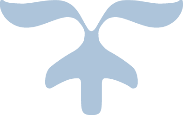


資料結構-HW02

報告書



1114522 林暐倫

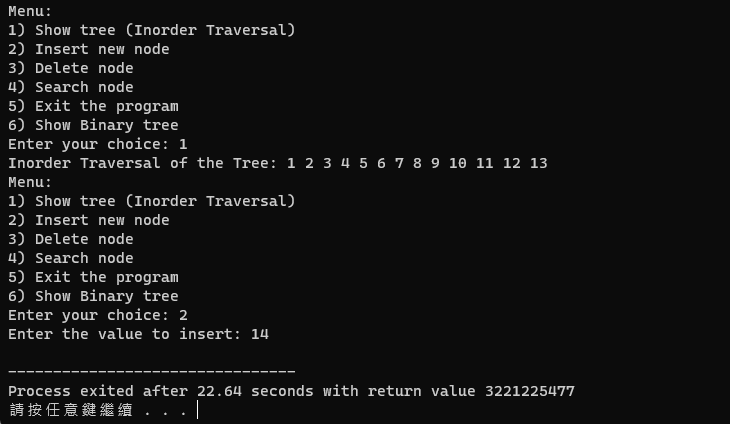
1. 作業概況與程式特性:

**已經完整完成作業需求。**  
(1)可以正常執行作業所要求的五個功能，有做出以圖形顯示的功能，一開始Binary search tree 中並無任何資料。  
  
(2) Inorder Traversal sequence的特點是節點值從小到大開始排列，一種先從左子樹到右子樹的按順序排列，Binary search tree具有以下特性：左子節點的值小於父節點的值，父節點的值小於右子節點的值。這些特性導致了在Inorder Traversal時以從左到右的方式依序訪問節點。

1. 使用手冊:

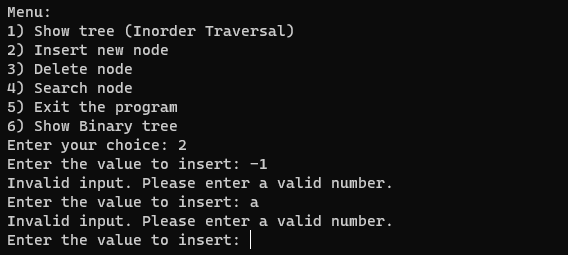
1.使用說明

(1)陣列設定為10000個數字，最多能夠新增13層，2的13次方等於8192，需要注意不要一直往右節點填寫。舉例:從1開始新增值到13時，一直往右時，到新增14會有緩衝區溢出（buffer overflow）。

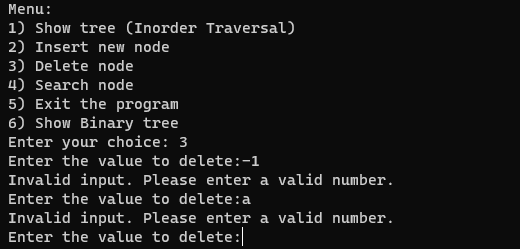


(2)使用者輸入要新增、刪除、搜尋的值為整數，例外是**-1**，會要求使用者重新輸入， 因為我先將陣列每個節點都設成**-1**，故使用者不能輸入**-1**，會顯示無效的輸入。當使用者輸入英文字也會無效的輸入。

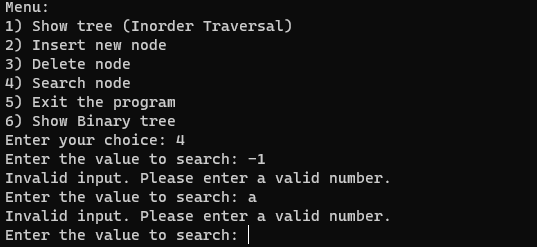
新增 **-1**、**a**:



刪除 **-1**、**a**:



