

Lesson 5&6 Practice: (树、集合和散列)

221900180 田永铭

1. 如果一棵树有 n_1 个度为 1 的结点, 有 n_2 个度为 2 的结点, \dots , n_m 个度为 m 的结点, 试问有多少个度为 0 的结点? 试推导之。

答: $n_1 + 2 \cdot n_2 + 3 \cdot n_3 + \dots + m \cdot n_m = \text{ans} + n_1 + n_2 + \dots + n_m - 1$ (树中点比边多一),
所以 $\text{ans} = (n_1 + 2 \cdot n_2 + 3 \cdot n_3 + \dots + m \cdot n_m) - (n_1 + n_2 + \dots + n_m) + 1$
 $= n_2 + 2 \cdot n_3 + \dots + (m-1) \cdot n_m + 1$

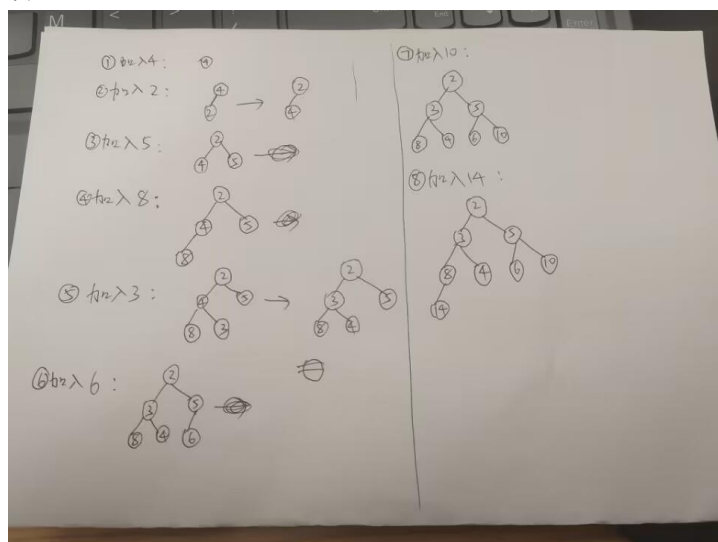
2. 请描述满足以下条件的所有二叉树应有的形态:

- (1) 二叉树的前序序列与中序序列相同;
- (2) 二叉树的中序序列与后序序列相同;
- (3) 二叉树的前序序列与后序序列相同。

答: (1)空树 或者 所有节点没有左子树的树;
(2)空树 或者 所有节点没有右子树的树;
(3)空树 或者 只有一个节点 (根节点) 的树;

3. 写出向空堆 (最小堆) 中加入数据 4, 2, 5, 8, 3, 6, 10, 14 时, 每加入一个数据后堆的变化。

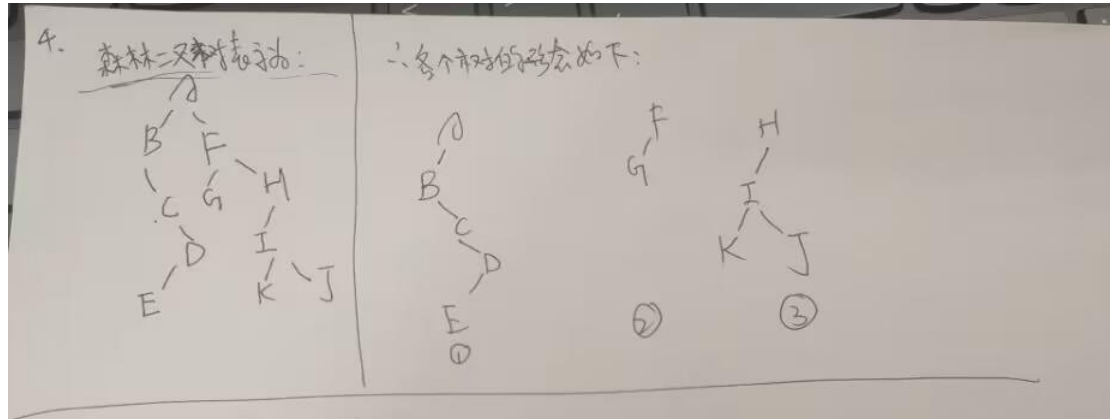
答:



