# 第七章知识小结

* 查找表：由同一类型的数据元素（或记录）构成的集合
* 静态查找表：查找的同时对查找表不做修改操作（如插入和删除）
* 动态查找表：查找的同时对查找表具有修改操作
* 关键字：记录中某个数据项的值，可用来识别一个记录
* 主关键字：唯一标识数据元素（身份证号）
* 次关键字：可以标识若干个数据元素（姓名、籍贯、专业等）

1. 顺序查找

应用范围：顺序存储或链式存储的线性表（即任何结构的线性表）， 表内元素之间无序。

顺序查找算法的特点：

* 算法简单，对表结构无任何要求（顺序和链式）
* n很大时查找效率较低
* 改进措施：非等概率查找时，可按照查找概率进行排序。

顺序查找的性能分析

（1）时间复杂度：O(n)

（2）查找成功时的平均查找长度

设表中各记录查找概率相等

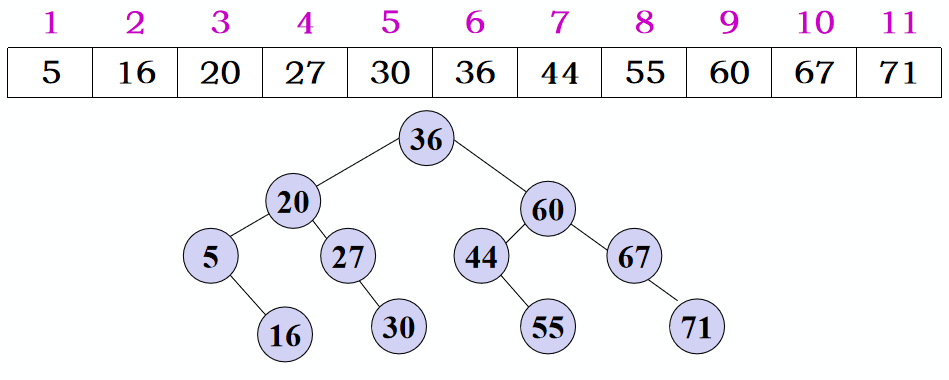
ASLs(n)=(1+2+ ... +n)/n =(n+1)/2

（3）查找不成功时的平均查找长度：

ASLf =n+1

二、折半查找（适用于有序的顺序表）

折半查找的性能分析－判定树



（1）树中每个结点对应表中一个记录。

假定每个元素的查找概率相等，则：ASL＝ (1\*1＋2×2＋4×3+4\*4 )/11＝33/11=3

（2）表中中间结点作为根。

（3）左子表作为左子树。

（4）右子表作为右子树。

折半查找的性能分析

查找过程：每次将待查记录所在区间缩小一半，比顺序查找效率高,时间复杂度O(log2 n)

