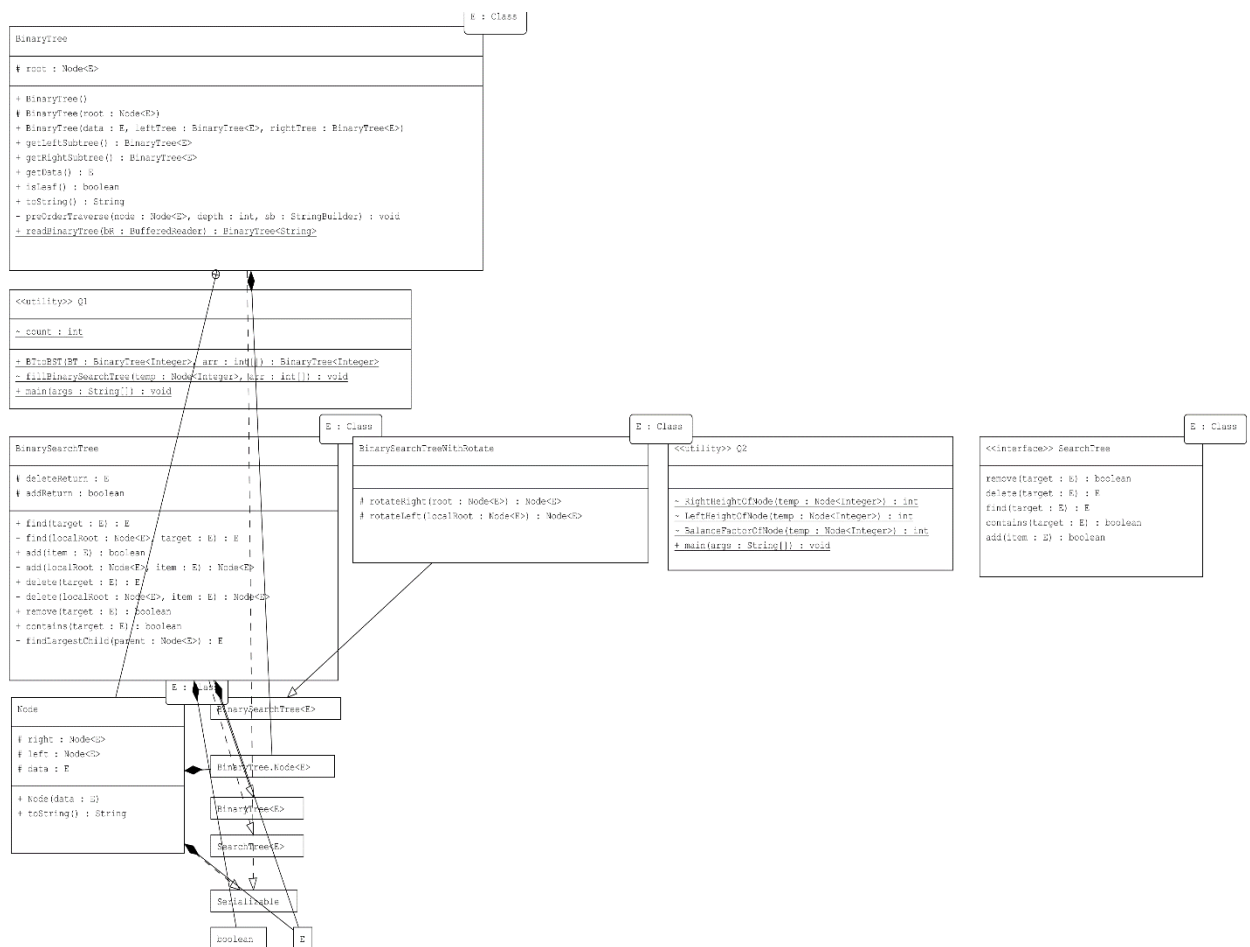


GIT Department of Computer Engineering
CSE 222/505 - Spring 2022
Homework #7 Report

Ufukcan Erdem
1901042686



3. PROBLEM SOLUTION APPROACH

For part 1,

I firstly created an empty binary tree structure and array which has n nodes or items. After that I take the array and sort it. I placed the elements of the sorted array into binary tree nodes with "Inorder Traversal". As a result, I have created a valid Binary Search Tree.