4. 已知下表是森林的“左子女-右兄弟”表示的静态二叉链表的存储表。

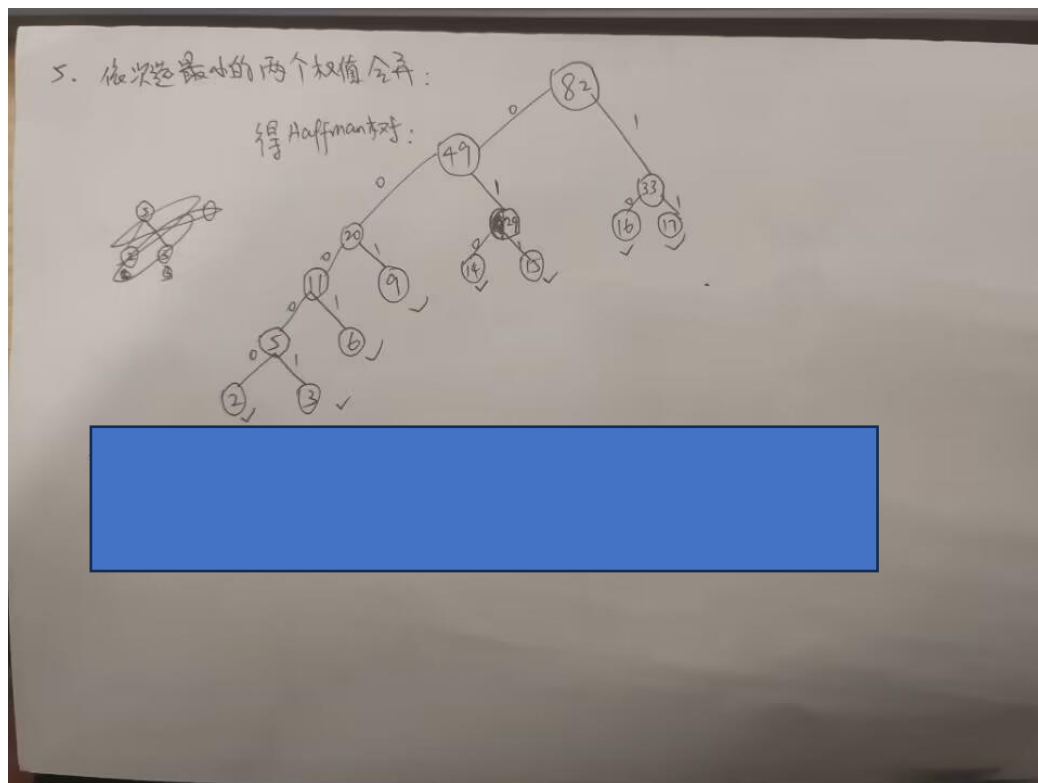
	0	1	2	3	4	5	6	7	8	9	10
llink	B	-1	-1	E	-1	G	-1	I	K	-1	-1
data	A	B	C	D	E	F	G	H	I	K	J
rlink	F	C	D	-1	-1	H	-1	-1	J	-1	-1

请将该森林对应的各个树的形态画出。



5. 给定权值集合{15, 3, 14, 2, 6, 9, 16, 17}, 构造相应的霍夫曼树, 并计算它的带权外部路径长度。

2, 3, 6, 9, 14, 15, 16, 17

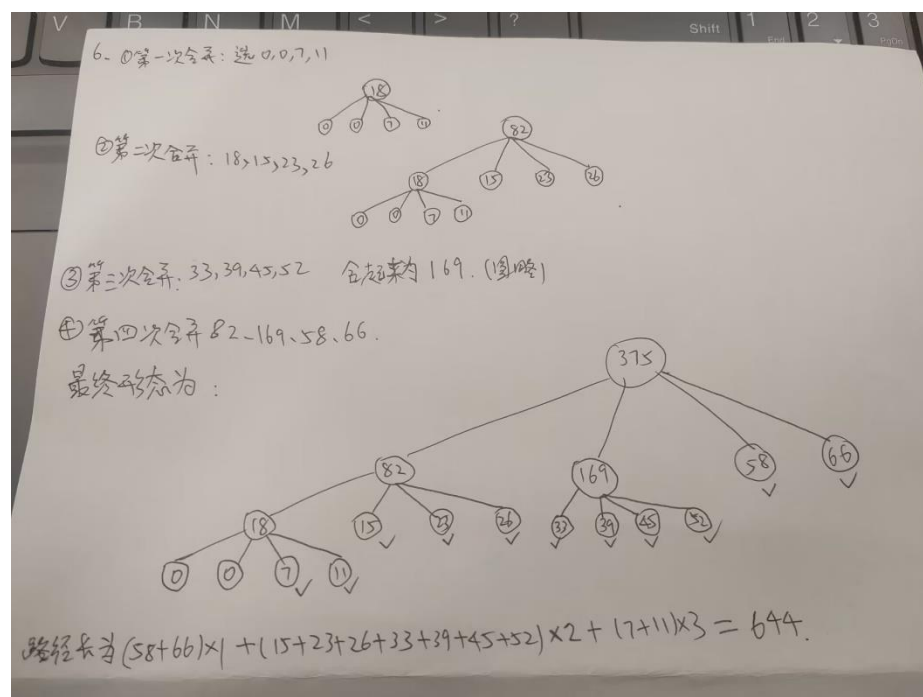


路径长为 $(16+17)*2+(9+14+15)*3+6*4+(2+3)*5 = 229$

6. 给定一组权值: 23, 15, 66, 07, 11, 45, 33, 52, 39, 26, 58, 试构造一棵具有最小带权外部路径长度的扩充4叉树(只有度为4的内结点和度为0的外结点), 要求该4叉树中所有内部结点的度都是4, 所有外部结点的度都是0, 如果外结点 $n_0 \neq 3 * n_4 + 1$, 可按需补充外结点数量(补充的外结点权值为0)。这棵扩充4叉树的带权外部路径长度是多少?

