Programing Assignment 1

Name: Chhay Lay Heng

Class: CS3345.004

1. Screenshot  
   A screenshot of a computer

   Description automatically generated
2. Extra

public class SplayTree {

private Node root;

private static class Node {

int value;

Node left, right, parent;

Node(int value) {

this.value = value;

}

}

// Performs a single rotation on the parent of node

private void zig(Node node) {

Node parent = node.parent;

if (parent != null) {

Node grandParent = parent.parent;

if (grandParent != null) {

if (parent == grandParent.left) {

grandParent.left = node;

} else {

grandParent.right = node;

}

node.parent = grandParent;

} else {

node.parent = null;

root = node;

}

}

if (node == parent.left) {

parent.left = node.right;

if (node.right != null) {

node.right.parent = parent;

}

node.right = parent;

} else {

parent.right = node.left;

if (node.left != null) {

node.left.parent = parent;

}

node.left = parent;

}

parent.parent = node;

}

// Performs two rotations to splay node to the root

private void zigZig(Node node) {

Node parent = node.parent;

Node grandParent = parent.parent;

zig(parent);

zig(node);

}

// Performs two rotations to splay node to the root

private void zigZag(Node node) {

Node parent = node.parent;

Node grandParent = parent.parent;

zig(node);

zig(node);

}

// Performs splaying and searching for node with value k

public Node splayAndSearch(int k) {

if (root == null) {

return null;

}

Node parent = null;

Node grandParent = null;

Node node = root;

while (node != null) {

if (node.value == k) {

break;

} else if (node.value > k) {

if (node.left == null) {

break;

}

parent = node;

node = node.left;

} else {

if (node.right == null) {

break;

}

parent = node;

node = node.right;

}

grandParent = parent;

parent = node;

}

if (node != null) {

if (grandParent == null) {

zig(node);

} else if (parent == grandParent.left && node == parent.left) {

zigZig(node);

} else if (parent == grandParent.right && node == parent.right) {

zigZig(node);

} else if (parent == grandParent.left && node == parent.right) {

zigZag(node);

} else {

zigZag(node);

}

}

return node;

}

}