

实验四 散 列

【实验目的】

- (1) 掌握散列表、散列函数与哈希冲突的概念
- (2) 掌握散列表的构造方法及其计算机的表示与实现

【实验内容】

1. 给定输入 {4371, 1223, 6173, 4199, 4344, 9679, 1989} 和散列函数 $h(x) = x \% 11$ 。写程序完成下列不同的哈希表，并用上述数据序列来验证程序：
 - i. 分离链接散列表，必须使用教材的 `SeparateChainingHashTable` 类。
 - ii. 使用线性探测的开放地址散列表， $f(i)=i$ ，要求使用教材提供的 `QuadraticProbingHashTable` 类并进行必要修改而完成。
 - iii. 使用二次探测的开放地址散列表， $f(i) = f(i-1) + 2i - 1$ ，要求使用教材提供的 `QuadraticProbingHashTable` 类。
 - iv. 使用双散列的开放地址散列表， $h_2(x) = 7 - (x \% 7)$ ， $f(i)=i*h_2(x)$ ，要求使用教材提供的 `QuadraticProbingHashTable` 类并进行必要修改而完成。
 - v. 使用再散列的开放地址散列表。
2. 将题 1 中的数据序列分别加入 `HashMap`, `HashSet`, `TreeMap` and `TreeSet` 对象，然后删除 1~2 数据，并且输出这些对象的所有元素。努力理解这两种数据结构：二叉树和哈希表。
3. 编程实现如下功能：
 - (1) 记录的个数为 $n(n \leq 20)$ ，请设计合适的哈希表长度，并用除留余数法构造一个哈希函数
 - (2) 输入哈希表中记录的个数 n 和各记录的关键字值，选择一种合适的冲突解决方法（要求尽量减少冲突），建立一个开放地址哈希表，并输出最终的哈希表。
 - (3) 输入一个待查找记录的关键字 `key`，执行开放地址哈希表的查找操作。如果查找成功，则函数返回查找到的记录在哈希表中的位置值，否则给出查找失败的提示信息。
 - (4) 非编程要求：在 AVL 树、链表中输入与题（2）同样的数据，请画出三种数据结构（包括题（2）的哈希表）的最终状态，分别计算这三种数据结构中的平均查找长度、分析三种数据结构中查找操作的时间复杂度并排序。