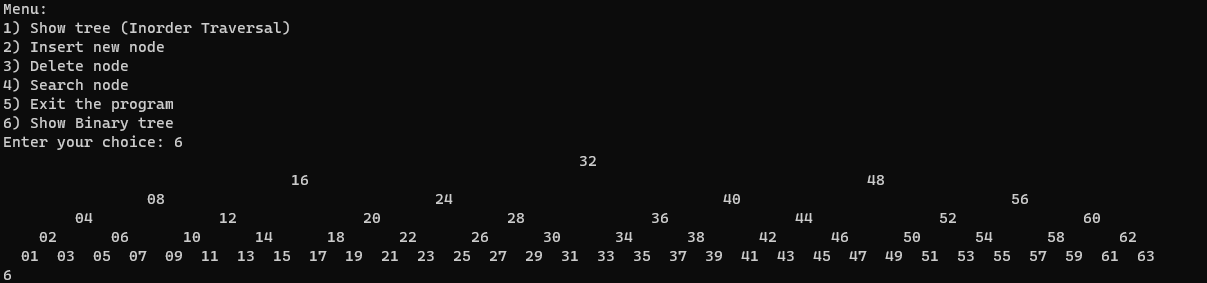
搜尋 **-1**、**a**:



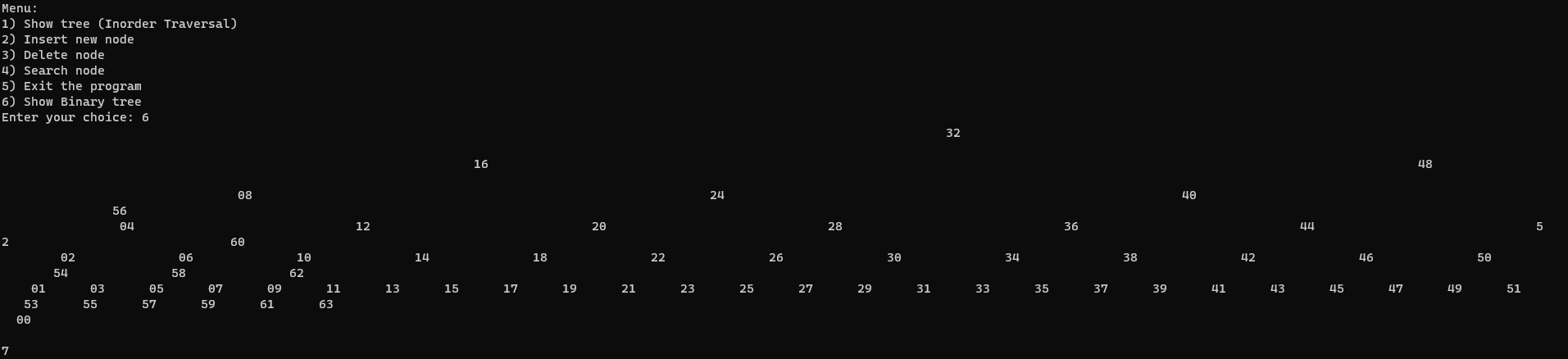
(3)如果要顯示出樹的圖形，建議將視窗放到最大，這樣會有良好的觀看。當樹到第7層時，會因為螢幕太小而跑掉的畫面，值的大小盡量為2位數，會對稱

