数据结构

## 哈希表

在记录的存储位置和他的关键字之间建立一个确定的对应关系f，使得每个关键字和结构中一个唯一的存储位置对应，f叫哈希函数，按这个思想建立的表为哈希表。

### 1.1、哈希函数构造方法

**A直接定址法**

H(key)=key或H(key)=a\*key+b

直接定址法所得地址集合和关键字集合大小相同，因此对于不同的关键字不会产生冲突，但是实际中能使用这种哈希函数的情况很少。

**B数字分析法**

**C平方取中法**

取关键字平方后的中间几位为哈希地址

**D折叠法**

将关键字分割成位数相同的几部分（最后一部分位数可以不同），然后取这几部分的叠加和（舍去进位）作为哈希地址。

**E除留余数法**

取关键字被某个不大于哈希表表长m的数p除后所得的余数作为哈希地址

H（key）=key MOD p ,p<=m (p的选择很重要)一般情况下，可以选p为质数或者不包含小于20的质因数的合数。

**F随机数法**

选择一个随机函数，取关键字的随机函数值为他的哈希地址。H(key)=random（key）

### 1.2 处理冲突的方法

**A 开放定址法**

HI=(H(key)+di) MOD m i=1，2，……，k（k<=m-1）

H(key)为哈希函数m为哈希表表长，di为增量序列，有3种取法：线性探测再散列，二次探测再散列，伪随机数序列。

线性探测再散列：只要哈希表未填满，总能找到一个不发生冲突的地址，但可能发生“二次聚集”；

**B 再哈希法**

在同义词产生地址冲突时计算另一个哈希函数地址，不易产生“聚集”，但是增加了计算时间。

**C 链地址法**

将所有关键字为同义词的记录存储在同一线性链表中。

**D 建立公共溢出区**

将发生冲突的词填入溢出表