

## Handwriting

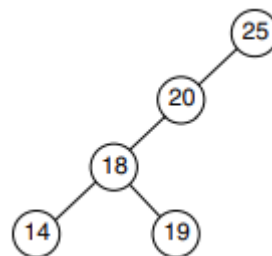
1. (25%) Create a binary search tree using the following data entered as a sequential set:

80 70 66 56 33 23 14 10 7

- 請由左至右insert, 並依left subtree < root ≤ right subtree的邏輯建出BST。

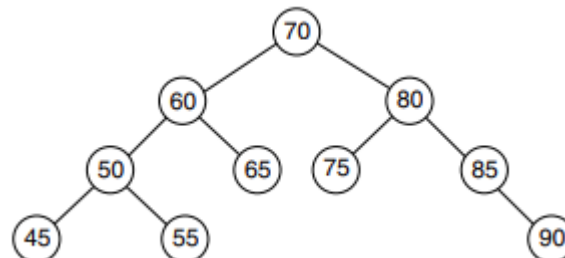
2. (25%) The binary search tree in Figure 7-19 was created by starting with a null tree and entering data from the keyboard. In what sequence were the data entered? If there is more than one possible sequence, identify the alternatives.

FIGURE 7-19



3. (25%) Delete the node containing 85 from the binary search tree in Figure 7-22.

FIGURE 7-22



4. (25%) Develop a nonrecursive algorithm for Algorithm 7-3, "Search BST."

- Write pseudo code or c-style code.

### ALGORITHM 7-3 Search BST

```

Algorithm searchBST (root, targetKey)
Search a binary search tree for a given value.
  Pre   root is the root to a binary tree or subtree
        targetKey is the key value requested
  Return the node address if the value is found
        null if the node is not in the tree
1 if (empty tree)
  Not found
  1 return null
2 end if
3 if (targetKey < root)
  1 return searchBST (left subtree, targetKey)
4 else if (targetKey > root)
  1 return searchBST (right subtree, targetKey)
5 else
  Found target key
  1 return root
6 end if
end searchBST
    
```

## Programming

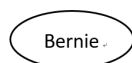
5. (50%)

Write a program that reads a list of names and telephone numbers from a text file and inserts them into a BST tree. Once the tree has been built, present the user with a menu that allows him or her to search the list for a specified name, insert a new name, delete an existing name, or print the entire phone list. At the end of the job, write the data in the list back to the file. Test your program with at least 10 names.

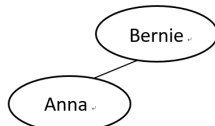
- 請使用BST進行實作，Tree node依個人需求決定其結構，若非使用BST則不予計分
- Insert時比較名字的字母的大小來決定插入位置

Ex.

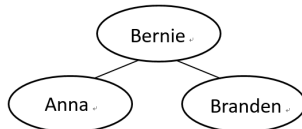
1. insert Bernie



2. insert Anna :  $A < B$



3. insert Branden :  $B = B, r > e$



- 讀取同目錄下 [input.txt]，每組資料包含一個英文名字加上一串電話號碼(10碼)，之間以一個space隔開(請在自己要執行的作業系統重新生成測資)，如下所示

```
input.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
Jay 0912121212
Branden 0934343434
Bernie 0956565656
Jason 0978787878
Eric 0990909090
```

Input 內包含一至十組資料

- 須包含以下功能：**搜尋名稱**、**新增資料**、**刪除現有資料** 以及 **印出整個電話表**
- 選擇功能的選項以數字表示：
  - 1->search 2->insert new 3->delete 4->印出整個電話表 5->Exit
- 在Terminal中**循環**讓使用者選擇要執行何種功能或退出，並依所選功能Print出結果，
  - 搜尋名稱：Input 一個名字，Output 名字+電話號碼，若輸入不存在名字則 Output None
  - 新增資料：Input 一組名字加電話號碼，無須Output，若有輸入相同名字的情況，則更新Tree中該名字的電話號碼
  - 刪除現有資料：Input 一個名字，無須Output，刪除後須維持BST的樹狀性質，若輸入不存在名字則Output None
  - 印出整個電話表：無須Input，Output格式如範例輸出

範例輸出：

圖片中執行的動作為：

- step1: 輸入1 輸入Tony 名字不存在印出None  
 step2: 輸入2 輸入Tony 0987878787 名字+電話號碼  
 step3: 輸入4 印出新增過Tony的電話表  
 step4: 輸入3 輸入Tony 刪除Tony  
 step5: 輸入4 印出刪除Tony的電話表  
 step6: 輸入5 離開

