

数据结构第十二节课官方笔记

目录

- 一、 课件下载及重播方法
- 二、 本章/教材结构图
- 三、 本章知识点及考频总结
- 四、 配套练习题
- 五、 其余课程安排

一、课件下载及重播方法

二、教材结构图



三、本章知识点及考频总结

(一) 选择题 (共 12 道)

1. 顺序查找**成功查找查找次数为: $(n+1)/2$**

不成功: $n+1$

2. 二分查找 (Binary Search)又称折半查找，是一种效率较高的查找方法。二分查找要求查找对象的线性表必须是**顺序存储结构的有序表**（不妨设递增有序）。

可用近似公式表示二分查找成功时的平均查找长度：

$$ASL = \log_2(n+1) - 1$$

3. 二分查找算法：

```
int BinSearch(SeqList R,KeyType k,int low,int high)
{   //在区间 R[low..high]内进行二分递归，查找关键字值等于 k 的记录

    //low 的初始值为 1,high 的初始值为 n

    int mid;

    if (low<=high) {

        mid= (low+high) /2;

        if (R[mid] .key==k) return mid;           //查找成功，返回其下标

        if (R[mid] .key>k)

            return BinSearch(R,k,low,mid-1);    //在左子表中继续查找

        else

            return BinSearch(R,k,mid+1, high);   //在右子表中继续查找

    }

    else

        return 0;                               //查找失败，返回 0 值

}
```

4. 索引顺序查找又称分块查找，是一种介于顺序查找和二分查找之间的查找方法。

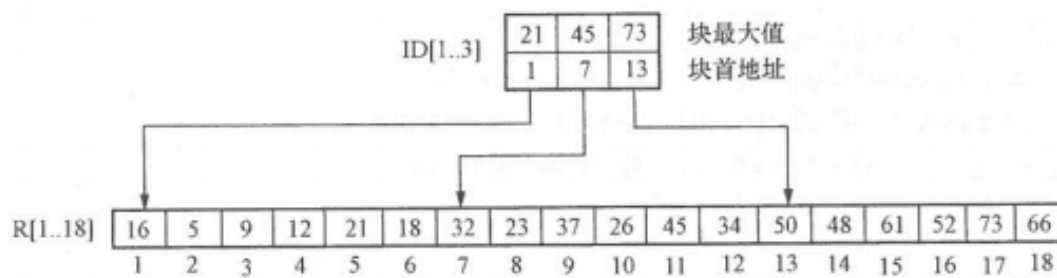


图 8.4 分块有序表及其索引表的存储表示

查找块有两种方法，一种是二分查找，若按此方法来确定块，则分块查找的平均查找长度为：

$$ASL_{blk} = ASL_{bin} + ASL_{seq} = \log(b+1) - 1 + (s+1)/2 \approx \log(n/s+1) + s/2$$

另一种是顺序查找，此时的分块查找的平均查找长度为：

$$ASL_{blk} = (b+1)/2 + (s+1)/2 = (s^2 + 2s + n)/(2s)$$

5. 二叉排序树(Binary Sort Tree, BST)又称二叉查找树，是一种特殊的二叉树，它或者是一棵空树，或者是具有下列性质的二叉树：

- (1) 若它的右子树非空，则右子树上所有结点的值均大于根结点的值。
- (2) 若它的左子树非空，则左子树上所有结点的值均小于根结点的值。
- (3) 左、右子树本身又各是一棵二叉排序树。

从上述性质可推出二叉排序树的另一个重要性质，即按**中序遍历二叉排序树**所得到的**遍历序列是一个递增有序序列**。

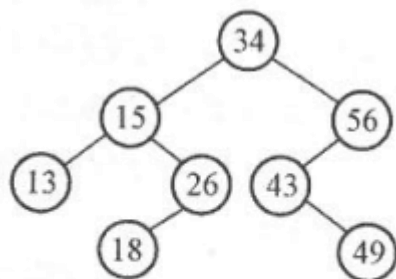


图 8.5 一棵二叉排序树

6.