。

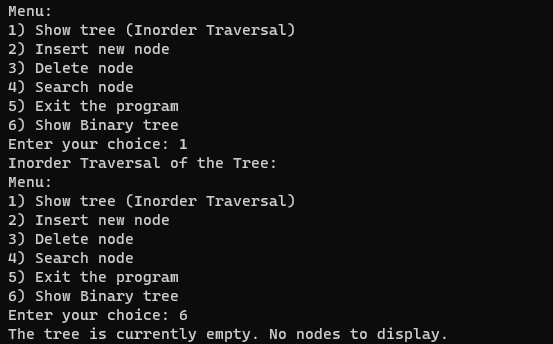
6層(最下面的數字為層數): 64個數字



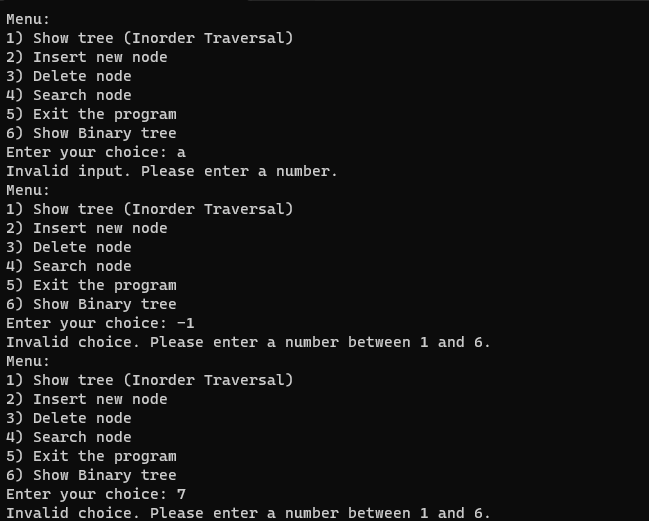
7層(最下面的數字為層數):128個數太多



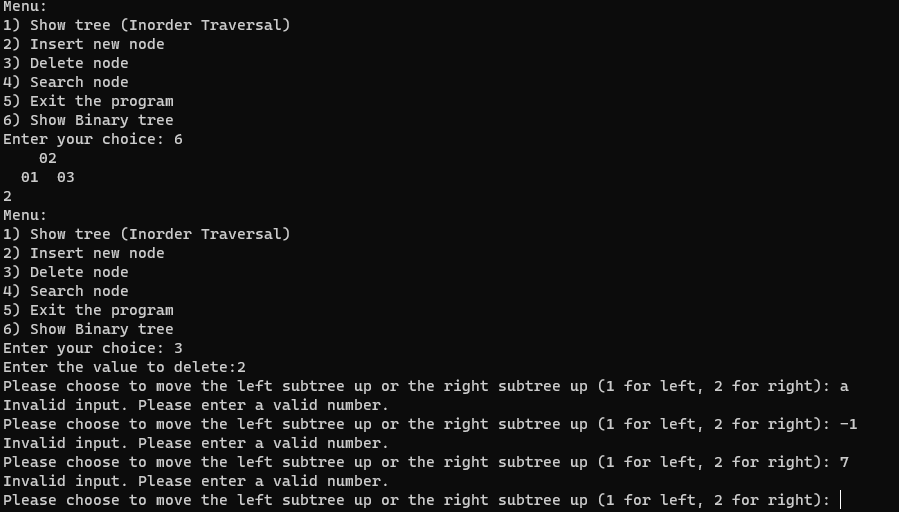
