Yuan Chao
0027135923

ECE 368 Project 2 Report

For this project were were told to implement a searching program using a linked list and binary search tree. In this process we had to use non recursive functions to print the tree using preorder traversal, inorder traversal, postorder traversal and breadth first traversal. The time complexity of preorder, inorder, postorder, and breadth first traversal should all be O(n), where n is the total number of nodes in the tree. My algorithm needs to visit every node exactly once so thats why its O(n). As for space complexity it should all be O(n) aswell. The n represents the height of the tree. Preorder traversal, inorder traversal, and postorder traversal all use stack. If we don't consider size of stack for function calls then it would O(1) instead.

	Keys	Searching with Linked List. # of comparisons	Searching with BST. # of comparisons	Searching with BST after tree balancing. # of comparisons
Input file with 10 values [12, 13, 24, 20, 21, 45, 50, 9, 5, 34]	2, 3 ,6	40	13	9
Input file with 15 values [12, 13, 24, 20, 21, 45, 50, 9, 5, 34, 36, 2, 4, 7, 11]	2, 3, 6	49	19	7
Input file with 20 values [12, 13, 24, 20, 21, 45, 50, 9, 5, 34, 36, 2, 4, 7, 11, 15, 16, 22, 23, 30]	2, 3, 6	69	39	7
Input file with 25 values [12, 13, 24, 20, 21, 45, 50, 9, 5, 34, 36, 2, 4, 7, 11, 15, 16, 22, 23, 30, 31, 32, 41, 42, 46]	2, 3, 6	89	59	7

As the input value size increases the number of comparisons increase aswell. The time complexity of searching a linked list should be O(n). The time complexity of searching a binary search tree is O(n) aswell. If the tree is balanced the time complexity should be O(log n). In my balance tree function, I used an insertion sort to sort the linked first first. That has a time complexity of O(n^2). After sorting I build a tree and therefore it is balanced. The balance tree function takes a time complexity of O(log n).

The space complexity for most of my algorithms are O(n). I used a stack and queue for my tree traversals. Each push and pop to the stack require O(1) space complexity. On the other hand, for queue, enqueue requires O(1) and dequeue requires O(n) space complexity.