98 中央大學 資料結構與演算法

1.

(a)

0	1	2	3	4	5	6	7	8	9
			3	23	35	6	15	27	95

(b)

0	1	2	3	4	5	6	7	8	9
		23	3	15	35	6	27		95

(c)

插入”23”時無法插入

(d) 可以搭配 linear hashing 實作

2.

(a) 15 次(第一次呼叫不算 recursion call)

(b) 7 次

(c) 8+15+8 = 31 次(input + temp + output)

(d) 11, 1, 12, 13, 3, 14, 15, 9

3.

(a)

A = 01, B = 0000, C = 001, D = 0001, E = 1

(b)(任意舉兩個)

111110, 000000

4.

(a) 2, 3, 11, 22, 15, 77, 80, 95, 90, 84, 50

(b) YES, 因為

i. root 之 left child 比 root 小, root 之 right child 比 root 大

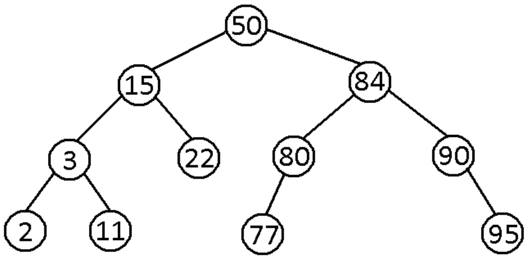
ii. 其 left child & right child 也都 recursive 符合此條件

(c)

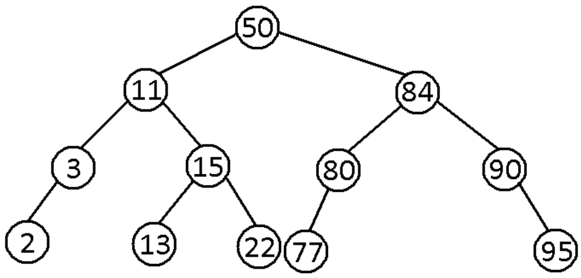
(1) remove node 2

(2) 因為去掉該 node，使得整樹的各子樹高度差<= 1

(d)



(e)



5.

```
for ( i=1 to n )
    d[i, i] = x_i;

for ( len=2 to n )
{
    for ( i=1 to n-len+1 )
    {
        j = i + len -1;
        d[i, j] = d[i, i] + d[i+1, j];
    }
}
```

6.

(a)

利用 AVL Tree 來作 sort，方法如下
將 number 依序插入 AVL Tree
若遇到相同的值，使在該值之 node 利用 count 紀錄個數，
最後再用 inorder 之 traversal print key 值

Time Complexity 分析：

因為相異的數值個數為 logn 個

所以此 AVL Tree 之高度≦$\lceil \log(\log n + 1) \rceil$ ≦ $\lceil \log\log n \rceil$

故插入每個 node 的 Time complexity 為 O(loglogn)

所以總共排序 n 個 number, 其 time complexity = O(nloglogn)

(b)

其 lower bound 的依據是排序 n 個相異的數字，故此題不在此限制範圍

7.

```
T <- empty tree;
E' <- SORT(E);
while ( E' ≠ Ø )
{
    e <- MIN{ E' };
    if( e 加入 T 中不會形成 cycle )
        ADD( T, e );
}
```

8.

Step1. 用 L : 所有點 X 座標的中位數，將平面分成 S_L, S_R

Step2. 遞迴地算出 S_L, S_R 之 maximal points, until 平面上只有一個點

Step3. 在 S_L 中的點的 Y 座標≦在 S_R 中的點的 Y 座標，則其非 maximal point，所以將之去除，

則剩下來的未被去除的點即為 maximal points

Time Complexity: $\theta(n) + 2T\left(\frac{n}{2}\right) + \theta(n) = \theta(n\log n)$