

# 《資料結構》

一、將整數資料80, 40, 19, 120, 94, 110, 115, 90, 88, 92, 98依序存入一棵空的二元搜尋樹 (binary search tree)。

(一)請畫出完成資料輸入的二元搜尋樹。(6分)

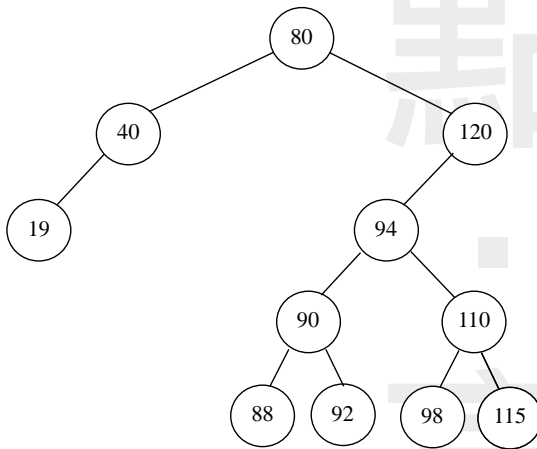
(二)從(一)產生的二元搜尋樹中刪除 (delete) 資料94, 請畫出完成刪除動作後的二元搜尋樹。(給出一個正確樹即可) (6分)

(三)請寫出自二元搜尋樹找到最大值資料所在節點 (node) 的演算法。(10分)

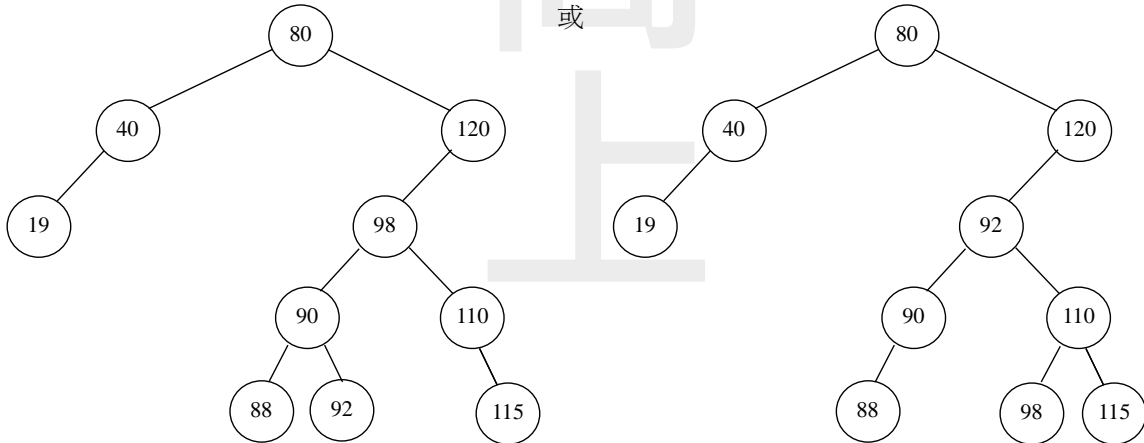
<b>試題評析</b>	本題測驗二元搜尋樹的基本操作，第一小題為插入操作，第二小題為刪除操作，第三小題則測驗考生是否能設法找出最大的一項資料。本題前兩小題應可容易拿到分數；第三小題，則需由二元搜尋樹的定義上思考最大的資料會出現在樹的何處下手。
<b>考點命中</b>	高點出版《資料結構》，王致強編著，頁6-40~6-45。

答：

(一)



(二)



【版權所有，重製必究！】

```

(三) NodePtr max(NodePtr t)
{
    if (t==NULL) return NULL;
    else
    {
        p=t;
        while(p->right !=NULL)
            p=p->right;
        return p;
    }
}

```

二、請寫出執行下列程式碼的時間複雜度，並敘明理由。(10分)

```

for (i=1; i<n; i++){
    a=1;
    b=n;
    while(a<b){
        a = 3 * a;
        b = b / 3;
    }
}

