演算法作業\_Binary Search Tree

楊子右 電機三甲 E24066250

編譯環境:

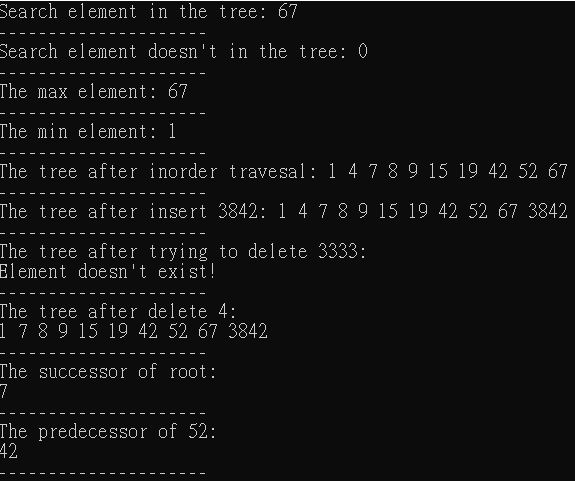
Visual Studio 2017,

Linux gcc version 7.5.0

Input = ”bt\_input.txt”



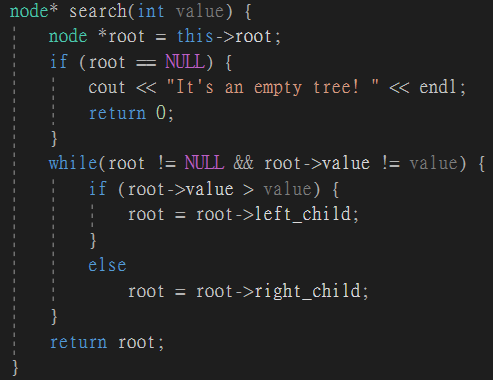
Output = cout in command line



* 二元搜尋樹之實作與其功能實現

1. Search

Code:

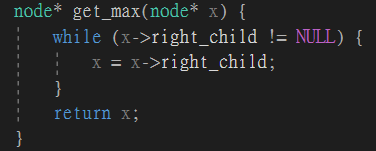


想法:

從root開始往下找，由節點的值與希望節點的值比大小來判斷要走左還是右子樹。

1. Max

Code:

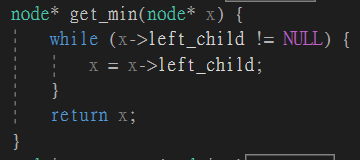


想法:

一路往右子樹找直到空節點。

1. Min

Code:

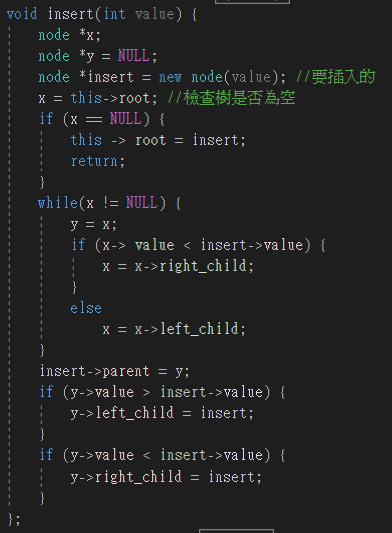


想法:

一路往左子樹找直到空節點。

1. Insert

Code:

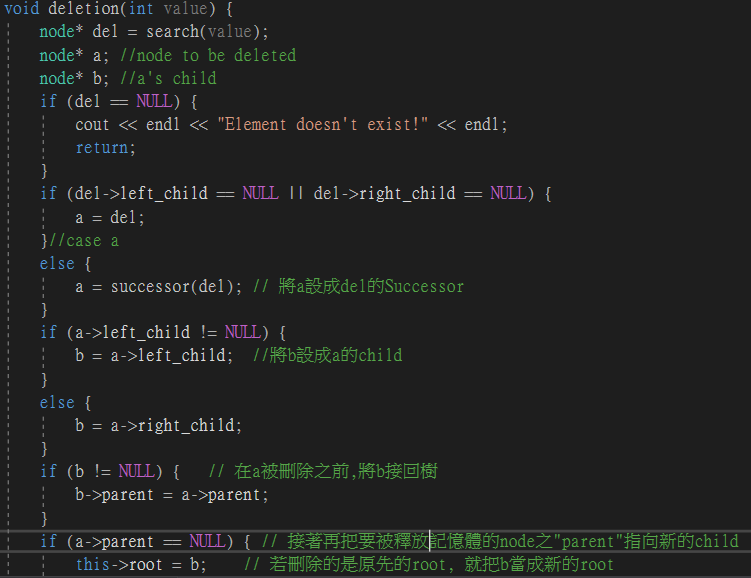


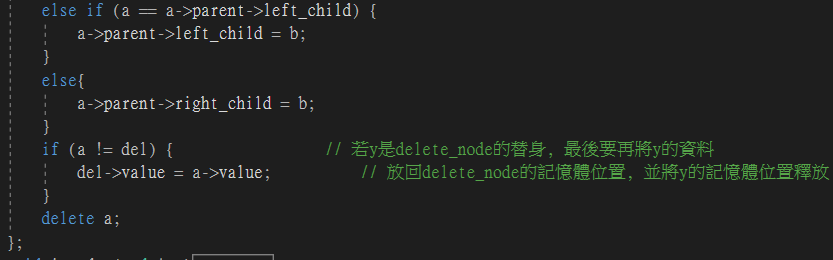
想法:

先分配記憶體空間給預計插入的node並檢查樹是否為空，將y令為x的上一階段節點，x與插入節點的值相互比較來決定走左還是右子樹。當走到leaf時，將y指派為node的父節點因為他是x節點的上一狀態，再由y的值來決定插入左還是右子樹。

1. Deletion

Code:



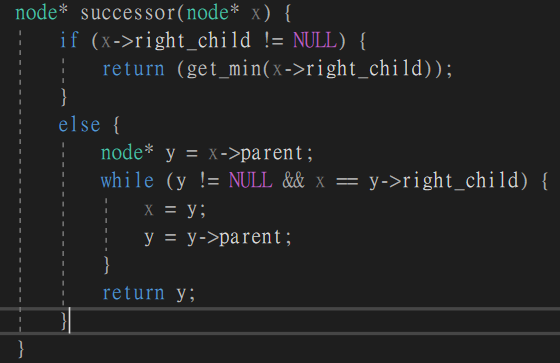


想法:

若要刪除的節點左或右子節點不為0的話則為課本所提case A 直接刪除即可。而之後礙於篇幅由註解解釋。

1. Successor

Code:

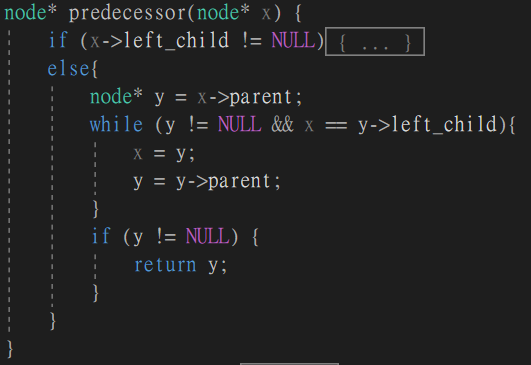


想法:

如果該節點之右子樹不為空的話則找其的最小節點即為Successor。如果為空的話就要往上找第一個最小的祖先節點。

1. Predecessor:

Code



想法:

如果該節點之左子樹不為空的話則找其的最大節點即為Predecessor。如果為空的話就要往上找第一個最大的祖先節點。