



國立中山大學資訊管理所

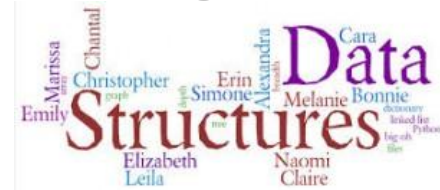
資料結構 上機實習

助教：林妤姍、謝博丞、陳亞琦
指導教授：李偉柏 博士

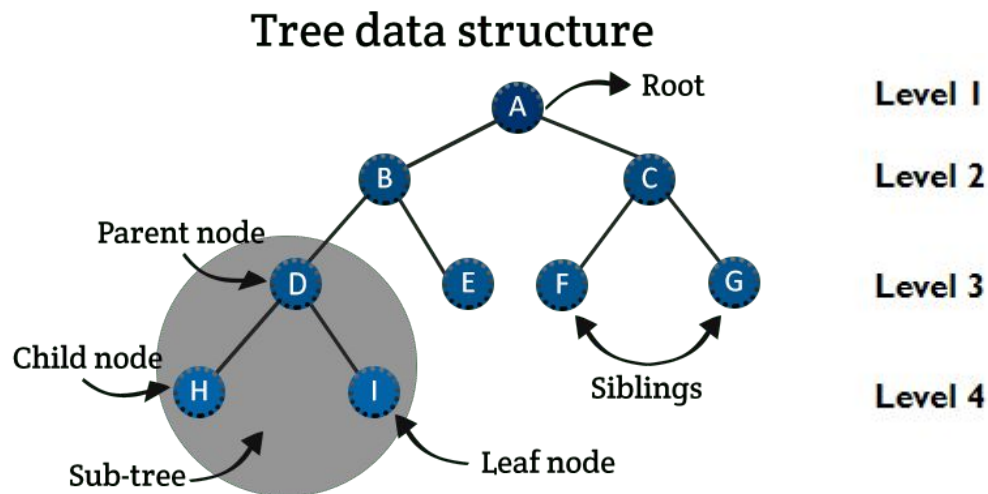
Tree



樹

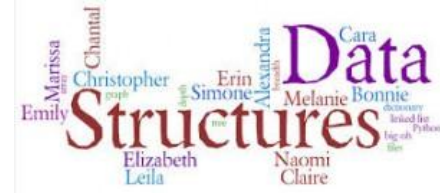


- 階層性
- 由一個或數個節點組成的有限集合,不可為空
- 特定節點, 樹根(root)
- 一個節點的子樹數稱為這個節點的分支度(degree)
- 分支度為零的節點則是樹葉(leaf)、終端(terminal)節點



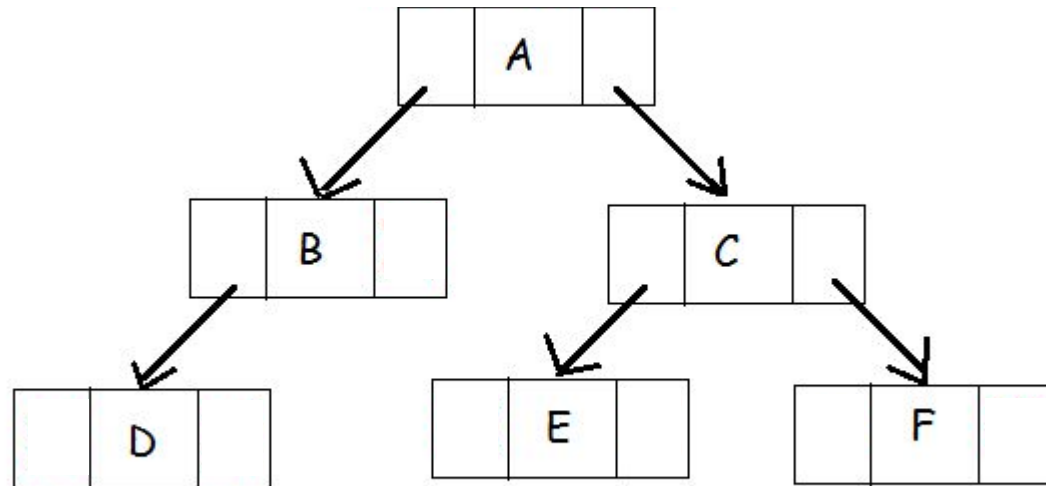


二元樹



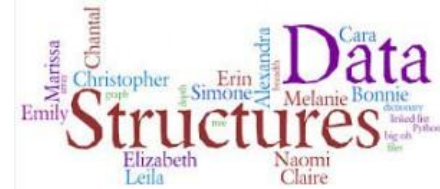
- 每個節點最多只有兩個子節點,可為空

```
typedef struct node {  
    int data;  
    struct node leftChild ;  
    struct node rightChild;  
};
```

