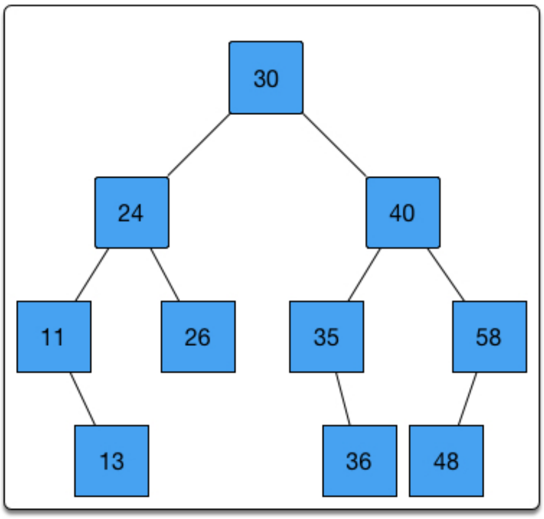
Assignment 1 Report

Haoze Xu
43400465

Exercise 2

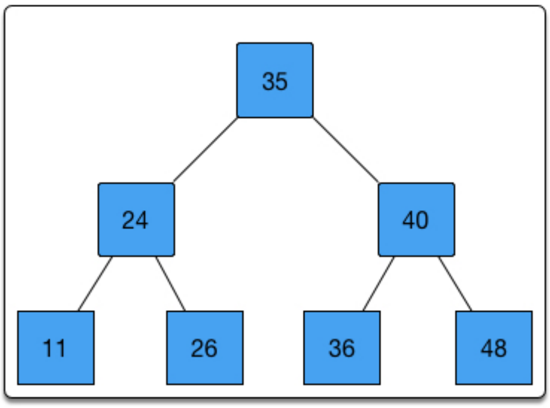
(a)



(b) The best case running time for BST insertion is O(1): all nodes are in the root's right subtree, the one to be inserted belong in the left, or all nodes are in the root's left subtree, the one to be inserted belong in the right.

The worst case time for BST insertion is O(n): all nodes are in the root's one subtree, and the one to be inserted will be the child of the leaf node.

(c)



(d) The best case time for BST deletion is O(log n): when removing a leaf node in a perfectly balanced BST.

The worst case time for BST deletion is O(n): when the tree is not perfectly balanced, remove the root node.

(e) Adding one item to a binary search tree is O(h)(h is the height of the BST), so for n element, the time complexity is O(n*h).

In the worst case, the height of the BST will be O(n), so the big-O characterization will be $O(n^2)$.

As for the lower bound, it is Ω(n log n), since the best case of the height of the BST is O(log n).

Exercise 3

(a) The constructor can be divided into two parts: the first part is traversing the string(an infix expression) to a postfix expression, and the second part is linking the nodes. Assuming that the length of the string is n, the time complexity in the first part would be O(n), for the reason that there is only a for loop to traverse the string. And in the second part, there is also a for loop, but the number of elements is at most n-2(excluding the spaces and parentheses). As a result, its time complexity would also be O(n). And totally, the time complexity of the constructor is O(n).