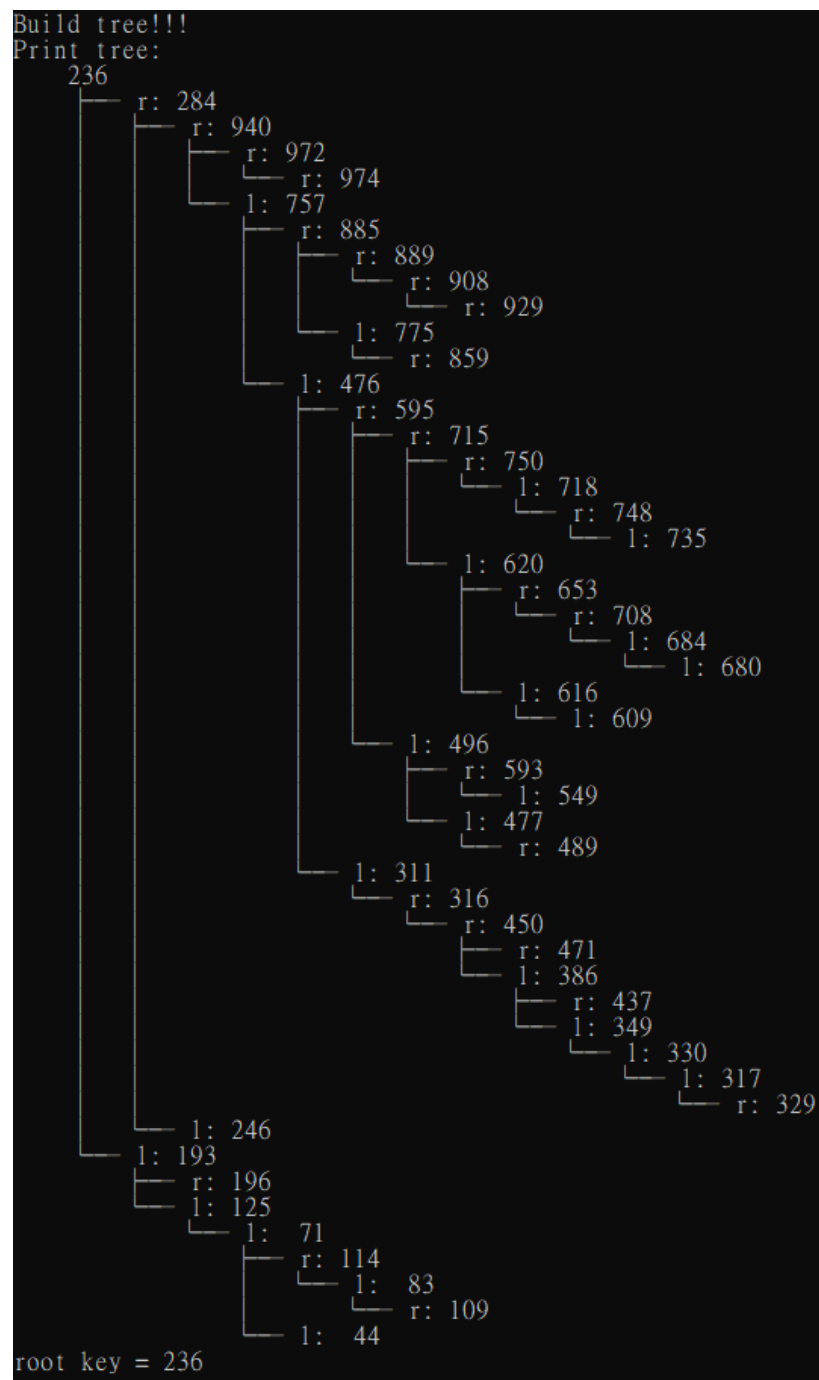


0819823 陳子祈 演算法導論 HW4

一、試驗成果

i. 將tree print出來並截圖 (l: left node; r: right node)



ii. 利用Inorder-Tree-Walk(x)將tree 中的元素按照大小print出來並截圖

44	71	83	109	114	125	193	196	236	246	284	311	316	317	329
330	349	386	437	450	471	476	477	489	496	549	593	595	609	616
620	653	680	684	708	715	718	735	748	750	757	775	859	885	889
908	929	940	972	974										