```

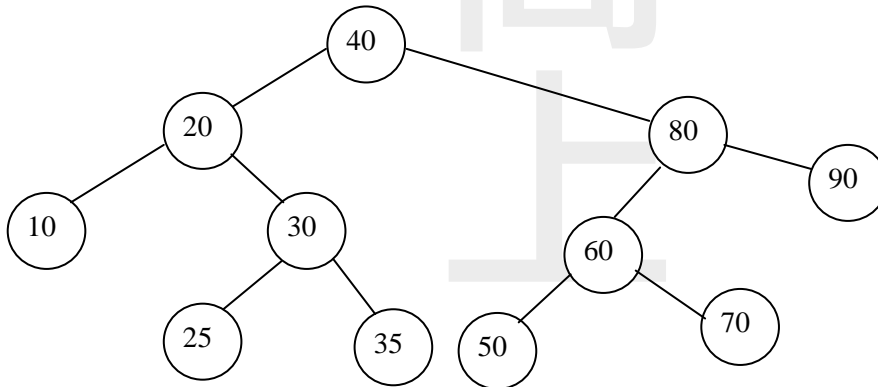
<b>試題評析</b>	本題為時間複雜度的分析，須分析兩層迴圈的次數。
<b>考點命中</b>	高點出版《資料結構》，王致強編著，頁 1-29 精選範例16。

**答：**

先計算內層迴圈的次數，當 while 迴圈執行  $k$  次之後， $a=3^k$ ， $b=n/3^k$ ，而且當 while 迴圈結束時， $a \geq b$ ，意即  $3^k \geq n/3^k$ ， $3^{2k} \geq n$ ，故  $k \geq \log_3 n$ 。

而外層的 for 迴圈，會執行  $n$  次，故整體的複雜度為  $n \times \theta(\log n) = \theta(n \log n)$ 。

三、下圖為一AVL樹T，請依各小題要求加入指定新資料後，畫出新產生的AVL樹。每小題各自獨立，都是對原先的AVL樹T，加入資料。



(一)加入資料27。(6分)

(二)加入資料45。(6分)

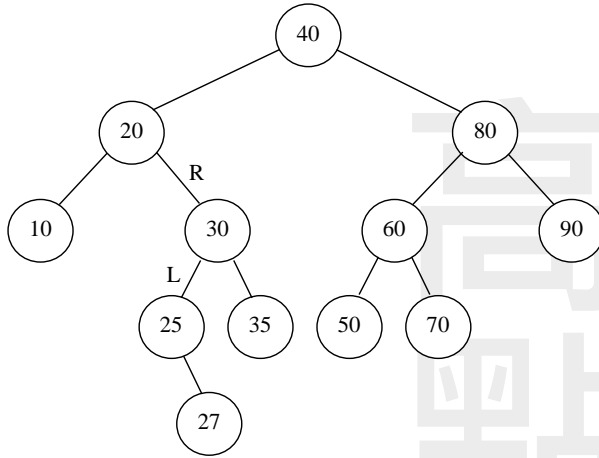
(三)加入資料95。(6分)

【版權所有，重製必究！】

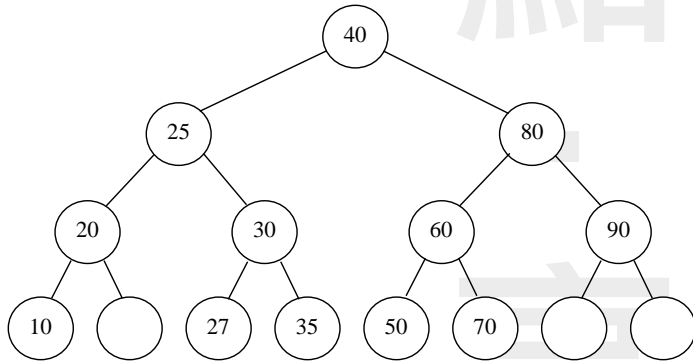
<b>試題評析</b>	本題測驗AVL-tree插入的旋轉操作，考生若有準備，應可輕鬆拿分。
<b>考點命中</b>	高點出版《資料結構》，王致強編著，頁11-26~11-29 精選範例11。

