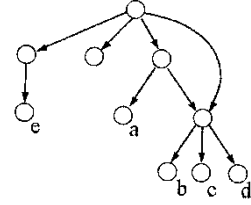
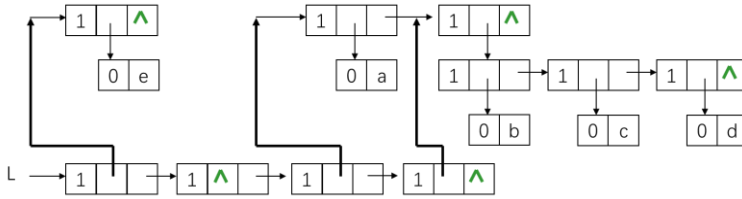


参考答案：

一、(1) ((e), (), (a, (b, c, d)), (b, c, d)) (2分) 见下图 (3分)

(2) 深度=3 (2分)



二、哈希表是一种根据关键字转换为地址的查找表。(2分)

(1) 地址映射 (哈希函数) H 冲突 (同义词) 解决策略 (4分)

(2) 优点：地址映射，查找速度快 (2分)

三、

(1) 通过一趟排序将要排序的数据分割成独立的两部分，其中一部分的所有数据都比另外一部分的所有数据都要小，然后再按此方法对这两部分数据分别进行快速排序，整个排序过程可以递归进行，以此达到整个数据变成有序序列。(5分)

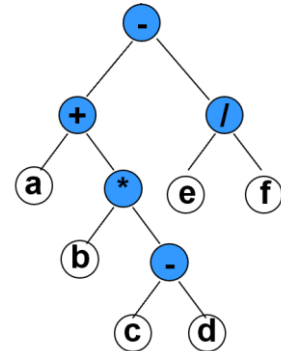
(2) 冒泡排序是两两比较的算法，时间复杂度为  $O(n^2)$ ；而快速排序改进了冒泡排序，采用分治策略，每一轮比较将排序序列一分为二，使得快排的时间复杂度在平均意义上为  $O(n \log n)$ ；(3分)

(3) 快速排序可能存在的问题是枢轴如果选择不合适，可能退化为冒泡排序。(2分)

四、

(1) 后缀：abcd-\*+ef/- (3分) 前缀：-+a\*b-cd/ef (3分)

(2) 右图。(4分)



五、

(1) 堆：顺序存储的二叉树 (2分)

(2) 最小值在根节点 (1分)，最大值在叶子中 (2分)

(3) 9次比较 (n-1)次 (1分)，最小堆见右图。(4分)

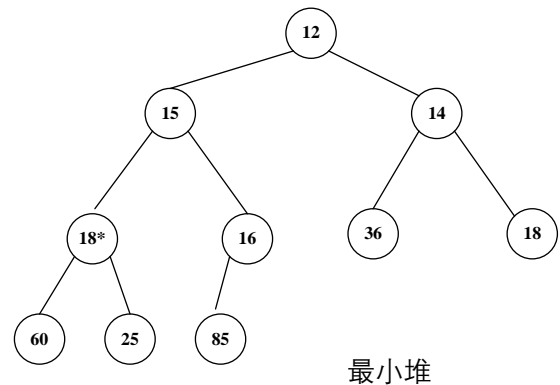
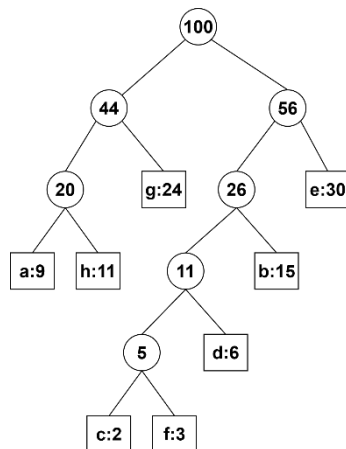
六、哈夫曼树

(6分)