(4)未輸入值時，樹是空的



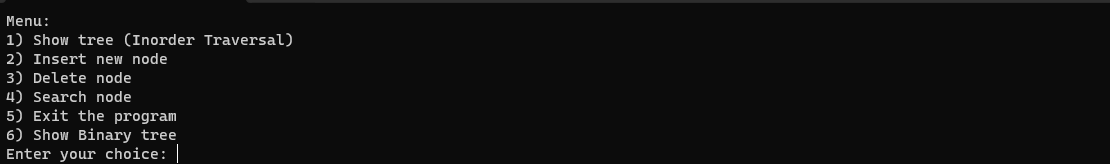
(5)當使用者輸入非選項(1~6)時會顯示無效的輸入。



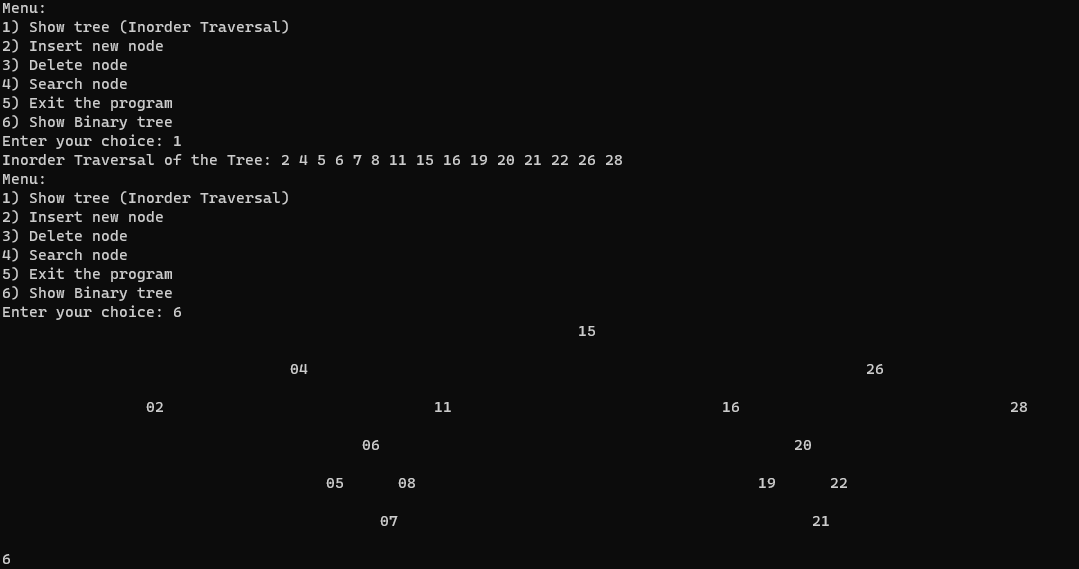
(6)在刪除兩個節點時，會出現選項(左子樹1、右子樹2)，當輸入非選項時，會 要求重新輸入