iii. 利用Tree-Minimum(x)將最小值print出來並截圖

Min = 44

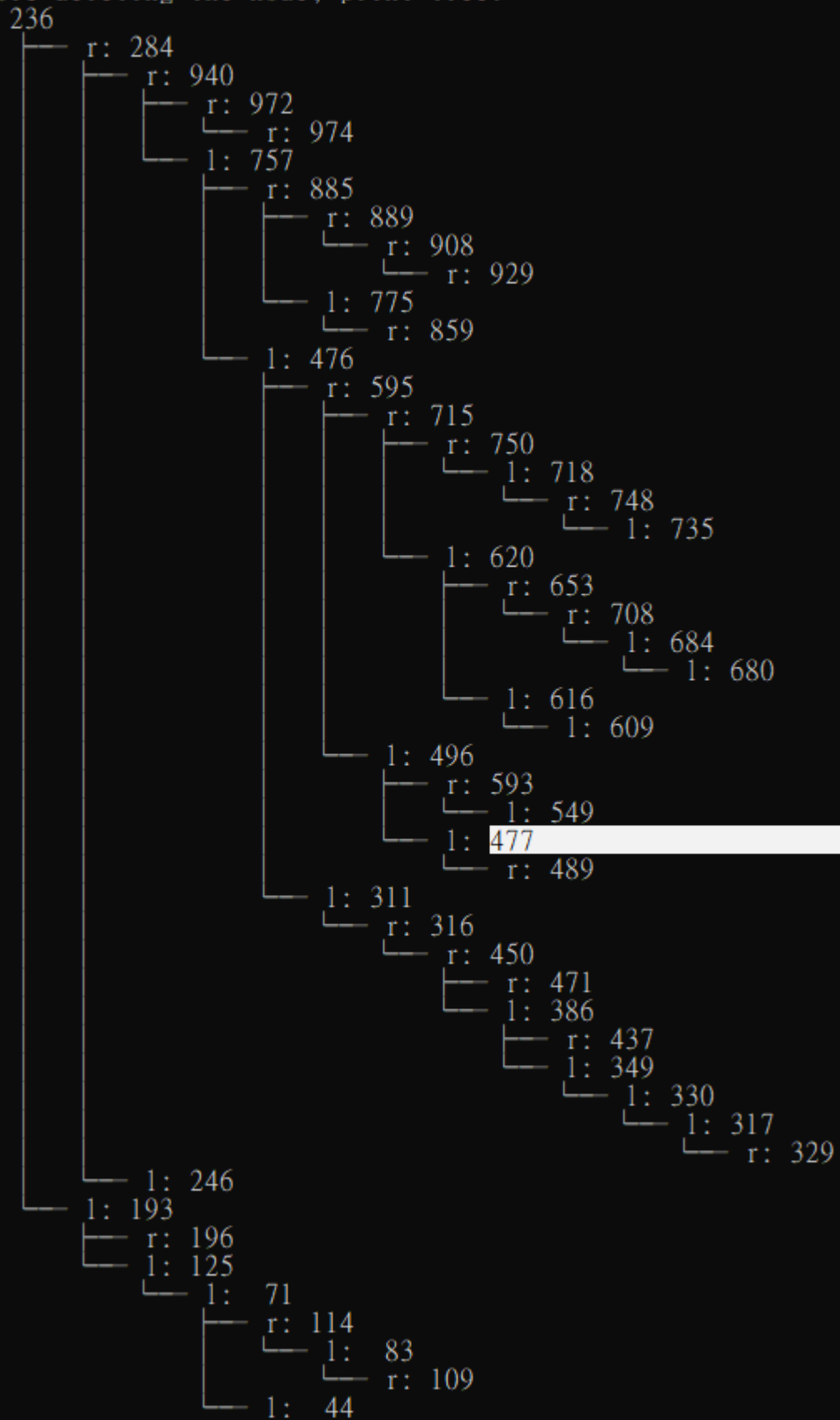
iv. 利用Tree-Maximum(x)將最大值print出來並截圖

Max = 974

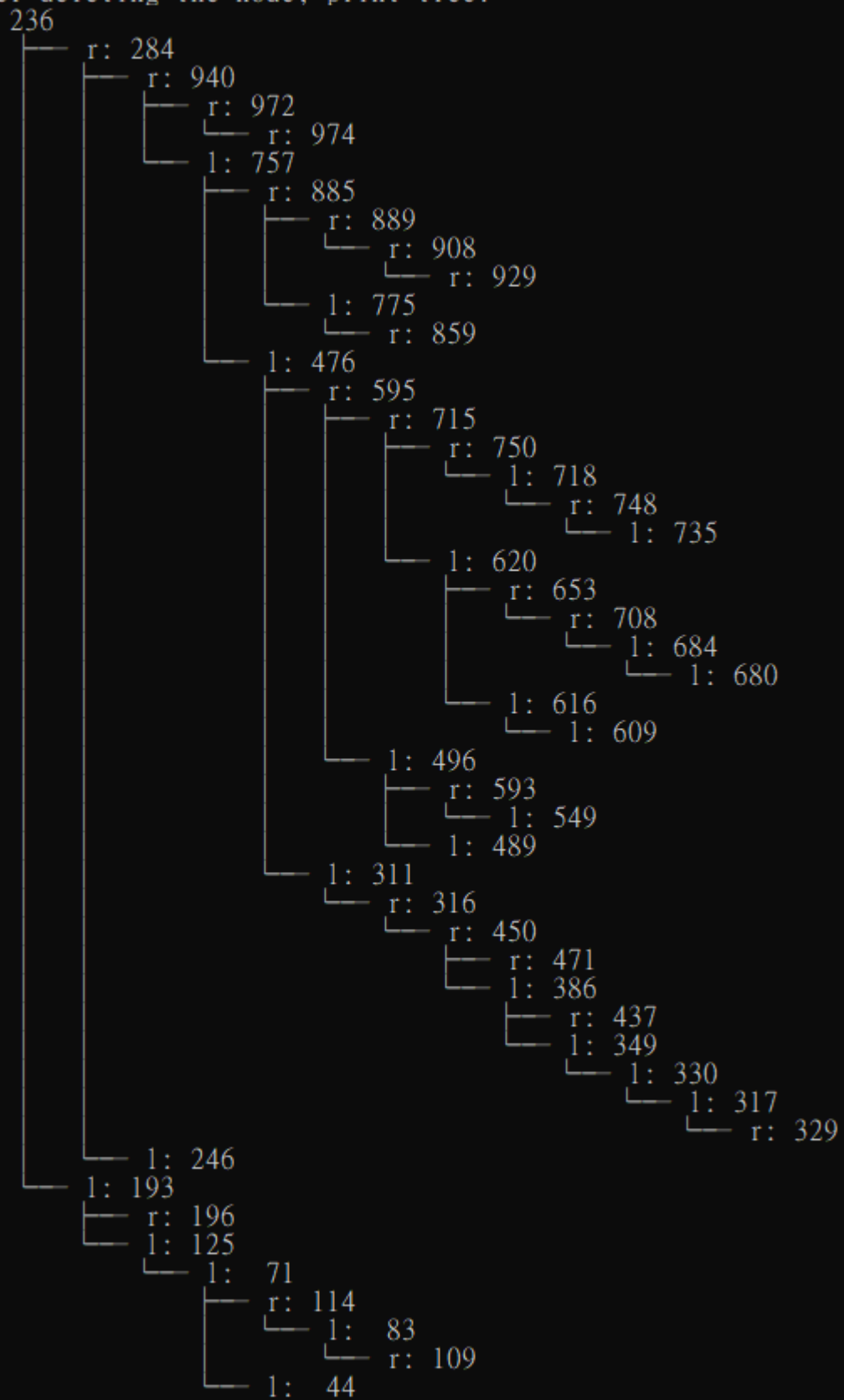
v. 利用Tree-Delete(T,z)自己選一個node刪除(可先用Tree-Search(x,k)

得到該node) · 將delete完後的tree print出來並截圖 (l: left node; r: right node)

Delete node with key = 477
Before deleting the node, print tree:

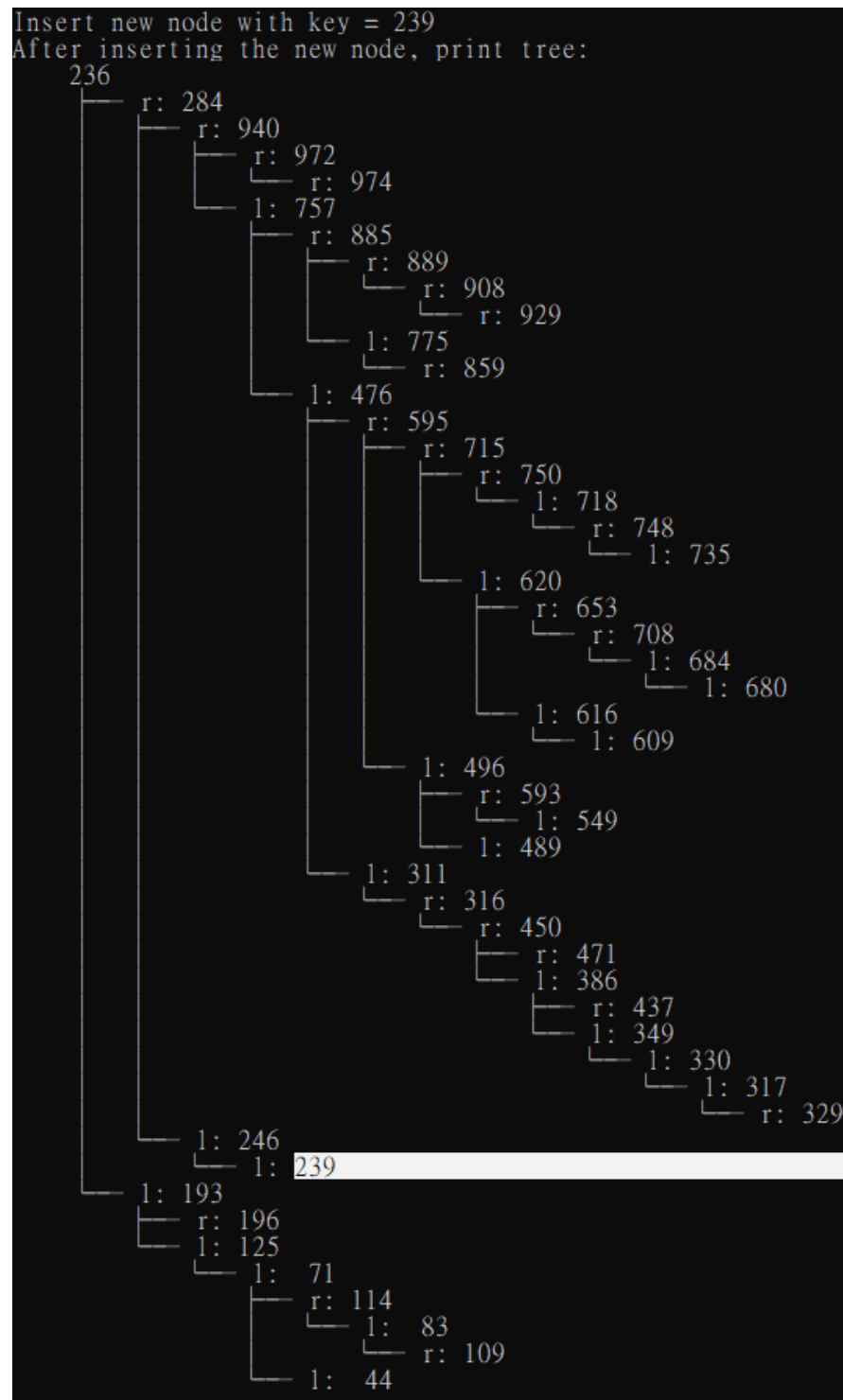


After deleting the node, print tree:



vi. 自己做一個node利用Tree-Insert(T,z) insert到tree裡，將insert完後的tree

print出來並截圖 (l: left node; r: right node)



二、心得

此作業與前幾次很不一樣，沒有需要分析程式的時間複雜度，但是除了要寫出講義裡面的幾個 **binary search tree** 的函式之外，還要印出 **2D** 樹圖。我有參考助教提供的方法印出 **2D** 樹圖，但是因為沒有分支的圖案，看不太出來 **parent** 與 **child node** 的關係，所以整張圖會看不太懂，我在網路上找到類似助教提供的 **recursion** 的方法來印樹圖，不過助教提供的方法使用 **inorder traversal (right-root-left)** 上到下印樹圖，我使用 **root-right-left** 上到下印樹圖，雖然左上到右下斜著印出樹圖，但是很好看懂，程式也比較簡單。如果真的要印出很精美的樹圖，程式非常複雜，無法使用遞迴的方式解，希望以後有空可以想看看怎麼解。