【解答】权值个数 $n = 11$, 扩充4叉树的内部结点的度都为4, 而外结点的度都为0。设内结点个数为 n_4 , 外结点个数为 n_0 , 则可证明有关系 $n_0 = 3 * n_4 + 1$ 。由于在本题中 $n_0 = 11 \neq 3 * n_4 + 1$, 需要补2个权值为0的外结点。此时内结点个数 $n_4 = 4$ 。仿照霍夫曼树的构造方法来构造扩充4叉树, 每次合并4个结点。

答:



7. 已知待散列的线性表为 (36, 15, 40, 63, 22), 散列用的一维地址空间为 [0..7], 假定选用的散列函数是 $H(K) = K \bmod 7$, 若发生冲突采用线性探查法处理。

(1) 计算出每一个元素的散列地址并在下图中填写出散列表:

0	1	2	3	4	5	6	7

(2) 求出在查找每一个元素概率相等情况下的平均查找长度。

(3) 求出查找失败的平均查找长度。

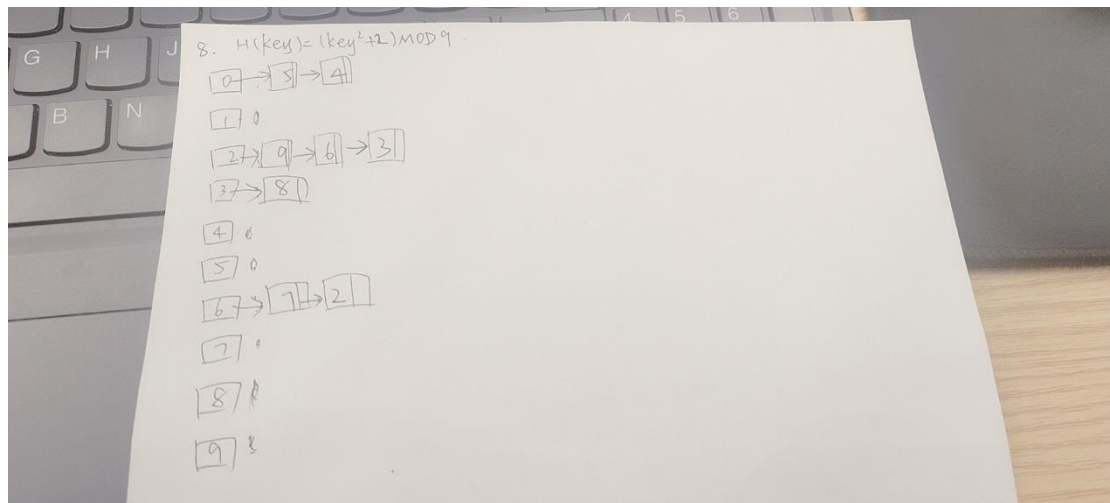
答: (1)

0	1	2	3	4	5	6	7
63	36	15	22		40		

(2) $AL = (1+1+2+3+1) / 5 = \frac{8}{5}$

(3) $AL' = (5+4+3+2+1+2+1)/7 = \frac{18}{7}$

8. 设散列表的地址范围是[0..9]，散列函数为 $H(\text{key}) = (\text{key}^2 + 2) \text{MOD } 9$ ，并采用链表处理冲突，请画出元素 7、5、4、9、6、2、8、3 依次插入散列表的存储结构。



9. 下列算法实现在顺序散列表 `hashtable[0..m-1]` 中查找值为 `k` 的关键字，请在下划线处填上正确的语句。

```
typedef struct record {int key; int delflag; T others;};
int hashsearch(record hashtable[ ],int k)
{
    int i,j;   j=i=k % p;
    while (hashtable[j].key!=k&&hashtable[j].delflag!=0)
    {
        j=(j+1) %m;
        if (i==j) return(-1);
    }
    if (hashtable[j].key==k) return(j);
    else return(-1);
}
```

10. 设散列函数 $H(k)=k \bmod p$ ，解决冲突的方法为链地址法。要求在下列算法划线处填上正确的语句完成将数组 `a[0..n-1]` 中的关键字装入散列表 `hashtable[0..m-1]` 中。

```
typedef struct node {int key; struct node *next;} lklist;
void createlkhash(lklist *hashtable, int a[ ])
{
    int i, k;   lklist *s;
    for(i=0;i<m;i++) hashtable[i] = nullptr;
    for(i=0;i<n;i++)
    {
        s=new lklist; s->key=a[i];
        k= a[i]%p; s->next=hashtable[k]; hashtable[k] = s;
    }
}
```