2.功能介紹  
  
 (1)程式執行時，會先跳出選項詢問使用者要使用下列哪一項功能，並以輸入選  
 項，使用個別功能

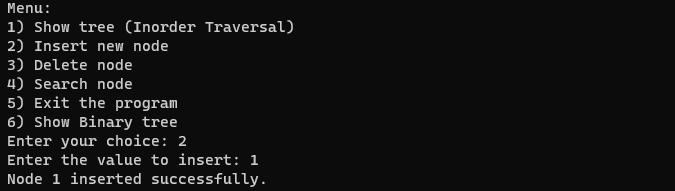


(2)功能1，以Inorder Traversal的方式顯示Binary search tree。**功能介紹都以這張圖為舉例。**

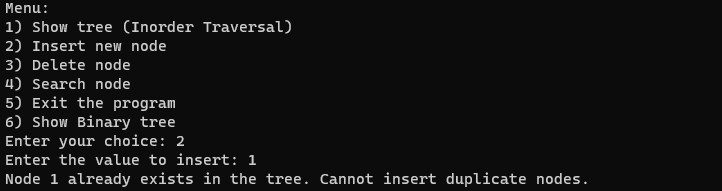


(3)功能2，新增節點

成功新增節點:1

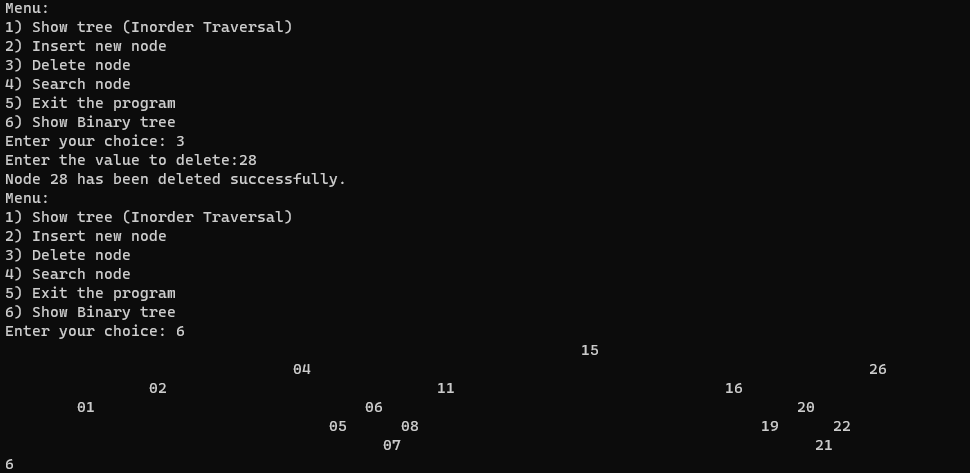


節點已經存在:1

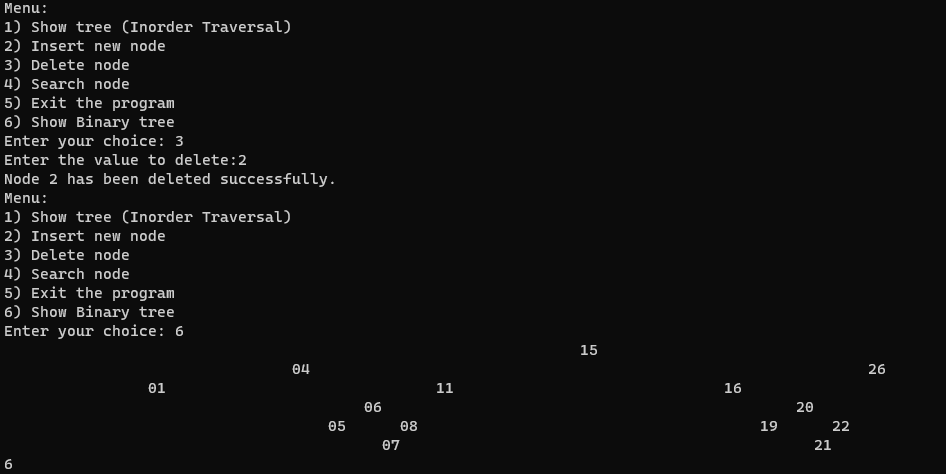


(4)功能3，刪除節點

成功刪除節點(節點沒有子節點):28

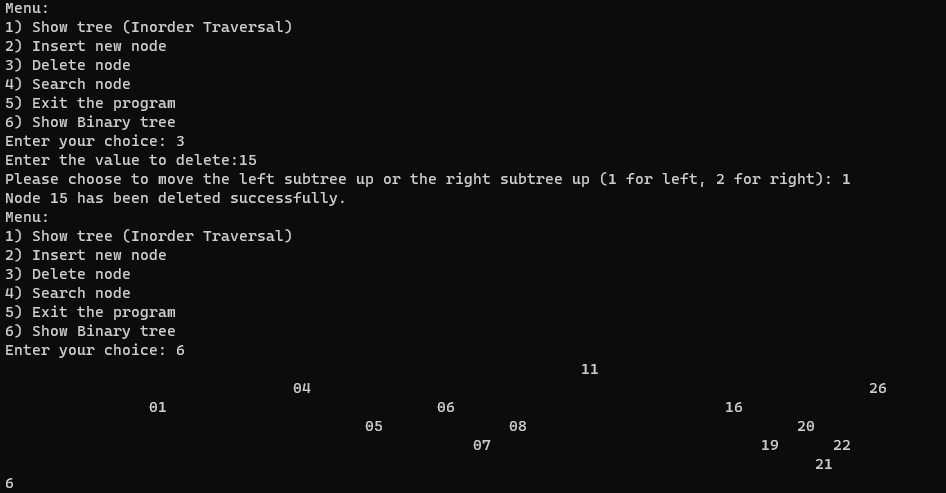


成功刪除節點(節點只有一個子節點):2