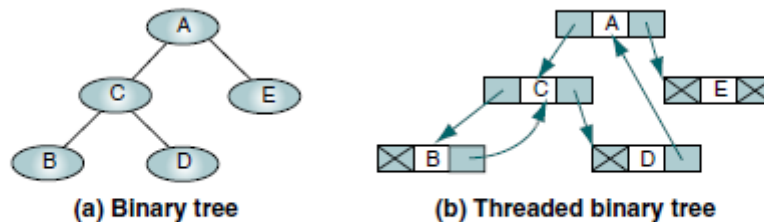
```

andrew@andrew: ~/Downloads
File Edit View Search Terminal Help
(base) andrew@andrew:~/Downloads$ ./a.out
1
Tony
None
2
Tony 0987878787
4
Bernie 0956565656
Branden 0934343434
Eric 0909090909
Jason 0978787878
Jay 0912121212
Tony 0987878787
3
Tony
4
Bernie 0956565656
Branden 0934343434
Eric 0909090909
Jason 0978787878
Jay 0912121212
5
(base) andrew@andrew:~/Downloads$
  
```

**\*\*請不要印出和輸出無關的文字\*\***

6. (50%)

Write a program that processes a threaded binary tree. The program should first build the tree, then use an iterative traversal to process it using the threads.



### Threaded Binary Tree

本題題目為：

- I. 依照節點輸入順序建立一棵 threaded binary tree, 同時它也要滿足 binary search tree 的條件, key 即輸入的那些整數數值。
- II. 用 iterative 版本的 inorder traversal 來檢查樹的正確性, 共要執行兩遍：
  - A. 第一遍要用 **stack** 來完成
  - B. 第二遍則要用到 **threaded binary tree** 的特性來完成 (可參考課本 P335)

輸入/輸出規則：

- 本題採用標準輸入, 標準輸出 (即打字輸入後, 輸出在 terminal 上)
- 輸入: 一連串整數, 每個整數用一個空格隔開
- 輸出: 共有兩行, 分別是 A 和 B 的結果 (理應來說兩行結果會一樣)

其他：

- 樹的建立為求統一, 請依照輸入順序去做插入節點的動作, 可參考下面範例。
- 兩次執行皆需要用 iterative 版本的 traversal, 若用 recursive 的方式去寫則不給分。
- 上述 A 的執行 stack 可使用 C++ STL container 或是之前作業寫過的 ADT, 包在 .h 檔中或是放在同一個檔案裡皆可。
- 上述 B 的執行需在空間複雜度  $O(1)$  下完成 traversal, 意即不能用到 stack, 若沒有用到 threaded binary tree 的特性去做會斟酌扣分。另外, threaded binary tree 的架構沒有限定, 大致和 BST 方向相同, 可參考上面附圖去做增減, 只要能被利用來完成中序遍歷即可 (hint: 節點應該會需要增加變數紀錄指標為一般的 “link” 還是 “thread”)
- 本題學習重點是了解 threaded binary tree 的優點以及熟悉實作方法, 請依規定完成本題要求, 若有任何問題 (例如: 發現題目 bug、卡關做不出來), 歡迎到討論區發問

#### Input Format

One line. There are  $n$  integers  $X_1, \dots, X_n$ , where  $1 \leq n \leq 20$ , and  $-100 \leq X_1, \dots, X_n \leq 100$ . All the integers splitted by blankspaces.

#### Output Format

Two lines.  
First line is the result of inorder traversal by **using a stack**.  
Second line is the result of inorder traversal by **utilizing the properties of threaded binary tree**.

Sample Input	Sample Output
4 6 2 3 5 1	1 2 3 4 5 6 1 2 3 4 5 6
<b>Sample diagram</b> <div> <p>Blue: link Pink: thread</p> </div>	

```
demo@DS-HW:~/Downloads/hw7_6$ g++ HW7_6.cpp
demo@DS-HW:~/Downloads/hw7_6$ ./a.out
4 6 2 3 5 1
1 2 3 4 5 6
1 2 3 4 5 6
```

## Submission - **Deadline: 2022/12/9 13:20**

題目形式：

- 手寫題可以用手寫拍照、打字的方式完成，但最後要**統一轉成.pdf檔**繳交  
檔名為HW7\_學號.pdf。例如：**HW7\_0123456.pdf**
- **請於作業開頭標註學號與姓名，若無法判定作業是出自於本人，以檔名錯誤論。**
- 程式題則繳交程式原始碼(.c檔/.cpp檔/.h檔 if needed)  
檔名為HW7\_題號\_學號.c / .cpp。例如：**HW7\_5\_0123456.c / .cpp / .h**

繳交方式：

- 將上述**共三個檔案及h檔(if needed)**(手寫題pdf檔\*1+程式題c/cpp檔\*1)**直接上傳至e3**
- 檔名 / 格式錯誤者扣該次作業總分10分。
- 程式部分輸出格式請照作業說明，若不同會酌量扣分。

收作業規則：

- 遲交一個禮拜內分數打七折，超過一個禮拜即不接受補交。
- 遲交期限內僅接受原本沒交作業的同學補交，不接受先前交過作業的同學再次補交，若要修改答案請在繳交期限內修改完畢。
- 請務必重新整理，確認檔案已成功上傳至e3。

如有任何問題，麻煩從e3來信給所有助教。