**答：**

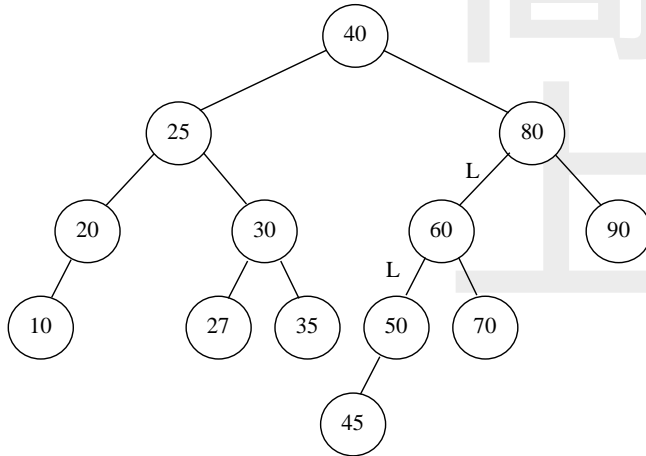
(一)插入27之後



經過 RL 雙旋轉之後為

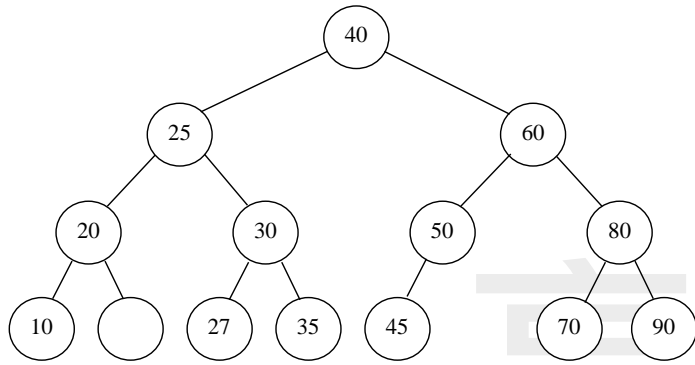


(二)再插入 45 之後

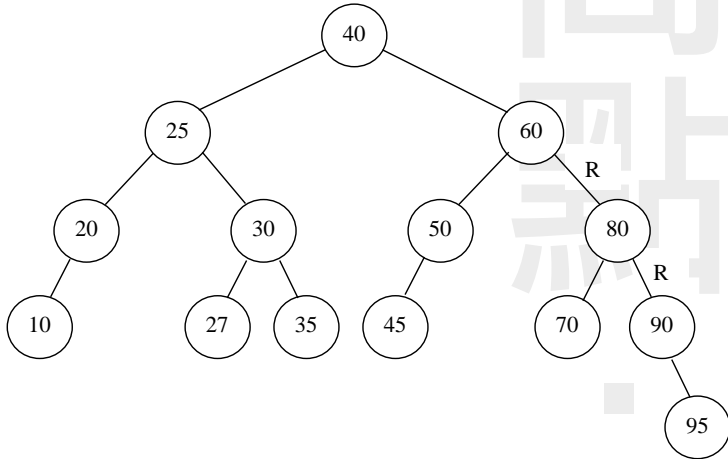


進行 LL 旋轉之後為

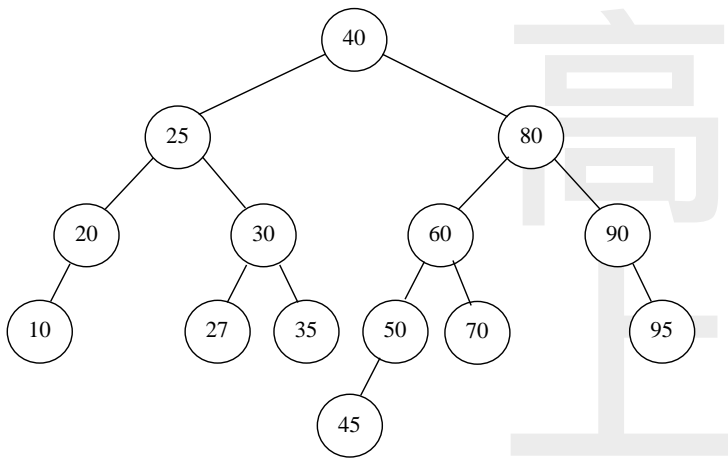
【版權所有，重製必究！】



(三)再插入 95 之後



進行 RR 旋轉，最後為



【版權所有，重製必究！】

四、函數 $f(n)$ 定義如下，其中 $n$ 為非負整數。

$$f(n) = \begin{cases} 0, & \text{若 } n=0 \\ 1, & \text{若 } n=1 \\ f(n-1) + f(n-2), & \text{若 } n > 1 \end{cases}$$

(一)請設計遞迴演算法，輸入非負整數 $n$ ，輸出 $f(n)$ 數值。(7分)

(二)請設計非遞迴演算法，輸入非負整數 $n$ ，輸出 $f(n)$ 數值。(7分)

(三)請分別說明(一)與(二)所設計演算法的時間複雜度(time complexity)。(10分)

試題評析	本題為Fibonacci數列的計算，包括遞迴與非遞兩種作法，並分析複雜度。屬於基本考題，取分不難。
考點命中	高點出版《資料結構》，王致強編著，頁5-9~5-11。

答：

(一) int f(int n)

```
{
    if (n<=1) return n;
    else return f(n-1)+f(n-2);
}
```

(二) int f(int n)

```
{
    int a[n+1];
    int i;
    a[0]=0; a[1]=1;
    for (i=2; i<=n; i++)
        a[i]=a[i-1]+a[i-2];
}
```

