

Project 3

Binary Search Tree

1)

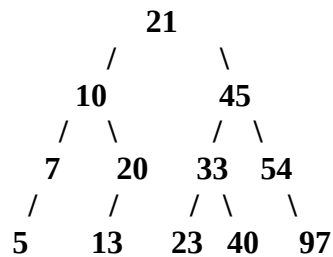
a) To create a balanced Binary Search Tree with the following numbers ($\{20, 10, 7, 13, 5, 22, 23, 54, 97, 33, 45, 21, 40\}$), insert in the following order:

22, 10, 45, 7, 20, 33, 54, 5, 13, 21, 23, 40, 97

b) The preorder of this tree:

22, 10, 7, 5, 20, 13, 21, 45, 33, 23, 40, 54, 97

c) The tree (with the original root eliminated):



2)

Using a Tree Sort algorithm would retain a best case running time of $O(n)$, and a worst case running time of $O(n^2)$.

The best case running time can be calculated by observing that the best case running time for inserting one element into a Binary Search Tree is $O(1)$ – i.e., inserting into a fairly balanced tree. Thus, if we are inserting more than one element (n elements), it can be stated that the best running time would be $O(1 * n)$.

The worst case running time can be calculated by observing that the worst case running time for inserting one element into a Binary Search Tree is $O(n)$ – i.e., inserting into a fairly unbalanced tree. Thus, if we are inserting n elements, the worst running time would be $O(n * n)$.