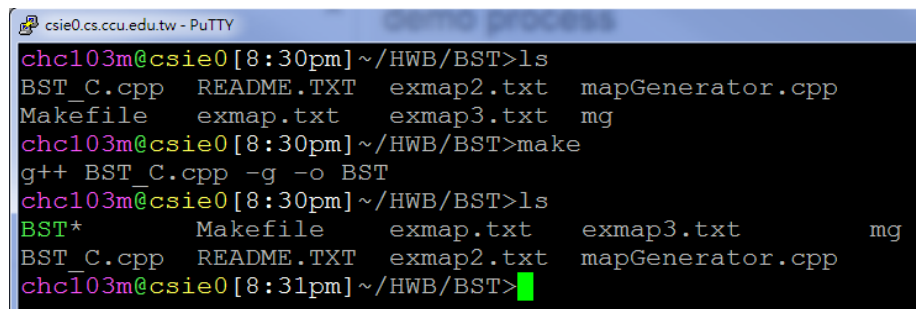


# Assignment B | Binary Search Tree and Treasure Hunter

TA: Chen ([rainywindsnow@yahoo.com.tw](mailto:rainywindsnow@yahoo.com.tw)) Deadline: Nov. 19th (Thursday) 11:59pm

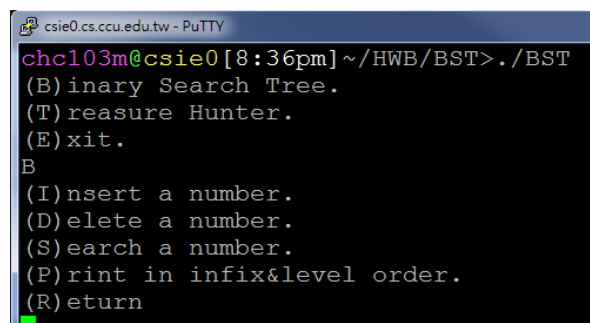
## Demo Steps

1. Use a **Makefile** to compile your program. (Make sure that you can compile and run your program successfully. Otherwise, you will not get the scores of HWB.)



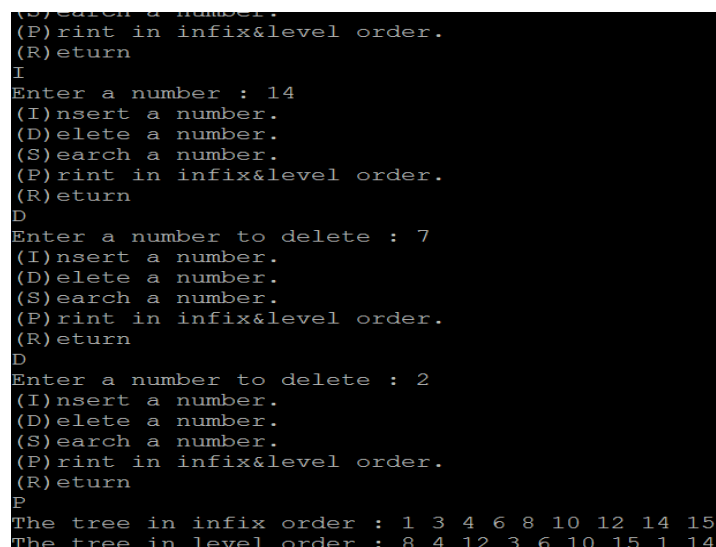
```
csie0.cs.ccu.edu.tw - PuTTY
chc103m@csie0[8:30pm] ~/HWB/BST>ls
BST_C.cpp  README.TXT  exmap2.txt  mapGenerator.cpp
Makefile   exmap.txt   exmap3.txt  mg
chc103m@csie0[8:30pm] ~/HWB/BST>make
g++ BST_C.cpp -g -o BST
chc103m@csie0[8:30pm] ~/HWB/BST>ls
BST*      Makefile   exmap.txt   exmap3.txt      mg
BST_C.cpp README.TXT  exmap2.txt  mapGenerator.cpp
chc103m@csie0[8:31pm] ~/HWB/BST>
```

2. You must provide a user interface to let a user choose either **Search Tree** or **Treasure Hunter**.
3. For the **Binary Search Tree**, let a user choose to insert, delete, and search a number. Print the binary search tree in infix order and level order.



```
csie0.cs.ccu.edu.tw - PuTTY
chc103m@csie0[8:36pm] ~/HWB/BST>./BST
(B)inary Search Tree.
(T)reasure Hunter.
(E)xit.
B
(I)nsert a number.
(D)elete a number.
(S)earch a number.
(P)rint in infix&level order.
(R)eturn
```

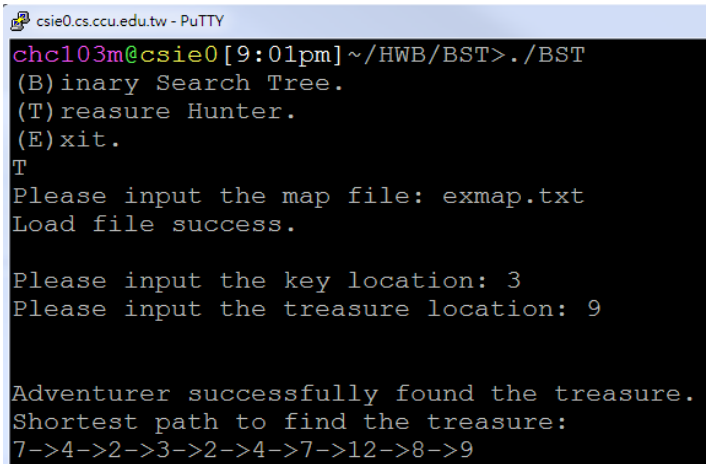
4. For example, sequentially insert the numbers (7, 4, 2, 12, 8, 10, 6, 15, 1, 3, 14). Then delete the numbers (7, 2) sequentially. The print result is as follow:



```
(S)earch a number.
(P)rint in infix&level order.
(R)eturn
I
Enter a number : 14
(I)nsert a number.
(D)elete a number.
(S)earch a number.
(P)rint in infix&level order.
(R)eturn
D
Enter a number to delete : 7
(I)nsert a number.
(D)elete a number.
(S)earch a number.
(P)rint in infix&level order.
(R)eturn
D
Enter a number to delete : 2
(I)nsert a number.
(D)elete a number.
(S)earch a number.
(P)rint in infix&level order.
(R)eturn
P
The tree in infix order : 1 3 4 6 8 10 12 14 15
The tree in level order : 8 4 12 3 6 10 15 1 14
```

5. For the Treasure Hunter

- (1) let a user read a map file (during the demo, we will use different map files)
- (2) let a user input a key location
- (3) let a user input a treasure location
- (4) print the result



```
chc103m@csie0[9:01pm]~/HWB/BST>./BST
(B)inary Search Tree.
(T)reasure Hunter.
(E)xit.
T
Please input the map file: exmap.txt
Load file success.

Please input the key location: 3
Please input the treasure location: 9

Adventurer successfully found the treasure.
Shortest path to find the treasure:
7->4->2->3->2->4->7->12->8->9
```