Time complexity of Q1 is;

- 1) `Arrays.sort(arr);` -> $O(n \log n)$
- 2) FillBST -> Inorder traversal -> $O(n)$

So total time complexity $T(n)$ is $O(n \log(n)) * O(n) = O(n^2 \log(n))$

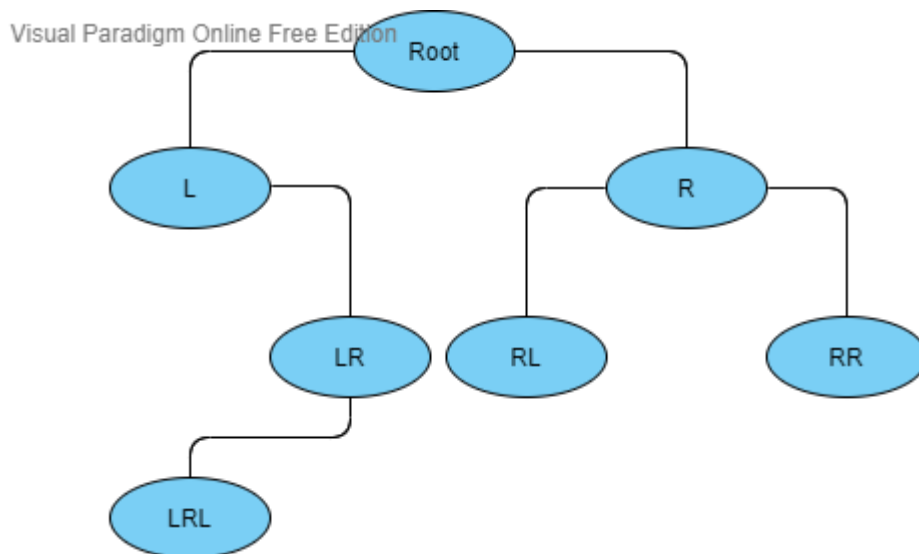
4. TEST CASES

There is two test cases;

---TEST1---

ARRAY -> `{14,4,5,9,11,7,16};`

STRUCTURE OF BINARY TREE;



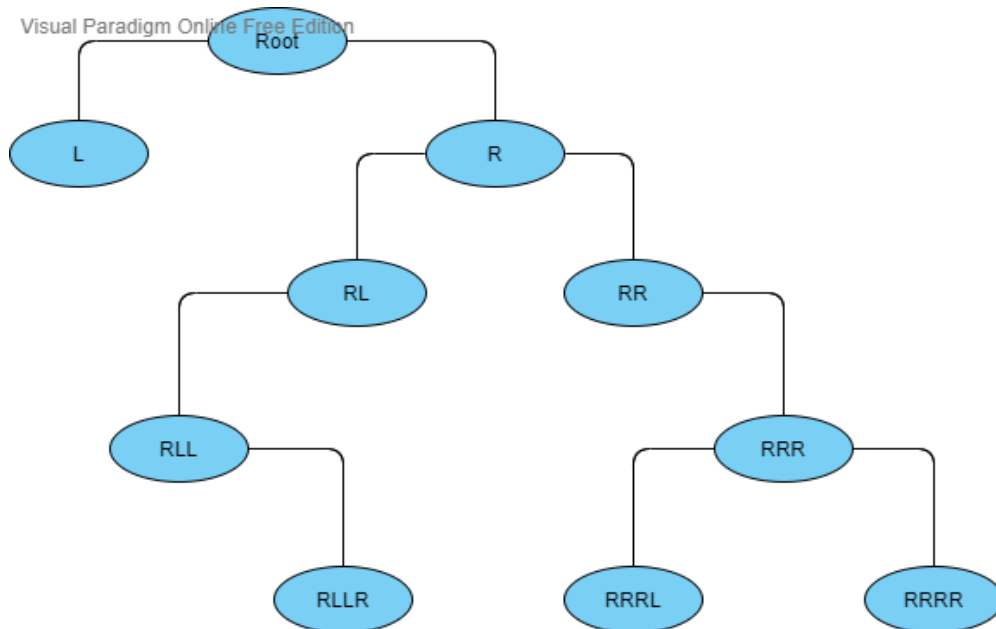
THE STRUCTURE OF EMPTY BINARY TREE FOR TEST1

Visual Paradigm Online Free Edition

---TEST2---

ARRAY -> {13,11,87,2,99,113,25,186,4,12};

STRUCTURE OF BINARY TREE;