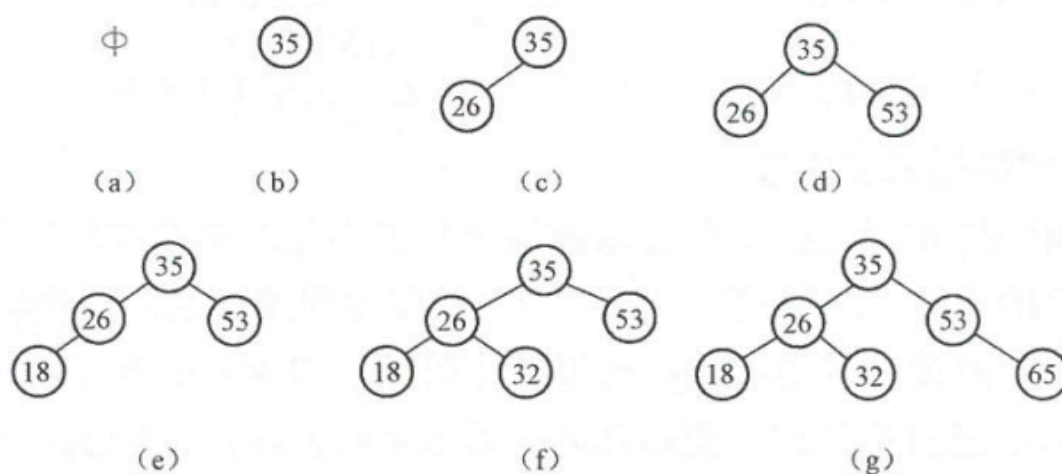


图 8.6 二叉排序树的构造过程

$$ASL = \sum_{i=1}^6 p_i c_i = (1 + 2 \times 2 + 3 \times 3) / 6 \approx 2.3$$

7. B 树的定义

一棵 m ($m \geq 3$) 阶的 B 树，或为空树，或为满足下列性质的 m 叉树：

(1) 每个结点至少包含下列信息域：

$$(n, p_0, k_1, p_1, k_2, \dots, k_n, p_n)$$

其中， n 为关键字的个数； k_i ($1 \leq i \leq n$) 为关键字，且 $k_i < k_{i+1}$ ($1 \leq i \leq n-1$)；

p_i ($1 \leq i \leq n$) 为指向子树根结点的指针，且 p_i 所指向子树中所有结点的关键字均小于 k_{i+1} ，

p_n 所指向子树中所有结点关键字均大于 k_n ；

(2) 树中每个结点至多有 m 棵子树。

(3) 若树为非空，则根结点至少有 1 个关键字，至多有 $m-1$ 个关键字。因此，若根结点不是叶子，则它至少有两棵子树。

(4) 所有的叶结点都在同一层上，并且不带信息（可以看作是外部结点或查找失败的结点，实际上这些结点不存在，指向它们的指针均为空），叶子的层数为树的高度 h 。

(5) 每个非根结点中所包含的关键字个数满足： $\lceil m/2 \rceil - 1$ 。因为每个内部结点的度数正好是关键字总数加 1，所以，除根结点之外的所有非终端结点（非叶子结点的最下层的结点称为终端结点）至少有 $\lceil m/2 \rceil$ 棵子树，至多有 m 棵子树。

8.

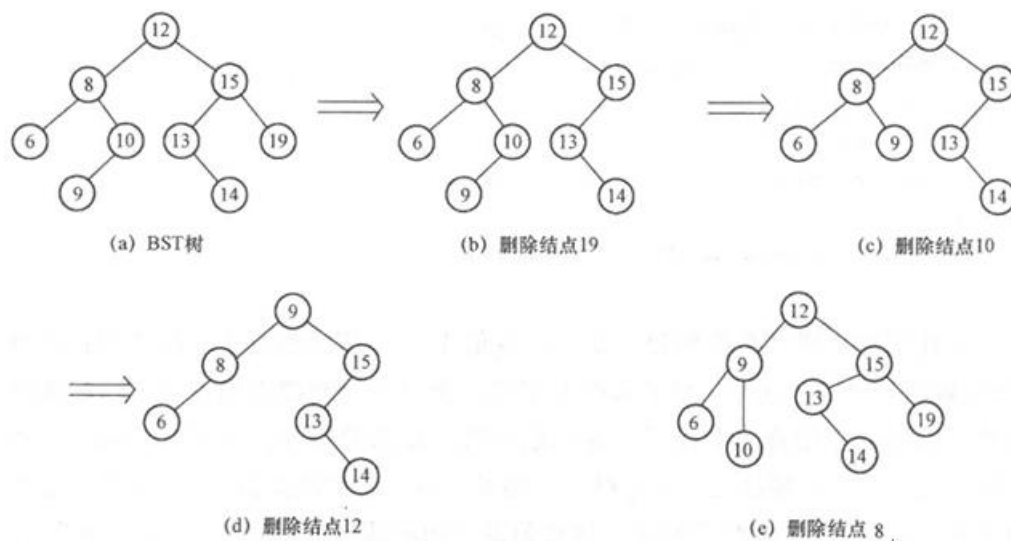


图 8.8 BST 树的结点删除情况

9. 以线性表中的每个元素的关键字 key 为自变量，通过一种函数 $H(key)$ 计算出函数值，把这个函数值解释为一块连续存储空间的单元地址（即下标），将该元素存储到这个单元中。散列存储中使用的函数 $H(key)$ 称为**散列函数或哈希函数**，它实现关键字到存储地址的映射（或称转换）。 $H(key)$ 的值称为散列地址或哈希地址，使用的数组空间是线性表进行散列存储的地址空间，所以被称之为散列表或哈希表。

10. 散列函数的构造方法：

直接地址法、数字分析法、除余数法、平方取中法、折叠法。

11. 处理冲突的方法：

开放定址法：分为**线性探测法、二次探查法和双重散列法**。

拉链法（链地址法）

12.