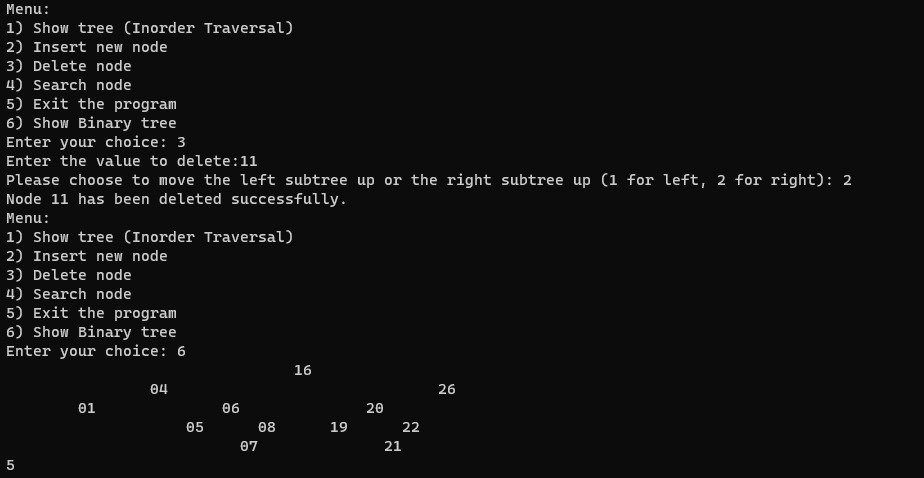


成功刪除節點(節點有兩個子節點):會詢問要搬移左子樹還是右子樹

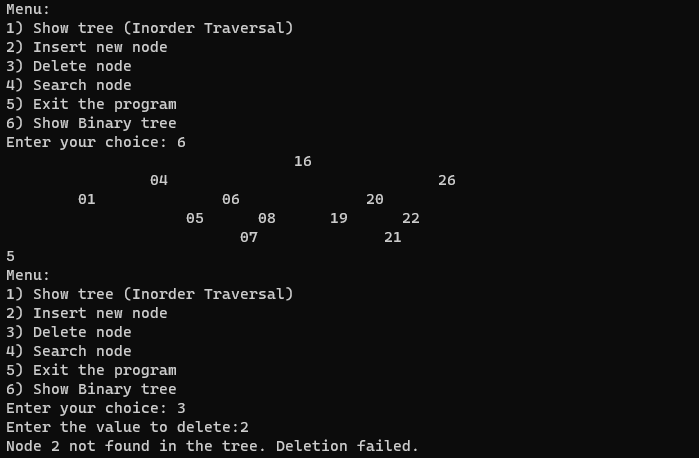
1.搬左子樹(最大值)；成功刪除15



2.搬右子樹(最小值)；成功刪除11

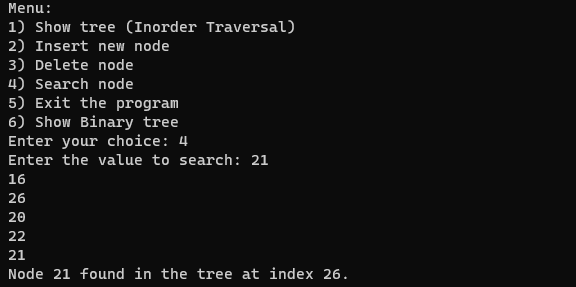


節點不存在，無法刪除:2

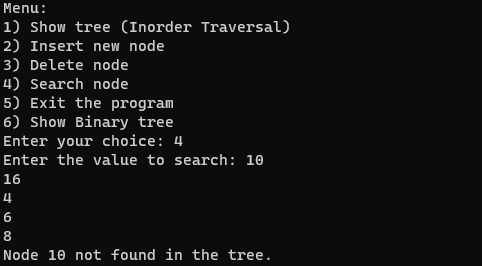


(5)功能4，搜尋節點，會出現搜尋路徑與Index值

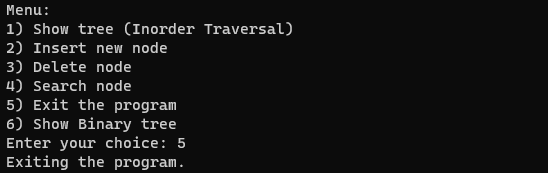
成功搜尋節點:21



未搜尋到節點:10



(6)功能5，結束程式



(7)功能6，用圖形顯示樹的形狀，最下方為Inorder Traversal的方式顯示Binary search tree

