### HashMap 总结

在java及面向对象中，都有自己的hashCode,是区分不同对象的标识。hashCode 都是一个整型。通过哈希函数，可以将字符串或其他类型的key,转化成数组下标index.

Index = HashCode(“this”)%Array.length=3559070%8=6。

写入hashMap.put(“0002931”,”张三”),插入一组key0002931、value 为张三的键值对。

步骤1.通过哈希函数，把key专成数组的下标5.

步骤2.如果数组下标5对应的数组没有元素，则把这个Entry填充到数组下标5的位置。

由于数组的长度是有限的，插入的Entry 原来越多，不同的key 通过哈希函数获得的下标是有可能相同的。这种叫做**哈希冲突**。

哈希冲突解决：

1.通过后移1位，看看下标有空，如果占用，则继续后移。

2.链表法，hashMap数组每一个元素Entry对象，还是链表的头节点。Entry通过next指针指向下一个Entry的节点，当新来的Entry映射与冲突的数组位置时，只需要插入对应的链表中。

### Tree

Linked List 是特殊的Tree

Tree 是特殊的图

二叉树

前序遍历：根->左子树->右子树

中序遍历：左子树->根->右子树

后序遍历：左子树->右子树->根

### 递归

1. 终止条件
2. 处理当前层逻辑
3. 下探到下一层
4. 清理当前层

注意：  
1.不要人肉递归

2.找重复的问题

3.利用数学归纳法