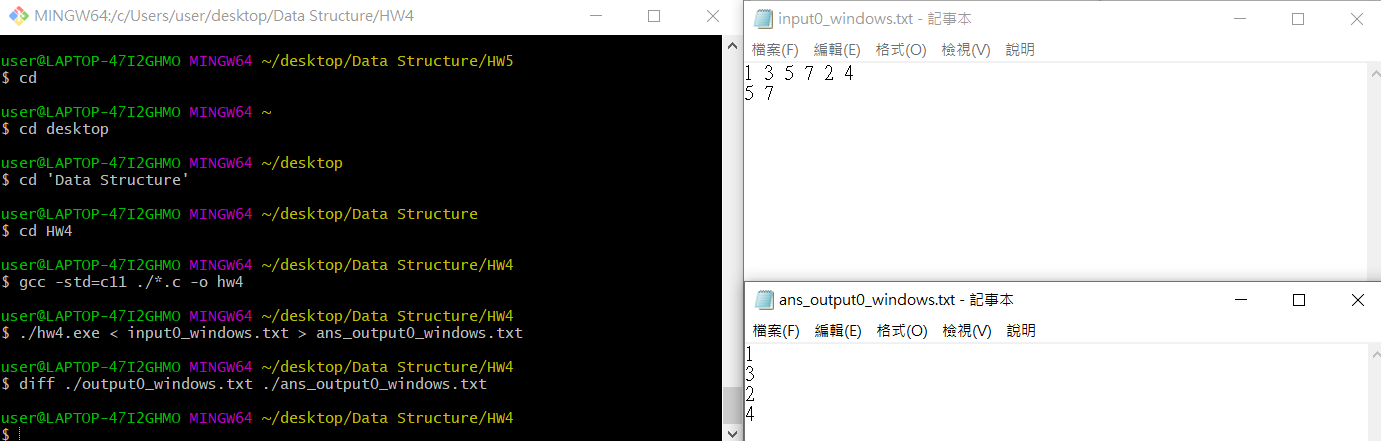
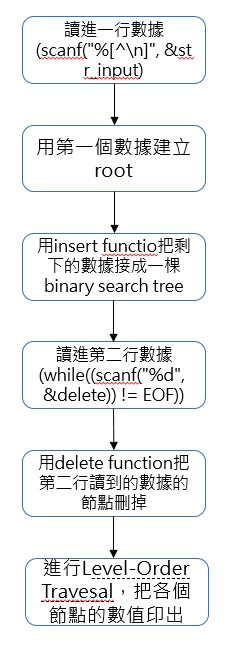
1. Result screenshot



1. Program structure



1. Program functions

(1)

tree\_pointer max(tree\_pointer node):用來尋找存放整個樹最大值的node

node – 用來不斷指向右邊的node(最右邊最下面的node存放的值即是整個數最大的值)

return – 整個樹最右邊最下面的node

(2)

tree\_pointer insert(tree\_pointer node, int data):用來新增node

node – 如果此node為NULL則新增為新的node，如果不是就用來指向下一個node(用data進行大小比對來看要放在右邊還左邊)

data – 要存放在node裡的值

return – 要新增或進行下一次比對的node

(3)

tree\_pointer delete\_node(tree\_pointer node, int data):用來刪除node

node – 用來尋找要刪除的node

data – 存放在要刪除的node裡的值

return – 透過比對不斷回傳node，最後找到要刪除的node

(4)

void traversal(tree\_pointer root):用來把整棵樹印出來(Level-order)

root – 整棵樹的root