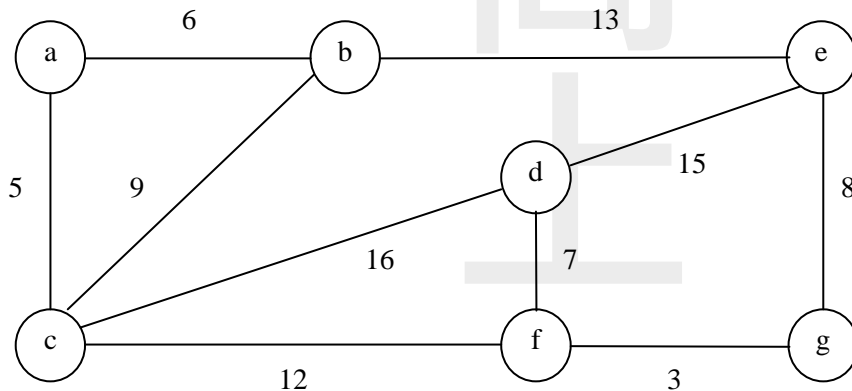
(三)(一)的時間複雜度

$$T(n) = T(n-1) + T(n-2)$$

可以推算出  $T(n) = \theta(\phi^n)$  或  $O(2^n)$ 。

(二) 主要使用迴圈，進行  $n$  次，故時間複雜度  $\theta(n)$ 。

五、(一)依據下圖內容，請寫出它的相鄰矩陣(adjacency matrix)表示法。(4分)



(二)請定義生成樹(spanning tree)。(6分)

(三)請畫出此圖的最小成本生成樹(minimum cost spanning tree)，以及計算最小成本。(10分)

【版權所有，重製必究！】

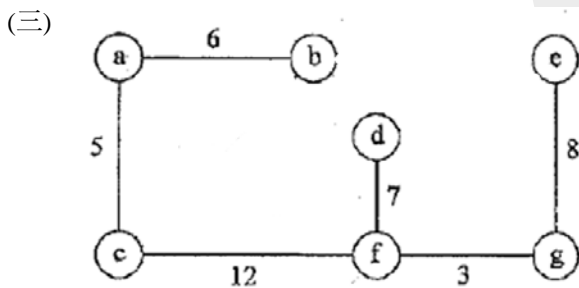
<b>試題評析</b>	本題測驗圖形表示法、生成樹與最低成本生成樹。亦屬於較本的題目，取得分數應不難。
<b>考點命中</b>	高點出版《資料結構》，王致強編著，頁7-10~7-11、8-39~8-41。

答：

(一)

	a	b	c	d	e	f	g
a	0	6	5	$\infty$	$\infty$	$\infty$	$\infty$
b	6	0	9	$\infty$	13	$\infty$	$\infty$
c	5	9	0	16	$\infty$	12	$\infty$
d	$\infty$	$\infty$	16	0	15	7	$\infty$
e	$\infty$	13	$\infty$	15	0	$\infty$	8
f	$\infty$	$\infty$	12	7	$\infty$	0	3
g	$\infty$	$\infty$	$\infty$	$\infty$	8	3	0

(二) 一個圖形的生成樹，包含原圖全部頂點(n個)，與  $n-1$  個邊，所形成的非循環連通子圖(acyclic connected subgraph)。



六、有一雜湊表格 (hash table) T 的記憶空間共含 11 個桶 (buckets)，位址編號由 0 至 10，每個桶有一個槽 (slot)。雜湊函數  $h_1$  定義為  $h_1(\text{key}) = \text{key} \% 11$ ，當有碰撞 (collision) 發生時採二次雜湊開放定址法 (open addressing with double hashing) 處理，其函數定義為  $h(\text{key}, j) = (h_1(\text{key}) + j * h_2(\text{key})) \% 11$ ，其中  $j$  為碰撞次數， $j = 1, 2, 3, \dots, 11$ ， $h_2(\text{key}) = 1 + (\text{key} \% 10)$ 。欲將 26 放入雜湊表格 T，總共經過 6 次探測才成功找到存放位址。請問 26 在雜湊表格 T 的探測順序為何？(6 分)

<b>試題評析</b>	本題所考的是雜湊法中的雙重雜湊(double hashing)，只要了解兩個函數的作用，亦可輕鬆拿到分數。
<b>考點命中</b>	高點出版《資料結構》，王致強編著，頁10-24~10-25 精選範例19。

答：

$$h_1(26) = 26 \% 11 = 4$$

$$h_2(26) = 1 + (26 \% 10) = 7$$

故第一個探索的桶子是第 4 個桶子，碰撞時增量為 7。

探索的桶子依序為：4、0、7、3、10、6。

【版權所有，重製必究！】