**CS 6301.002 IMPLEMENTATION OF ADVANCED DATA STRUCTURES AND ALGORITHMS**

**Long Project 2 – Minimum Spanning Trees**

**REPORT SUBMITTED BY G05**

Implementation of Level 1

**ABSTRACT**

A minimum spanning tree of an undirected graph connects all the vertices together with the minimal total weighting for its edges. A particular **graph** can have many different spanning trees. Three algorithms for determining the weight of the Minimum Spanning Trees – Prim One, Prim Two and Kruskal’s Algorithm were implemented.

**IMPLEMENTATION**

Three algorithms for determining the Minimum Spanning Tree was implemented which are:

* Prim Take I - was implemented using priority queue of edges
* Prim Take II - was implemented using priority queue of vertices, i.e., using indexed heaps
* Kruskal’s Algorithm

An undirected graph was given as input to the program and the corresponding weight of the Minimum Spanning Tree was displayed.

**DEVELOPMENT PLATFORM**

The project was implemented and tested using the following tools:

**Language:** Java SE 1.7

**IDE:** Eclipse

**TEST RESULTS**

**SAMPLE INPUTS**

Enter number of Nodes:

7

Enter number of Edges:

11

1 2 7

1 4 5

2 3 8

2 4 9

2 5 7

3 5 5

4 5 15

4 6 6

6 5 8

5 7 9

6 7 11

**SAMPLE OUTPUT**

**Prim one:**

Weight of the MST using Prim's Algorithm is:

39

**Prim two:**

Weight of the MST using Prim's Algorithm implemented using indexed Priority Queue is:

39

**Kruskal’s algorithm:**

Weight of the MST using Kruskal's Algorithm is:

39

**CONCLUSION**

We have implemented the three algorithms namely, Prim One, Prim Two and Kruskal’s Algorithm to determine the weight of the Minimum Spanning Tree.

**REFERENCES**

https://en.wikipedia.org/wiki/Prim%27s\_algorithm

https://en.wikipedia.org/wiki/Kruskal%27s\_algorithm