THE STRUCTURE OF EMPTY BINARY TREE FOR TEST2

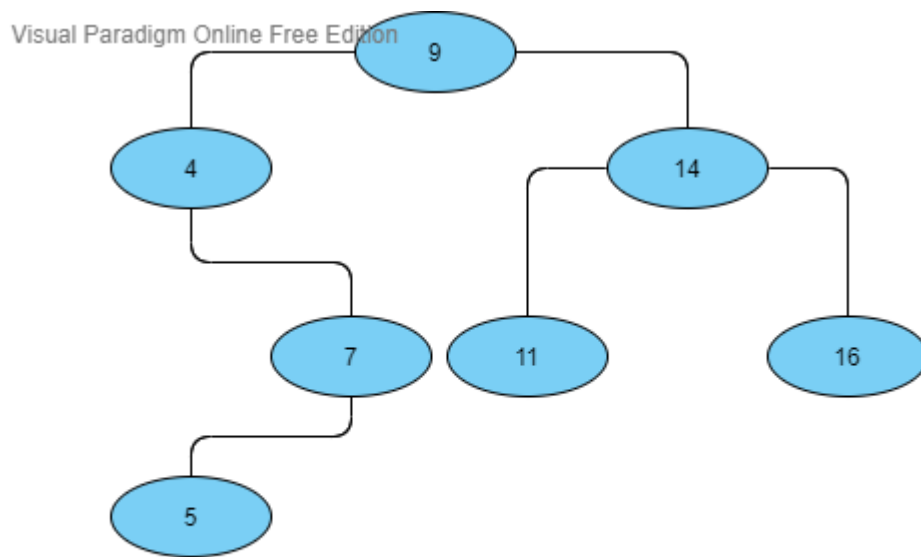
5. RUNNING AND RESULTS

Running Results of Q1;

```
-----TEST1-----
Root->9
Root-left->4
Root-left-right->7
Root-left-right-left->5
Root-right->14
Root-right-right->16
Root-right-left->11
-----TEST2-----
Root->4
Root-right->25
Root-right-right->87
Root-right-right-right->113
Root-right-right-right-right->186
Root-right-right-right-left->99
Root-right-left->13
Root-right-left-left->11
Root-right-left-left-right->12
Root-left->2
```

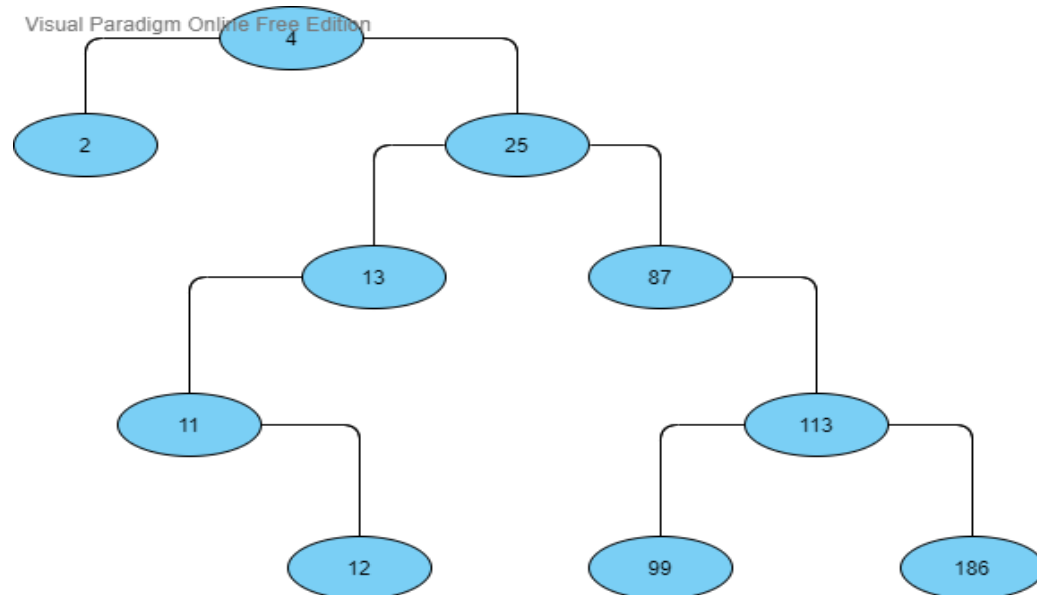
VISUAL RESULTS OF TREES OF Q1;

TEST1:



THE STRUCTURE OF BST FOR TEST1

TEST2:



THE STRUCTURE OF BST FOR TEST2