Nama: M Irfan S

NRP : 171111051

Tugas 7

Graph.java

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */


import java.util.ArrayList;

/**
 *
 * @author yohan
 */
public class Graph {
    ArrayList<GraphNode> nodes;
    ArrayList<GraphEdge> edges;

    /* set this.nodes into new Arraylist<GraphNode>
    set this.edges into new Arraylist<GraphEdge>
    */
    public Graph() {
      this.nodes = new ArrayList<GraphNode>();
      this.edges = new ArrayList<GraphEdge>();

    }

    void add_node(GraphNode new_node) {
        this.nodes.add(new_node);
    }

    void add_edge(GraphEdge new_edge) {
        this.edges.add(new_edge);
    }

    void remove_node(GraphNode deleted_node) {
        this.nodes.remove(deleted_node);
        int i = 0;
        while (i < this.edges.size()) {
            GraphEdge edge = edges.get(i);
            if (edge.src == deleted_node || edge.dst == deleted_node) {
                this.edges.remove(edge);
            } else {
                i++;
            }
        }
    }

    void remove_edge(GraphEdge deleted_edge) {
        this.edges.remove(deleted_edge);
    }

    ArrayList<GraphEdge> get_edges_by_source_node(GraphNode node) {
```

```
        ArrayList<GraphEdge> node_edges = new ArrayList<GraphEdge>();
        for (int i = 0; i < this.edges.size(); i++) {
            GraphEdge edge = this.edges.get(i);
            if (edge.src == node || edge.dst == node) {
                node_edges.add(edge);
            }
        }
        return node_edges;
    }

    GraphNode get_node_by_data(int data) {
        for (int i = 0; i < this.nodes.size(); i++) {
            GraphNode node = this.nodes.get(i);
            if (node.data == data) {
                return node;
            }
        }
        return null;
    }

    Tree to_tree(int root_data) {
        TreeNode first_tree_node = new TreeNode(root_data);
        first_tree_node = this.completing_tree_node(first_tree_node);
        Tree t = new Tree(first_tree_node);
        return t;
    }

    TreeNode completing_tree_node(TreeNode tree_node) {
        int data = tree_node.data;
        GraphNode graph_node = this.get_node_by_data(data);
        ArrayList<GraphEdge> edges = this.get_edges_by_source_node(graph_node);
        for (int i = 0; i < edges.size(); i++) {
            GraphEdge edge = edges.get(i);
            if (edge.src == graph_node) {
                int new_data = edge.dst.data;
                boolean should_add_new_data = true;
                TreeNode current_tree_node = tree_node;
                while (current_tree_node != null) {
                    if (current_tree_node.data == new_data) {
                        should_add_new_data = false;
                        break;
                    }
                    current_tree_node = current_tree_node.parent;
                }
                if (should_add_new_data) {
                    TreeNode new_tree_node = new TreeNode(new_data);
                    tree_node.add_child(new_tree_node, edge.distance);
                            int last_index = tree_node.children.size() - 1;
                            tree_node.children.set(last_index, this.completing_tree_
    node(new_tree_node));
                        }
                    }
                }
                return tree_node;
            }
        }
```

GraphEdge.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
```

```
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */


/**
 *
 * @author yohan
 */
public class GraphEdge {

    GraphNode src;
    GraphNode dst;
    double distance;

    /* set this.src into new_src
    * set this.dst into new_dst
    * set this.distance into new_distance
     */
    public GraphEdge(GraphNode new_src, GraphNode new_dst, double new_distance) {
      this.src = new_src ;
      this.dst = new_dst ;
       this.distance = new_distance;
    }
}
```

GraphNode.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */


/**
 *
 * @author yohan
 */
public class GraphNode {

    int data;

    public GraphNode(int new_data) {
        this.data = new_data;
    }
}
```

Modul7.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
```

```
 * @author yohan
 */
public class Modul7 {

    public static void main(String[] args) {
        Graph g = new Graph();
        GraphNode[] graph_node_list = {
            new GraphNode(0),
            new GraphNode(1),
            new GraphNode(2),
            new GraphNode(3),
            new GraphNode(4),};

        for (GraphNode graph_node : graph_node_list) {
            g.add_node(graph_node);
        }

        int[][] path_list = {{0, 1, 1},
        {0, 2, 1},
        {1, 3, 1},
        {2, 3, 1},
        {3, 4, 2}
        };

        for (int[] path : path_list) {
            GraphNode first_node = graph_node_list[path[0]];
            GraphNode second_node = graph_node_list[path[1]];
            double distance = path[2];
            g.add_edge(new GraphEdge(first_node, second_node, distance));
            g.add_edge(new GraphEdge(second_node, first_node, distance));
        }
        g.to_tree(0).print();
    }

}
```

Tree.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */


/**
 *
 * @author yohan
 */
public class Tree {
    TreeNode root;

    public Tree() {
        this.root = null;
    }

    public Tree(TreeNode root) {
        this.root = root;
    }

    void print() {
```

```
        if (this.root == null) {
            System.out.println();
        } else {
            this.root.print();
        }
    }
}
```

TreeNode.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

import java.util.ArrayList;

/**
 *
 * @author yohans
 */
public class TreeNode {
    TreeNode parent;
    double distance;
    ArrayList<TreeNode> children;
    int data;

    public TreeNode(int new_data) {
        this.data = new_data;
        this.parent = null;
        this.distance = 0.0;
        this.children = new ArrayList<TreeNode>();
    }

    /* set this node's parent into new parent
    * set this node's distance into distance
    * if this node's parent is not nul, then add this as parent's child
    */
    void set_parent(TreeNode new_parent, double distance) {
        this.parent = new_parent;
        this.distance = distance;

        if (this.parent != null) {
          parent.children.add(this);
        }
    }

    // alias to set_parent(new_parent, 0)
    void set_parent(TreeNode new_parent) {
        this.set_parent(new_parent, 0);
    }

    /* call new_child.set_parent. The new parent of new_child should be this
    * The distance of new_child should be set to distance
    */
    void add_child(TreeNode new_child, double distance) {
    new_child.set_parent(this);
```

```
50.     new_child.distance = distance;
51.     }
52.
53.     /* Simply remove child from this node's children */
54.     void remove_child(TreeNode child) {
55.         child.set_parent(this);
56.         distance = child.distance;
57.         this.children.remove(child);
58.     }
59.
60.     /* print this node's data, this node's distance, and distance + this node's dis
    tance
61.     * for each of this node's children, recursively call child's print method
62.     */
63.     void print(String spaces, double distance) {
64.         System.out.println(data+ " distance from parent : "+this.distance+" te Dist
    ance from initial node : "+(distance+this.distance));
65.         for (int i = 0; i < this.children.size(); i++) {
66.         this.children.get(i).print(" " ,this.distance+distance);
67.         }
68.     }
69.
70.     void print() {
71.         this.print(" ", 0);
72.     }
73. }
```

Output