不严格限制左右

h:11 的位置也可不同

树的深度唯一。



最小堆

a: 000, b: 101, c: 10000, d: 1001, e: 11, f:10001, g: 01, h: 001 (4分)

01 序列可不同，但码长不对则一定错误。

## 七、迪杰斯特拉

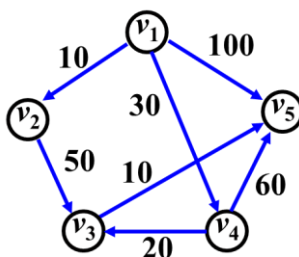
过程 (4 分), 数值结果 (4 分) 路径 (2 分)

$v_1 \rightarrow v_2, 10$

$v_1 \rightarrow v_4, 30$

$v_1 \rightarrow v_4 \rightarrow v_3, 50$

$v_1 \rightarrow v_4 \rightarrow v_3 \rightarrow v_5, 60$

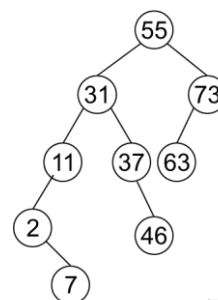


## 八、

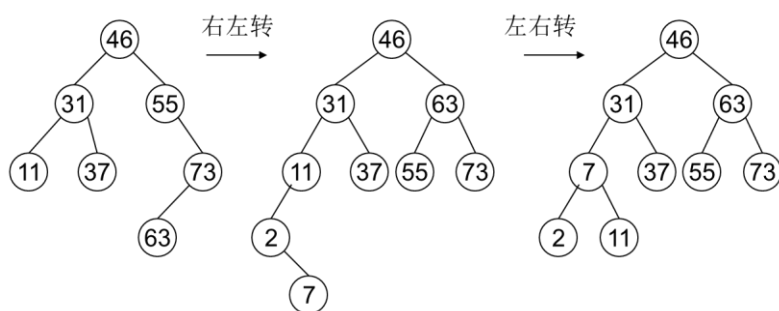
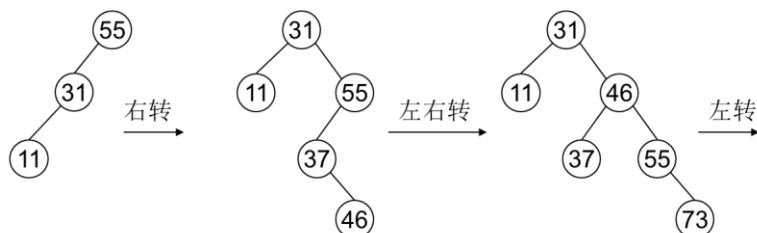
(1) 右图。(3 分)  $1+2*2+3*3+2*4+1*5=27/9$  (2 分)

(2) 下图。(4 种旋转, 共 8 分)

(3)  $1+2*2+4*3+2*4=25/9$  (2 分)



二叉搜索树



AVL 树

## 九、一个简单的动态规划问题,

没有伪代码 (程序) 或者算法不正确, 最多得 5 分

递归算法 (5)

假设某个节点  $node$ , 到它的最低叶节点的长度为  $len(node)$ , 所求的最长路径必然经过一个最高节点  $high\_node$ , 则它到最低叶节点的长度为

$len(high\_node) = \max(len(high\_node \rightarrow left), len(high\_node \rightarrow right)) + 1$ ,

路径长度为

$sum = len(high\_node \rightarrow left) + len(high\_node \rightarrow right) + 2$ ; (5 分)

## 十、链表合并 没有伪代码 (程序) 或者算法不正确, 最多得 5 分

循环直到两个链表中的一个被遍历完毕, 执行操作: 在 A 表头之后插入表 A 和表 B 中当前节点中小的元素, 如果相同, 删除表 B 中的相同元。 (6 分)

循环处理 A (B 被遍历完毕) 或者 B (A 被遍历完毕), 将 A 或 B 的剩余节点插入 A 表头之后 (2 分)

$O(A.length + B.length)$  (2 分)