Chapter 6 集合与字典

集合

集合：成员（元素）的群集，其中成员可以是原子也可以是集合，集合中的元素必须是互不相同的。

集合的特殊性：

- 集合的成员是无序的
- 集合可以保存实际数据值或者元素是否在集合中的指示信息
- 有重复的元素的情况下可以采用多重集合（包）

集合的基本操作：并，交，差，判存在

位向量实现集合：1表示在集合，0表示不在集合

并查集与等价类

并查集：支持Union（合并集合），Find（寻找元素所在集合）操作的数据结构

并查集的实现：树形结构，每个集合用一棵树表示，树的结点是集合中的元素，可用数组实现，其中数组下标是元素名称，数组元素是对应下标的元素的父亲，根的数组元素是负数，绝对值代表集合中元素的数量

- 合并操作：将两个根的值相加，作为新根的元素数，再将另一个根的父亲修改为新根。
- **加权合并**：将元素数量较少的根合并到元素数量较多的根，**减少树的高度**
- 查找操作：按数组元素的值查找直到到负数为止
- **压缩路径的查找**：将查询路径上经过的所有结点的父亲都修改为根，**减少树的高度**

并查集可用来划分等价类，以及解决图连通问题

字典

字典：一些元素的集合，每个元素有唯一的 key，字典元素一般是<键-值>对。字典的基本操作包括搜索，插入和删除

散列

散列表：建立key和存储位置的映射，不经过比较关键码直接获得元素

$Address = Hash(Key)$

冲突：将不同的关键码映射到同一个散列地址，产生冲突的关键码叫做同义词

搜索效率衡量：

ASL_{succ} ：找到表中已有元素的平均比较次数

ASL_{unsucc} ：在表中找不到待查元素但找到插入位置的平均比较次数

散列函数

散列函数的定义域必须包括所有需要存储的关键码，而若散列表可用地址个数为 m，则其值域需在 0 到 m-1，且**计算出的散列值应当均匀分布在地址空间**（减少冲突发生的概率）

- **除留余数法**：取一个不大于 m 且接近 m 的质数 p 作为除数，hash(key) = key % p。要求 p 不接近 2 或 10 的幂
- 数字分析法：取关键码中某几位数字出现比较平均的作为散列值，**完全依赖于关键码集合，换关键码需要重新选择作为散列值的位数**
- 平方取中法：取 key 的平方的中间几位，一般取散列地址为 8 的某次幂，如总数 m = 8^r，则散列值取关键码平方的中间 r 位
- 折叠法：把关键码自左到右分成位数相等的几部分，其中每一部分的位数和散列表地址位数相同，叠加起来即可得到散列值，叠加方法有移位法和分界法

处理冲突

闭散列法：若冲突则寻找下一个可用的存储位置

- **线性探查法**：d = 1，2，……
- 二次探查法：d = 1，-1，4，-4，9，……（散列表大小必须是满足 4k+3 的质数）
- 双散列法：d = hash(key)

开散列法：同义词归于同一子集合，每个子集合都是一个”桶“，桶可用链表或自平衡的二叉树等实现