|  |  |  |
| --- | --- | --- |
|  | 功能簡介說明  Binary Search Tree | 圖片 |
| 新增(Insert) | 1.先判斷是有root  2.要插入的值和root比較，如果較小的話會往左半部，而如果較大的話則往右半部  3.繼續往下執行，會在遇到空值可以填入的時候停止。  \*比較特別的時候是遇到重複的數，會先有一個暫存的位置，再相接。 |  |
| 刪除(Delete) | 主要有4種情形，分別為無子節點、一個左子節點、一個右子節點以及兩個子節點。**一個左子節點最複雜。因為會考慮到重複的數值。**  無子節點-只需要直接刪除，不需考慮其他狀況。  一個右子節點-刪除後，將子節點往上移動。  一個左子節點-刪除後，將子節點往上。但是有可能會遇到重複的情形。  兩個子節點-考慮左側選取最大值，右側則選取最小值。 |  |
| 查詢(Search) | 1.搜尋方式如下，一開始比大小，然後再分別向下或是向左移動。直到找到值。 |  |
| 修改(Modify) | 1.運用Delete、Insert兩個概念結合。  2.先Delete再 Insert。 |  |

參考資料來源：

<http://alrightchiu.github.io/SecondRound/binary-search-tree-searchsou-xun-zi-liao-insertxin-zeng-zi-liao.html>

<https://emn178.pixnet.net/blog/post/94574434>

<http://squall.cs.ntou.edu.tw/cpp/103spring/labtest/test1/BinarySearchTree.html>

<http://www.csie.ntnu.edu.tw/~u91029/BinaryTree.html>