный номерами сегментов 0, 1, … В-1, содержит заголовки для В списков. Элемент х i-го списка – это элемент исходного множества, для которого h(x)=i.

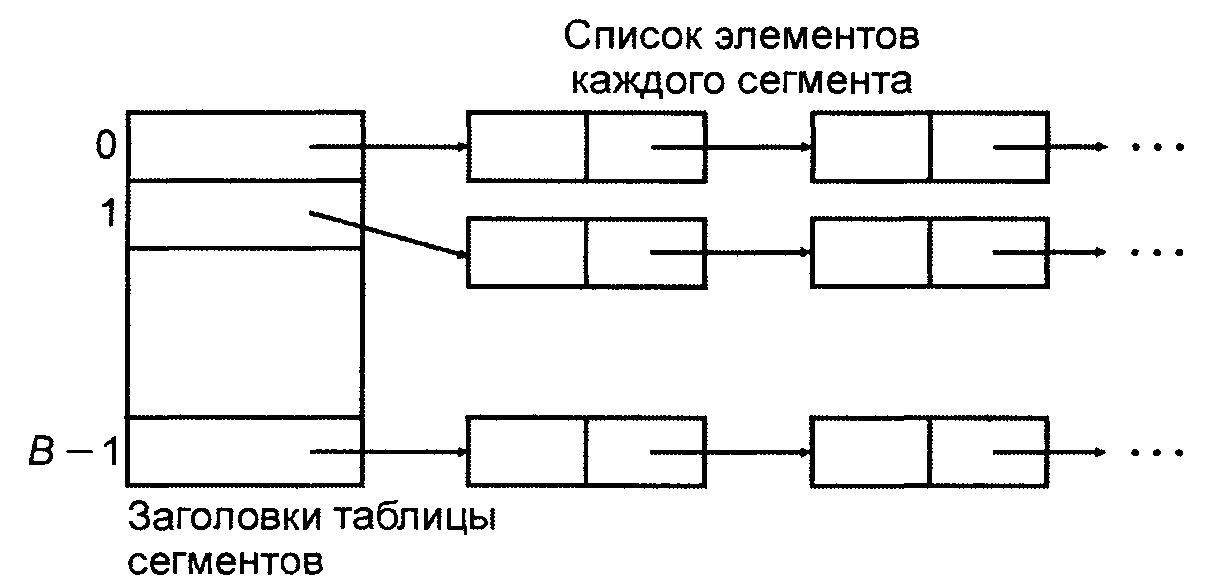


Рис. 1 – Организация данных при открытом хешировании

Если сегменты приблизительно равны по размеру, то в этом случае списки всех сегментов должны быть наиболее короткими при данном числе сегментов. Если исходное множество состоит из N элементов, тогда средняя длина списков будет N/B элементов. Если удается оценить величину N и выбрать B как можно ближе к этой величине, то в каждом списке будет один-два элемента. Тогда время выполнения операций с данными будет малой постоянной величиной, зависящей от N или от В. Однако не всегда ясно, как выбрать хеш-функцию h так, чтобы она примерно поровну распределяла элементы исходного множества по всем сегментам.

Идеальной хеш-функцией является такая, которая для любых двух неодинаковых ключей выдает неодинаковые адреса, т.е.

r:\Temp\Rar$DRa0.289\ЭУМКД_СиАОД\Практика\content\lb2\lb2.files\image004.gif

Однако подобрать такую функцию можно в случае, если все возможные значения ключей известны заранее. Такая организация данных носит название «совершенное хеширование». Если заранее не определено множество значений ключей, и длина таблицы ограничена, подбор совершенной функции затруднителен. Поэтому часто используют хеш-функции, которые не гарантируют выполнение условия (1).

Контрольные вопросы и задания:

1. С какой целью выполняется хеширование данных?

2. Какая хеш-функция является идеальной?

3. В чем отличия между открытым и закрытым хешированием данных?

4. Объясните суть коллизий при закрытом хешировании данных.

5. Перечислите способы борьбы с коллизиями.