Практическая работа №3

Алгоритм хеширования таблиц

Цель: Изучить алгоритмы хеширования таблиц

Хеш-таблица – это структура данных, реализующая интерфейс ассоциативного массива, то есть она позволяет хранить пары вида "ключ- значение" и выполнять три операции: операцию добавления новой пары, операцию поиска и операцию удаления пары по ключу. Хеш-таблица является массивом, формируемым в определенном порядке хеш-функцией.

В качестве использования хеширования в повседневной жизни можно привести примеры распределение книг в библиотеке по тематическим каталогам, упорядочивание в словарях по первым буквам слов.

Доступ к элементам осуществляется по его ключу. Основные операции, которые могут выполняться с хеш-таблицей:

- Insert — добавить новый элемент в хеш-таблицу
- Delete — удалить элемент из хеш-таблицы по ключу
- Search — получить значение по ключу

Основной принцип работы хеш-таблицы заключается в том, что в качестве входных параметров она принимает пары ключ-значение. Затем с помощью специальной хеш-функции получает короткий ключ на основе полученного ключа. И, наконец, добавляет данные в таблицу. Если в таблице уже существует значения с таким хешем, то объединяет их в коллекции. Внутри коллекции поиск выполняется по изначальному полученному ключу.

Код

```
//Хеш - функция
unsigned int hashtab_hash(char* key) {
        unsigned int h = 0; char* p;
        for (p = key; *p != '\0'; p++) {
                h = h * HASHTAB_MUL + (unsigned int)*p; }
        return h % HASHTAB_SIZE;
}
```

Диаграмма деятельности

					АиСД.09.03.02.050000 ПР		
Изм.	*Лист*	*№ докум.*	*Подпись*	*Дата*			
Разраб.		*Ермошина В.А*			Практическая работа №3	*Лит.*	*Лист* 2 / *Листов* 10
Провер.		*Берёза А.Н.*			Алгоритм хеширования таблиц		*ИСОиП (филиал) ДГТУ в г.Шахты ИСТ-Tb21*
Реценз							
Н. Контр.							
Утверд.							

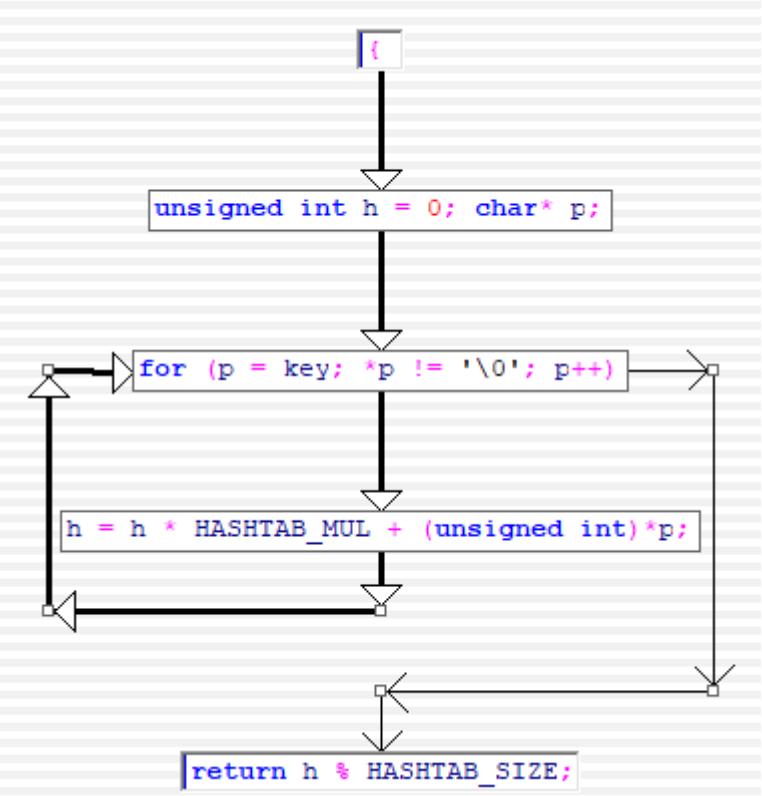


Рисунок 1 - Диаграмма деятельности для функции хеширования

Добавление элемента в хеш-таблицу

При добавлении элементов в хеш-таблицу выделяются куски динамической памяти, которые организуются в виде связанных списков, каждый из которых соответствует входу хеш-таблицы.

```
//Добавление элемента в хеш - таблицу
void hashtab_add(struct listnode * *hashtab, char* key,
        int value) {
        struct listnode* node;
        int index = hashtab_hash(key);
        //Вставка в начало списка
        node = malloc(sizeof(*node));
        if (node != NULL) {
                node->key = key;
                node->value = value;
                node->next = hashtab[index];
                hashtab[index] = node;
```

