Data Structure Homework 7

繳交期限： 2020/12/1 17:00 前

補交期限(7 折)： 2020/12/8 17:00 前

手寫題：

2. Create a binary search tree using the following data entered as a sequential set:

```
14 23 7 10 33 56 80 66 70
```

10. 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.

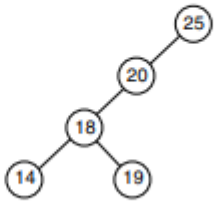


FIGURE 7-19 Figure for Exercise 10

14. Delete the node containing 85 from the binary search tree in Figure 7-22.

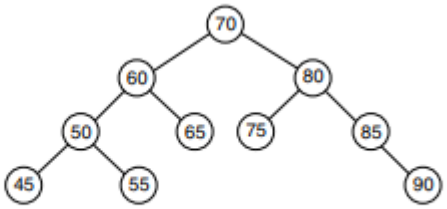


FIGURE 7-22 Figure for Exercises 13 and 14

16. Develop a nonrecursive algorithm for Algorithm 7-3, "Search BST."

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
```

說明事項: 上圖是 recursive 的寫法，同學們可以用 iterative 的寫法來達成一樣的效果。同學們只需寫出 binarysearch()的函式，裡面必須包含 4 個變數，分別為 1. array[ ]　2. 陣列最小值所在的陣列位置　3. 陣列最大值所在的陣列位置　4. 要搜尋的數值。

必須用 C / C++ 寫!!!!

程式題：

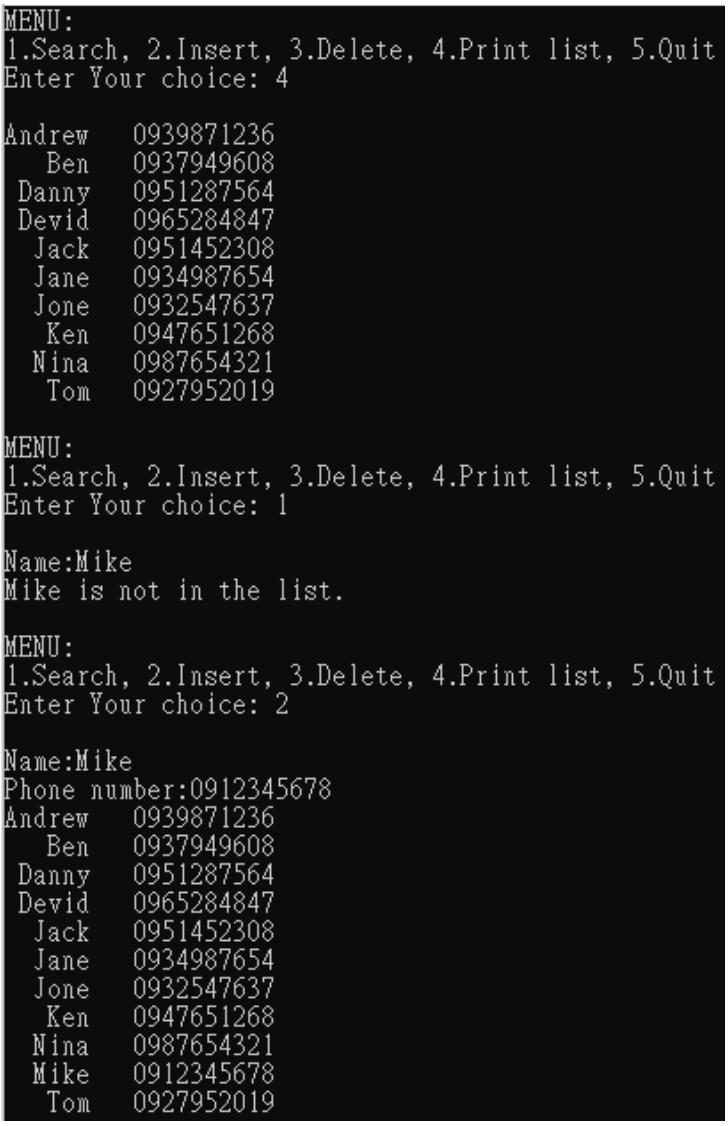
22. Write a program that reads a list of names and telephone numbers from a text file and inserts them into a BST tree. Once the tree has been built, present the user with a menu that allows him or her to search the list for a specified name, insert a new name, delete an existing name, or print the entire 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.