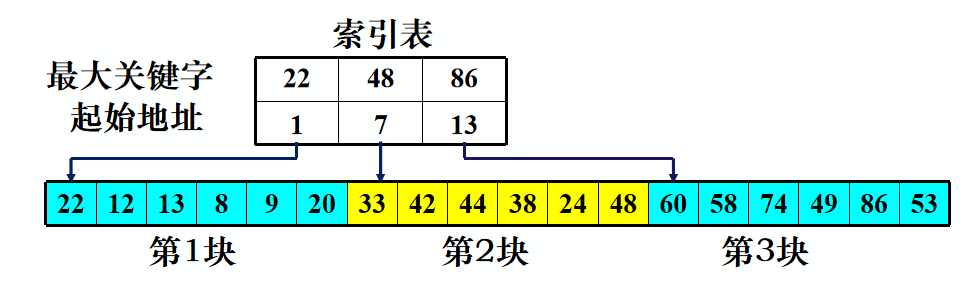
适用条件：采用顺序存储结构的有序表，不宜用于链式结构

1. 分块查找

分块查找过程

① 对索引表使用折半查找法（因为索引表是有序表）；

② 确定了待查关键字所在的子表后，在子表内采用顺序查找法（因为各子表内部是无序表）；



分块查找性能分析

查找效率：ASLbs=Lb+Lw

Lb :对索引表查找的ASL, Lw :对块内查找的ASL

其中： S为每块内部的记录个数，n/s即块的数目

例如，当n=9，s=3时，ASLbs=3.5,而折半法为3.1,顺序法为5

优点：插入和删除比较容易，无需进行大量移动。

缺点：要增加一个索引表的存储空间并对初始索引表进行排序运算。

适用情况：如果线性表既要快速查找又经常动态变化，则可采用分块查找。

四、树表的查找

线性表查找的缺点：

（1）顺序查找虽然对存储结构没有要求，但查找太慢。

（2）折半查找虽然较快， 但必须有序，且顺序存储。

（3）不方便进行插入、删除操作，属于静态查找。

若要动态查找，可采用树形结构组织数据，统称为树表。

（1）二叉排序树

（2）平衡二叉树

（3）B-树

五、二叉排序树

二叉排序树的定义：

二叉排序树或是空树，或是满足如下性质的二叉树：

(1)若其左子树非空，则左子树上所有结点的值均小于根结点的值；

(2)若其右子树非空，则右子树上所有结点的值均大于等于根结点的值；

(3)其左右子树本身又各是一棵二叉排序树。

二叉排序树的操作：查找、插入、生成、删除

1. 平衡二叉树

平衡二叉树（AVL树）定义：

* 所有结点的左、右子树深度之差的绝对值≤ 1
* 左、右子树是平衡二叉树；
* 平衡因子:该结点左子树与右子树的高度差

平衡二叉树的特点：

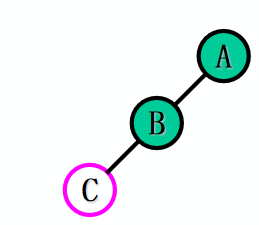
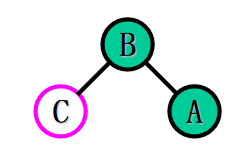
* 任一结点的平衡因子只能取：-1、0 或 1；
* 如果树中任意一个结点的平衡因子的绝对值大于1，则这棵二叉树就失去平衡，不再是AVL树；

平衡二叉树的变化：

如果在一棵AVL树中插入一个新结点，就有可能造成失衡，此时必须重新调整树的结构，使之恢复平衡。我们称调整平衡过程为平衡旋转。

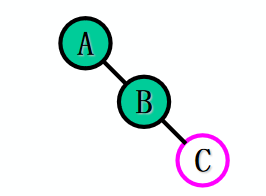
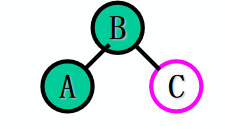
1）LL平衡旋转：

若在A的左子树的左子树上插入结点，使A的平衡因子从1增加至2，需要进行一次顺时针旋转。（以B为旋转轴）

->

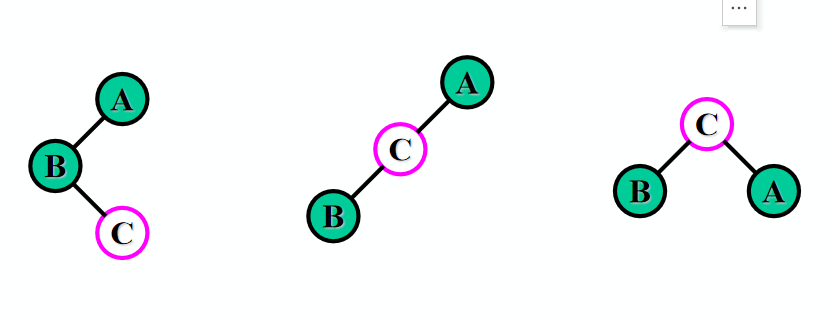
2）RR平衡旋转：

若在A的右子树的右子树上插入结点，使A的平衡因子从-1增加至-2，需要进行一次逆时针旋转。（以B为旋转轴）

->

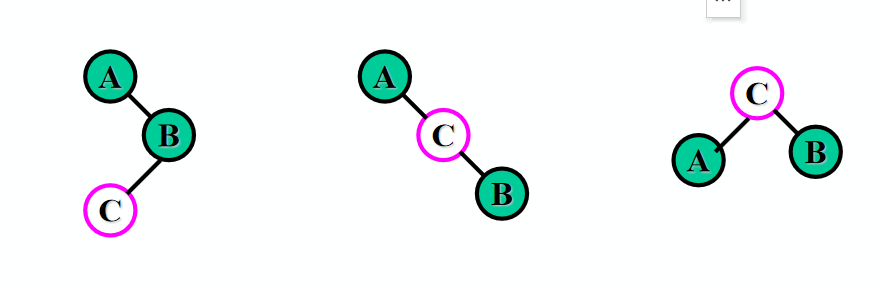
3）LR平衡旋转：

若在A的左子树的右子树上插入结点，使A的平衡因子从1增加至2，需要先进行逆时针旋转，再顺时针旋转。(以插入的结点C为旋转轴）



4）RL平衡旋转：

若在A的右子树的左子树上插入结点，使A的平衡因子从-1增加至-2，需要先进行顺时针旋转，再逆时针旋转。(以插入的结点C为旋转轴）



1. 哈希表的查找

哈希函数的构造方法

1. 直接定址法
2. 数字分析法
3. 平方取中法
4. 折叠法
5. 除留余数法（最常用重点掌握）
6. 随机数法
7. 处理问题冲突的方法

1.开放地址法

* 线性探测法

线性探测法的特点

优点：只要哈希表未被填满，保证能找到一个空地址单元存放有冲突的元素。

缺点：可能使第i个哈希地址的同义词存入第i+1个地址，这样本应存入第i+1个哈希地址的元素变成了第i+2个哈希地址的同义词，……，产生“聚集”现象，降低查找效率。

* 二次探测法
* 伪随机探测法

1. 链地址法（拉链法）

关于哈希表的几点结论：

1.对哈希表技术具有很好的平均性能，优于一些传统的技术。

2.链地址法优于开放地址法。

3.除留余数法作哈希函数优于其它类型函数。