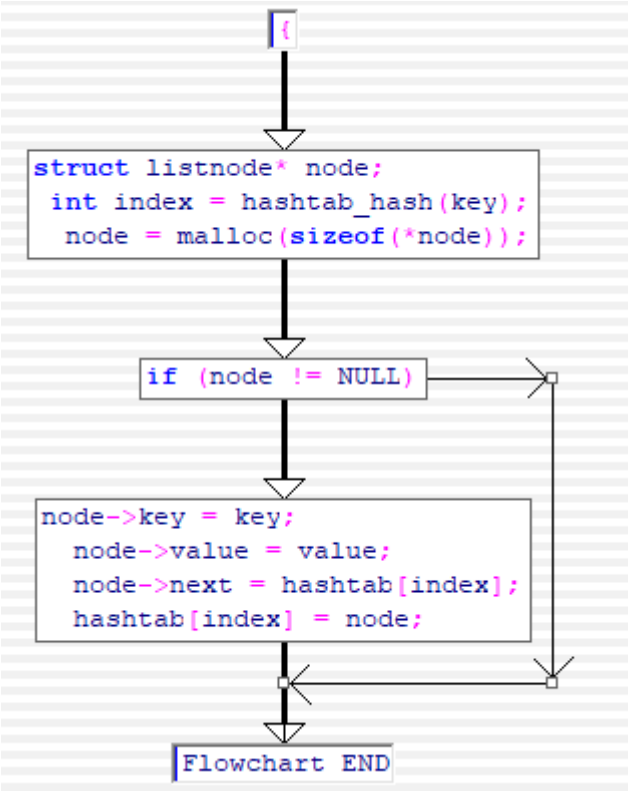
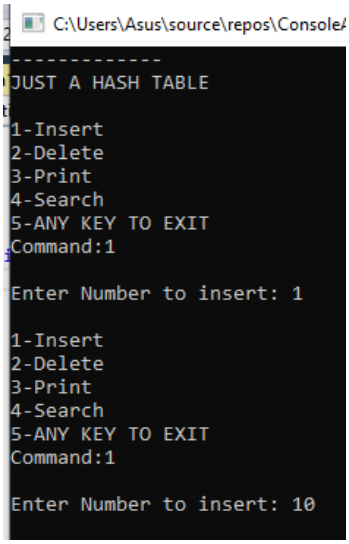


Рисунок 2 - Диаграмма деятельности для добавления элемента в таблицу

Результат



Поиск элемента

```
//Поиск элемента
struct listnode* hashtab_lookup(
        struct listnode** hashtab, char* key);
{
        int index;
        struct listnode* node;
        index = hashtab_hash(key);
        for (node = hashtab[index];
                node != NULL; node = node->next)
        {
                if (strcmp(node->key, key) == 0)
```

```
                return node;
            }
            return NULL;
        }
    }
```