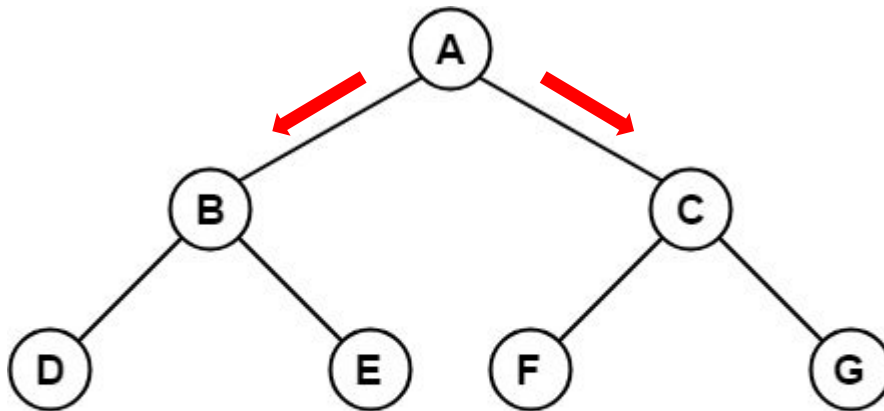




走訪



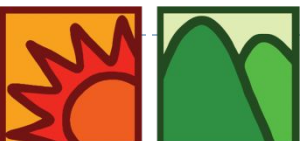
- 前序(pre_order): 中左右(DLR)
- 中序(in_order): 左中右(LDR)
- 後序(post_order): 左右中(LRD)



Pre-order: A, B, D, E, C, F, G

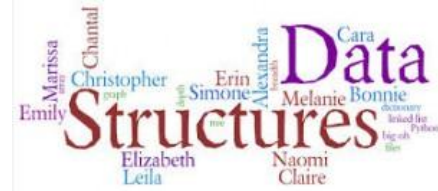
In-order: D, B, E, A, F, C, G

Post-order: D, E, B, F, G, C, A





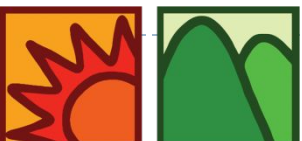
上機練習(一)

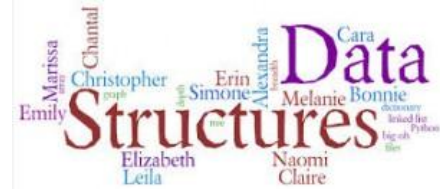


- 給定一棵二元樹，找出該樹的：
 1. 前序(pre_order)
 2. 中序(in_order)
 3. 後序(post_order)

```
Preorder Traversal: 7 43 5 13 21 9 0 4
Inorder Traversal: 13 5 43 21 7 0 9 4
Postorder Traversal: 13 5 21 43 0 4 9 7
```

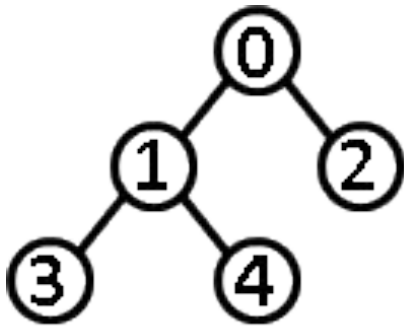
- 作業程式-> [hw] traversal.c



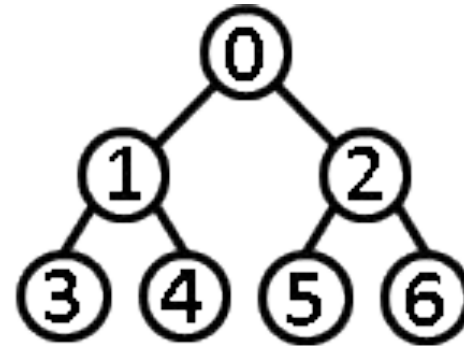


是否相同

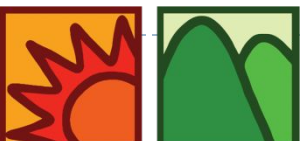
- 給定二棵二元樹，判斷樹是否相同：



Tree1



Tree2

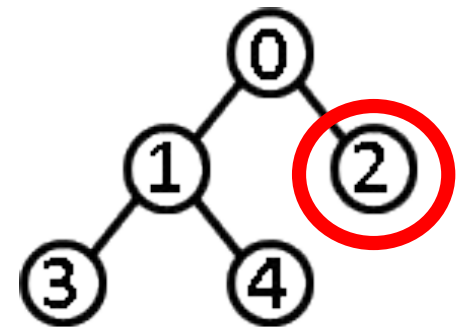




上機練習(二)

- 給定二棵二元樹，判斷樹是否相同：
- 找尋樹中是否含有某元素

```
Does tree and tree1 identical ?  
Yes  
Does tree and tree2 identical ?  
No  
Tree Search:  
Does 2 exist in tree1 ?  
No  
Does 2 exist tree2 ?  
Yes
```



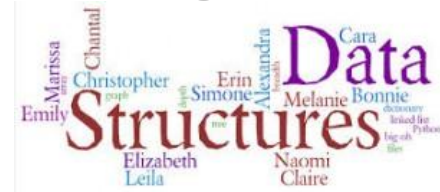
Tree I

- 作業程式-> [hw] tree_identical.c





上機練習(三)