【例 8.7】 设散列函数 $f(k) = k \% 13$, 散列表地址空间为 $0 \sim 12$, 对给定的关键字序列 (19, 14, 01, 68, 20, 84, 27, 26, 50, 36) 分别以拉链法和线性探查法解决冲突构造散列表, 画出所构造的散列表, 指出在这两个散列表中查找每一个关键字时进行比较的次数, 并分析在等概率情况下查找成功和不成功时的平均查找长度以及当结点数为 $n=10$ 时的顺序查找和二分查找成功与不成功的情况。

分析:

(1) 用线性探测法解决冲突, 其散列表如图 8.21 所示。

下标 (地址)	0	1	2	3	4	5	6	7	8	9	10	11	12
散列表 T[0..12]	26	14	01	68	27		19	20	84		36	50	
探查次数	1	1	2	1	4		1	1	3		1	1	

图 8.21 线性探查法建立的散列表

因此, 在该表上的平均查找长度为

$$ASL = (1+1+2+1+4+1+1+3+1+1) / 10 = 1.6$$

(2) 用拉链法解决冲突, 构造的散列表如图 8.22 所示。

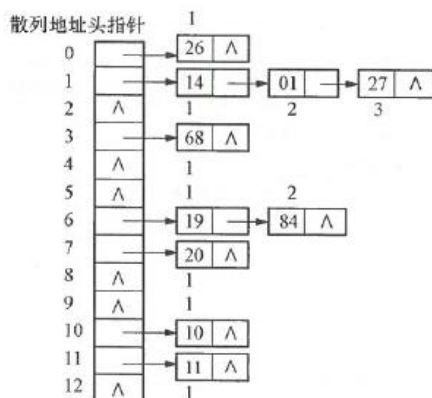


图 8.22 拉链法建立的散列表

在该散列表上的平均查找长度为

$$ASL = (1 \times 7 + 2 \times 2 + 3 \times 1) / 10 = 1.4$$

而当 $n=10$ 时, 顺序查找和二分查找的平均长度分别为

$$ASL_{seq} = (10+1) / 2 = 5.5$$

$$ASL_{bin} = (1 \times 1 + 2 \times 2 + 3 \times 4 + 4 \times 3) / 10 \approx 3$$

对于查找不成功的情况, 顺序查找和二分查找所需要进行的關鍵字仅取决于表长,

而散列表查找所需要进行的比较次数和待查结点有关。

(二) 主观题 (共 1 道)

记录下标:	1	2	3	4	5	6	7	8	9	10	11
初始关键字	[13	25	36	42	48	56	64	69	78	85	92]
第 1 次比较								↑ mid=6			
在左子区间	[13	25	36	42	48]	56	64	69	78	85	92
第 2 次比较				↑ mid=3							
在右子区间	13	25	36	[42	48]	56	64	69	78	85	92
第 3 次比较					↑ mid=4						

(a) 查找 $k=42$ 的过程 (三次比较后查找成功)

记录下标:	1	2	3	4	5	6	7	8	9	10	11
初始关键字	[13	25	36	42	48	56	64	69	78	85	92]
第 1 次比较								↑ mid=6			
在右子区间	13	25	36	42	48	56	[64	69	78	85	92]
第 2 次比较									↑ mid=9		
在右子区间	13	25	36	42	48	56	64	69	78	[85	92]
第 3 次比较											↑ mid=10
	13	25	36	42	48	56	64	69	78	85	92
								high ↑		↑ low	

(b) 查找 $k=80$ 的过程 (三次比较后查找失败)

图 8.1 二分查找查找成功和查找失败的查找过程示意图

四、配套练习题

1、对有序表(1,9,12,41,62,77,82,95,100)采用二分查找方法查找值 82，查找过程中关键字的比较次数是 ()。

A:1

B:2

C:4

D:7

2、查找较快，且插入和删除操作也比较方便的查找方法是 ()。

A:分块查找

B:二分查找

C:顺序查找

D:折半查找

3、下列叙述中，不符合 m 阶 B 树定义的是 ()。

A:根结点可以只有一个关键字

B:所有叶结点都必须在同一层上

C:每个结点内最多有 m 棵子树

D:每个结点内最多有 m 个关键字

[参考答案]: BAD

五、其余课程安排