

week02-哈希表，树，递归

哈希表

特性

时间复杂度

注意点

树，二叉树，二叉搜索树

定义

时间复杂度

程序应用

递归

模版

哈希表

特性

通过关键码值来映射到一个位置来访问记录以加快访问速度

此处这个映射函数叫做散列函数，存放记录的数组叫哈希表。

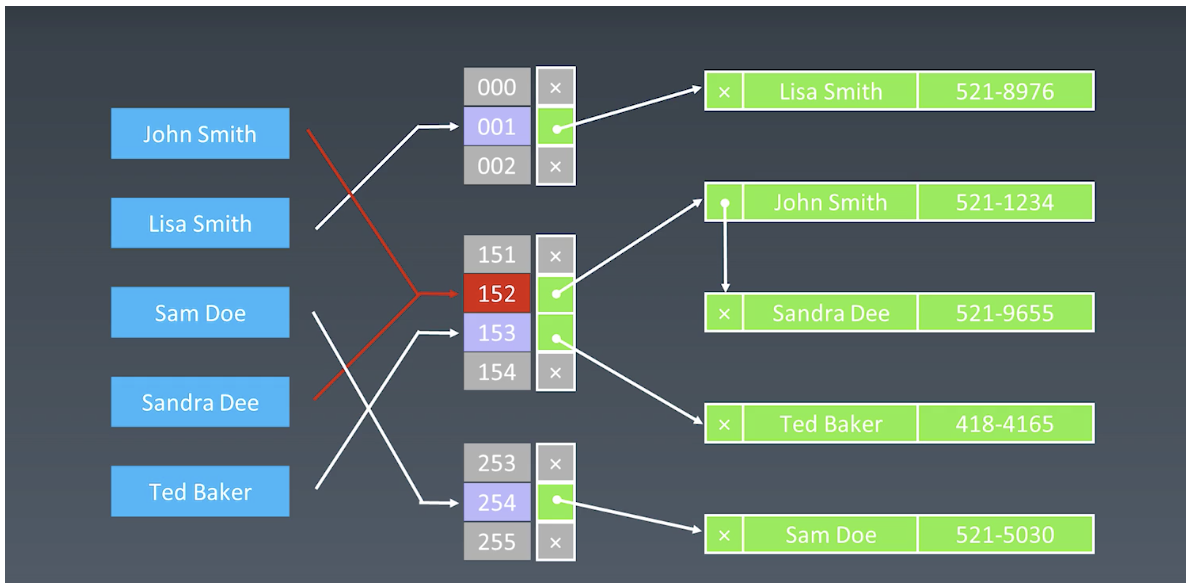
时间复杂度

他的插入，更新，查询的时间复杂度就是 $O(1)$ 的，像redis等会用到大量的hash表来提高性能

注意点

哈希函数的设计要注意哈希碰撞，即对所要存储的值计算出来的hash值一样，这样他们所存放在的数组中的位置是一样的。那在那个位置里，存放的值会以链表的形式存在

如下图所示



树，二叉树，二叉搜索树

定义

二叉树即为，儿子节点只有2个的树

链表即为特殊化的树，树为特殊化的图（树如果形成环就是图）

二叉树的示例代码

```

1 class TreeNode {
2     public int val;
3     public TreeNode left, right;
4
5     public TreeNode(int _val) {
6         val = _val;
7         this.left = null;
8         this.right = null;
9     }
10 };

```

时间复杂度

树：插入是 $O(1)$ ，查询是 $O(n)$

二叉树：插入，更新，删除，查询都是 $O(\log N)$

程序应用

树的解法一半会用递归方式

递归

模版

```
1 public void recur(int level, int param) {  
2  
3     // 终结条件  
4     if (level > MAX_LEVEL) {  
5         // process result  
6         return;  
7     }  
8  
9     // 进入下一个  
10    process(level, param);  
11  
12    // 执行逻辑  
13    recur( level: level + 1, newParam);  
14  
15    // 清除一些状态  
16  
17 }
```