

## Homework 1 報告

### 錯誤一

```
68     if (t == null)
69         t = new AVLNode(x);
70     else if (x < t.data)
71     {
72         t.left = insert( x, t.left );
73         if( height( t.left ) - height( t.right ) == 2 )
74             if( x < t.left.data )
75                 t = rotateWithLeftChild( t );    // origin t = rotateWithLeftChild( t.left );
76             else
77                 t = doubleWithLeftChild( t );
78     }
79     else if( x > t.data )
80     {
81         t.right = insert( x, t.right );
82         if( height( t.right ) - height( t.left ) == 2 )
83             if( x > t.right.data )
84                 t = rotateWithRightChild( t ); //origin t = rotateWithRightChild( t.right );
85             else
86                 t = doubleWithRightChild( t );
87     }
```

原本的程式碼如同註解標示。正確的程式碼沒有標註解。

在第 75 行的地方，原本的程式碼是 `t = rotateWithLeftChild( t.left );`;

但是應該是 `t=rotateWithLeftChild(t);`

因為我們 avl tree 是要自己跟自己的左子樹交換，因此要傳入的是 `t`

同樣的道理，下方第 84 行的地方 `rotateWithRightChild` 也有同樣的錯誤，應該也是要傳入 `t` 才會對

### 錯誤二

```
private AVLNode rotateWithLeftChild(AVLNode k2)
{
    AVLNode k1 = k2.left;
    //k2.right = k1.left;
    //k1.left=k2;
    k2.left = k1.right;
    k1.right = k2;
    k2.height = max( height( k2.left ), height( k2.right ) ) + 1;
    k1.height = max( height( k1.left ), k2.height ) + 1;
    return k1;
}

/* Rotate binary tree node with right child */
private AVLNode rotateWithRightChild(AVLNode k1)
{
    AVLNode k2 = k1.right;
    k1.right = k2.left;
    //k1.left = k2.right;
    //k2.right = k1;
    k2.left=k1;
    k1.height = max( height( k1.left ), height( k1.right ) ) + 1;
    k2.height = max( height( k2.right ), k1.height ) + 1;
    return k2;
}
```

原本的程式碼如同註解標示。正確的程式碼沒有標註解。

如果要 `root` 跟自己的左子樹交換，必須先把 `root` 的左子樹改成 `root` 左子樹的右子樹，接著再把 `root` 接到左子樹的右子樹完成翻轉。

同樣的道理，如果 `root` 要跟自己的右子樹交換，必須先把 `root` 的右子樹的左子樹接到 `root` 的右子樹，再把 `root` 接到 `root` 的右子樹的左子樹完成翻轉。

因此原本的程式撰寫有錯。