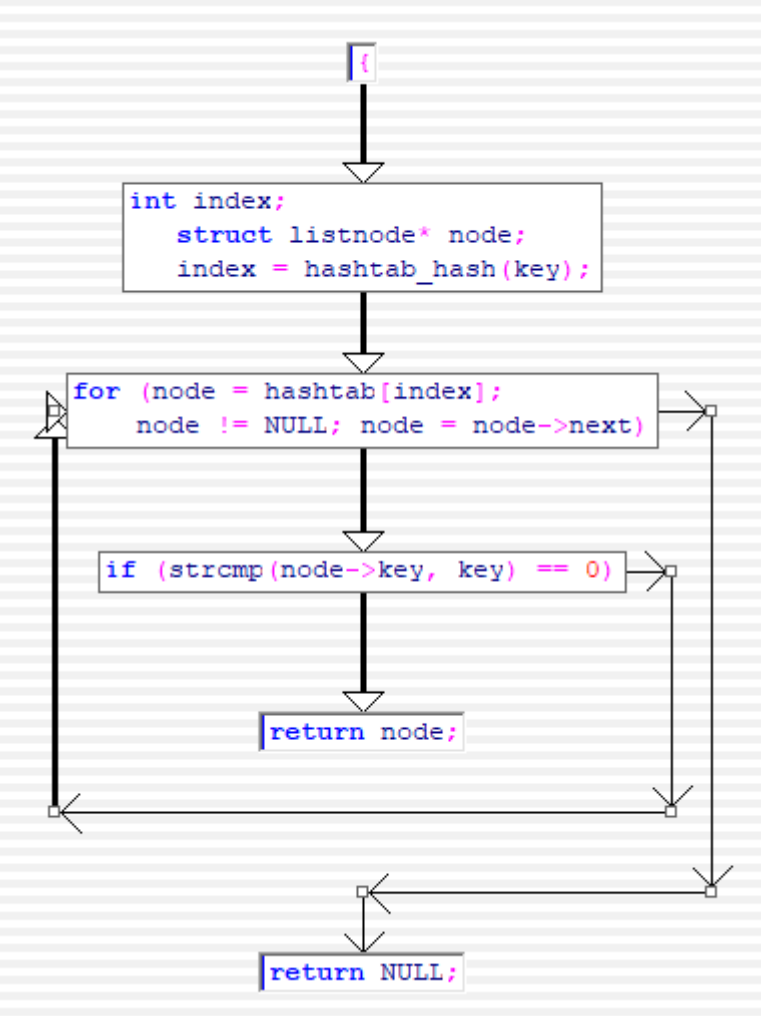
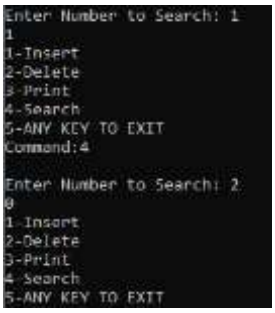


Рисунок 3 - Диаграмма деятельности поиска элемента

Результат



Удаление элемента

```
//Удаление элемента
    void hashtab_delete(struct listnode** hashtab, char* key) {
        int index;
        struct listnode* p, * prev = NULL;
        index = hashtab_hash(key);
```

```
for (p = hashtab[index]; p != NULL; p = p->next) {
        if (strcmp(p->key, key) == 0) {
                {
                        if (prev == NULLhashtab[index] = p->next;
                        else
                        prev->next = p->next;
                        free(p);
                        return; }
        prev = p; } }
```

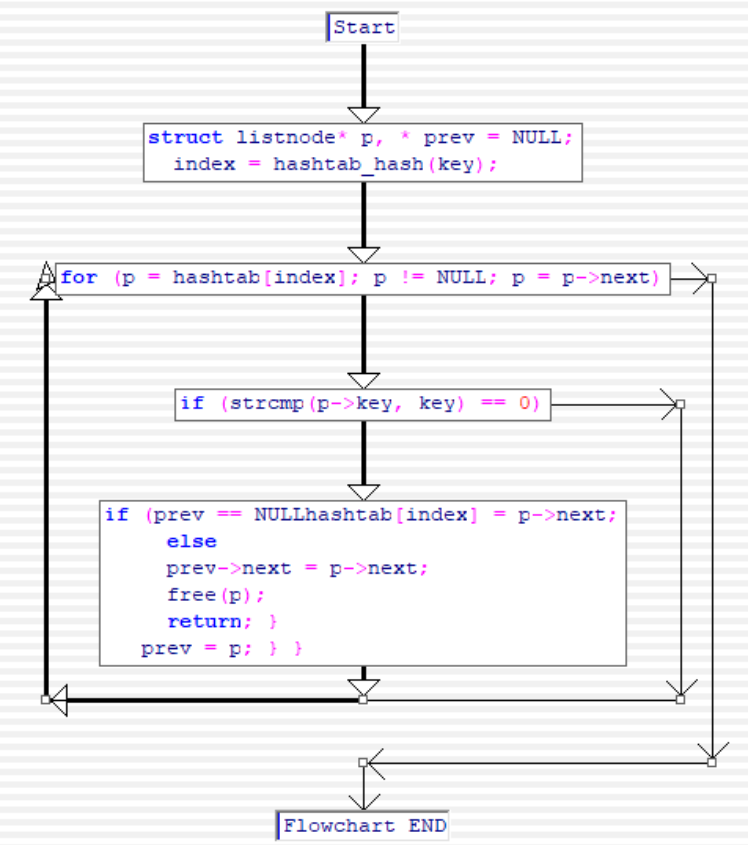


Рисунок 4 - Диаграмма деятельности удаления элемента

Результат

```
Enter Number to delete: 1

1-Insert
2-Delete
3-Print
4-Search
5-ANY KEY TO EXIT
Command:3
0:      10
1:      ----
2:      ----
3:      ----
4:      ----
```

Вывод: в ходе выполнения лабораторной работы изучила алгоритмы хеширования таблиц.