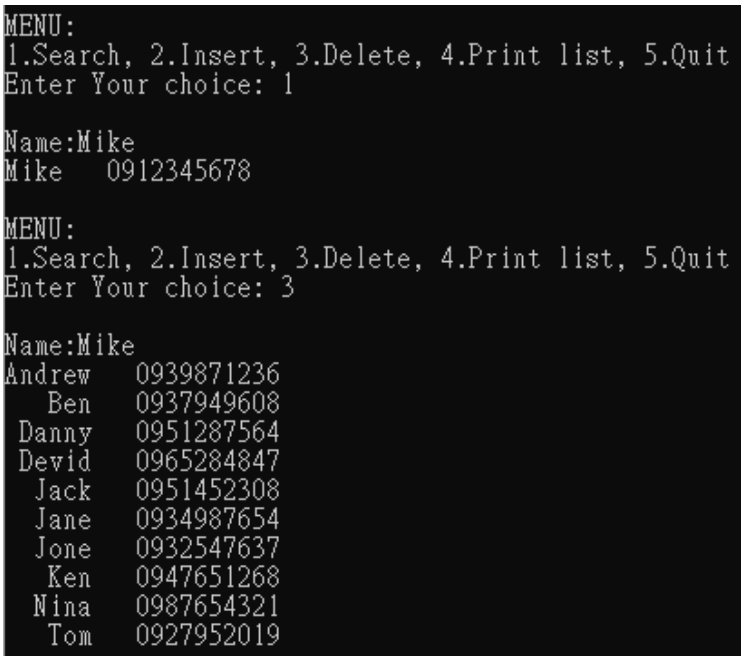
Input:

請讀取 "HW7_22.txt" 中的姓名和電話來建立初始的 BST tree，建立完後，請顯示 menu， 讓使用者輸入選擇，執行對應的動作

Output:

若選擇 Print list, Insert, Delete 請在執行後印出整個 list，若選擇 Search 請印出要搜尋的姓名及電話，可以參考下方的範例





注意事項:

1. 可以使用 ”HW7_ADT.zip” 中的提供的檔案，也可以自己寫 BST，如果有.h 檔，請將檔案命名為 “學號_姓名_hw7_22.h”，最多只能繳交一個.h 檔
2. 繳交檔名為“學號_姓名_hw7_22.cpp” 或 “學號_姓名_hw7_22.c”
3. 不需要壓縮，不要上傳 .h, .c, .cpp 以外的檔案
4. 請確保你的程式可以能被 g++或 gcc 編譯
5. 不符合以上格式者直接扣 2 分

24. 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.

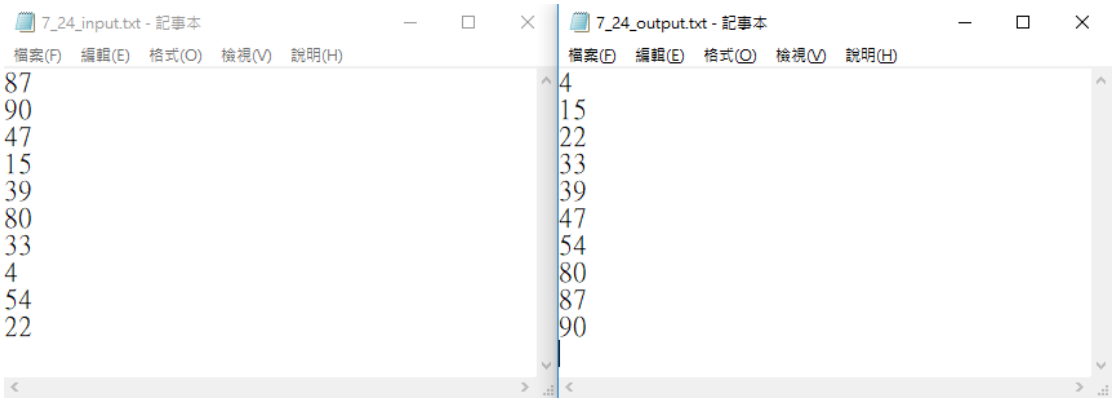
Input:

請依序讀取 “7_24_input.txt” 中的數值，數值皆為 0~99 之間的整數(包含 0 和 99)，實際測資為 50 筆，並建立 threaded binary tree (inorder)，資料的排序方法按照 binary search tree。

這邊提供一個方法，可以先按讀取進來的順序建立 binary search tree，再改良成 threaded binary tree (inorder)。

Output:

將”iterative” inorder traversal 的結果 output 至“7_24_output.txt”。



評分標準:

1. 有使用 threaded BST (3 分)
2. 有使用 iterative traversal (3 分)
3. 結果正確 (3 分)
4. 程式無法執行則只給 1 分
5. 無法正確讀取“7_24_input.txt” (扣 1 分)
6. 沒有正確輸出至“7_24_output.txt” (扣 1 分)

注意事項:

1. 注意是 iterative traversal，非 recursive traversal
2. 可以使用 ”HW7_ADT.zip” 中的提供的檔案，也可以自己寫 BST，如果有.h 檔，請將檔案命名為 “學號_姓名_hw7_24.h”，最多只能繳交一個.h 檔
3. 繳交檔名為“學號_姓名_hw7_24.cpp” 或 “學號_姓名_hw7_24.c”
4. 不需要壓縮，不要上傳 .h, .c, .cpp 以外的檔案
5. 請確保你的程式可以能被 g++或 gcc 編譯
6. 不符合以上格式者直接扣 2 分