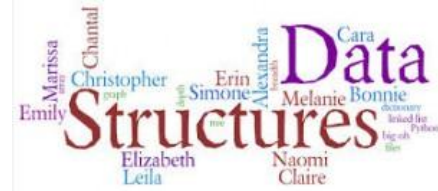
- 根據樹的中序與前序走訪結果，建立二元樹

```
tree1 preorder traversal: 4 2 1 3 6 5 7
tree1 inorder traversal: 1 2 3 4 5 6 7
tree1 postorder traversal: 1 3 2 5 7 6 4

tree2 preorder traversal: 5 9 3 8 4 7 6
tree2 inorder traversal: 8 9 5 3 7 4 6
tree2 postorder traversal: 5 3 9 4 6 7 8
```

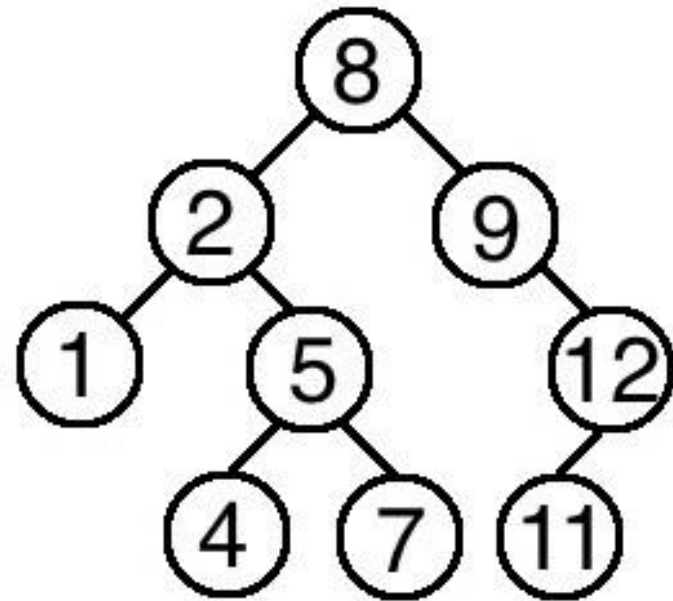
- 作業程式-> [hw] build_BT.c





二元搜尋樹

- 為一種二元樹
- 每個節點的鍵值皆不同
- 左子樹鍵值小於根(root)
- 右子樹鍵值大於根(root)
- 左、右子樹亦為二元搜尋樹
- 二元搜尋樹的中序走訪為從

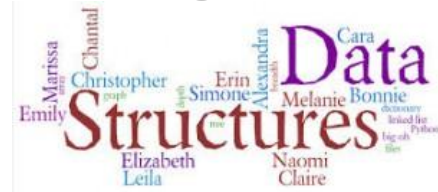


inorder: 1, 2, 4, 5, 7, 8, 9, 11, 12





上機練習(四)



- 根據題目給的數列，依序建立二元搜尋樹
- 計算：
 1. 葉節點個數
 2. 非葉節點個數
 3. 此二元搜尋樹的高度

```
inorder traversal: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Is it a Binary Search Tree(BST)?: Yes
root data: 8
the number of leaves: 8
the number of non-leaves: 7
height: 4
```

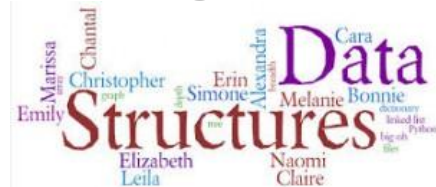
```
Modify Binary Search Tree(BST):
inorder traversal: 1 2 3 4 5 6 7 20 9 10 11 12 13 14 15
Is it a Binary Search Tree(BST)?: No
```

- 作業程式-> [hw] build_BST.c

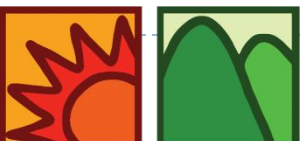




上機注意事項

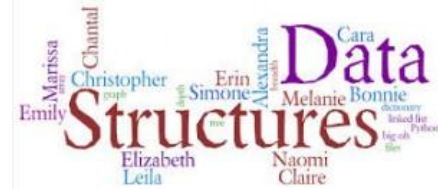


- 一律使用 C 語言撰寫!
- 程式請根據二元樹、二元搜尋樹的基本規則撰寫!
- 上機題目只要寫好其中一題, 就可以先拿給助教檢查、登記, 不用等到全部寫完!
- 若課堂上未完成, 可當作回家作業, 再於後兩周的助教時間拿給助教檢查即可!
- 程式碼不須繳交, 只須當下 DEMO 給助教看就好!
- 若無法解釋code邏輯, 不予計分!





其他事項



助教時間 (地點: CM 3038)

1. 5/3 (三) 19:00 ~ 20:00
2. 5/10 (三) 19:00 ~ 20:00

助教聯絡方式:

林妤姍:

yulei3847@gmail.com

謝博丞:

pocheng0301@gmail.com

陳亞琦:

m104020042@g-mail.nsysu.edu.tw

