

# 進階程式設計

## 物件導向實作二元搜尋樹

### 1.前言：

在這個學期的進階程式設計課中我不只練習了用python寫APCS的實作題目，老師也有教我們用以物件導向來解決二元搜尋樹的結構，這次的學習歷程檔案是要介紹二元搜尋樹。

### 2.樹的主要結構：

樹主要是用來儲存根節點，當節點需要往下延伸增加資料的時候就利用判別的方式來增加（如圖1）。函式add是用來判別數字大小跟往下增加節點的（如圖1、2）。

```
1  class Node():
2      def __init__(self,v):
3          self.value = v
4          self.left = None
5          self.right = None
6
7  def add(n):
8      temp = root
9      while(temp!=None):
10         if(temp.value>n):
11             if(temp.left==None):
12                 temp.left = Node(n)
13                 return
14             temp = temp.left
15         if(temp.value<n):
16             if(temp.right==None):
17                 temp.right = Node(n)
18                 return
19             temp = temp.right
```

```
49  value_list = [3,7,2,4,6,8]
50
51  for i in value_list:
52      add(i)
```

圖 1

### 3.搜尋、排序：

完成上述物件設計後，我利用迴圈讓樹搜尋、輸出或是排序我們所需要的資料(如圖2)。

```
20  ✓ def BinarySearch(n):
21      temp = root
22      while(temp!=None):
23          if(temp.value>n):
24              return n
25          elif(temp.value>n):
26              temp = temp.left
27          elif(temp.value<n):
28              temp = temp.right
29      return -1

30  ✓ def PreOrder(n):
31      if n!=None:
32          print(n.value)
33          PreOrder(n.left)
34          PreOrder(n.right)
35  ✓ def InOrder(n):
36      if n!=None:
37          InOrder(n.left)
38          print(n.value)
39          InOrder(n.right)
40  ✓ def PostOrder(n):
41      if n!=None:
42          PostOrder(n.left)
43          PostOrder(n.right)
44          print(n.value)
45      global root
46      root = Node(5)
47      print(root.value)
48
49      value_list = [3,7,2,4,6,8]
50
51      for i in value_list:
52          add(i)
53      PostOrder(root)
```

圖2

### 4.心得：

這次的作業讓我發現更多迴圈的用法，更發現了類別(Class)的便利性，也瞭解了前序(PreOrder)、中序(InOrder)、後序(PostOrder)的差別，學到了許多有用的技巧讓我在以後寫程式的路上更加得心應手。