# **Project Report**

Implementation and Evaluation of Graph Theory Algorithms Design and Analysis of Algorithms

**Submitted By**

Abdul Ghaffar Ansari (19k-0287)

**Names and IDs**

Abdul Ghaffar Ansari (19k-0287)

Maryam Hussain (19k-0286)

**Abstract:**

Displaying

directed graph for the seven algorithms: Prims, Kruskal, Dijkstra,

Bellman-Ford, Floyd Warshall Algorithm, clustering coefficients in graph

theory,(only local clustering), and Boruvkas’s algorithm.

**Introduction**

We are implementing 7 algorithms, which are Prims, Kruskal, Dijkstra,

Bellman-Ford, Floyd Warshall Algorithm, clustering coefficients in graph

Theory, (only local clustering), The final cost is the average of all local

Clustering of all nodes, and Boruvkas’s algorithm.

From the seven

Algorithms, the user is given to choose one algorithm. The user is ten asked to

Select which node files they want to plot the graph of. For example, if the user

Select the option of prims algorithm and input10 file, a directed graph for

Prim’s algorithm with 10 nodes will be displayed on the screen. The modules

Used in our project for the plotting of the graph are: import matplotlib.pyplot

As plt, from numpy import \*, from numpy import inf, and import networkx as nx

Graph of x-y

Coordinate with all nodes only, adjacency matrix containing, and full

Graph containing all the specified nodes and their edges will be displayed.

Initial values of the edges and the edges with no weight have value infinity (inf).

MST of the graph is shown in the end, showing minimum spanning tree cost for

The particular algorithm and nodes file.

**Experimental Setup**

The input of our system is the file which input10, input20, input30, input40, input50, input60, input70, input80, input90 and input100. These files are text files containing nodes, the numbered nodes and their weight of the edges.

**Results and Discussion**

In this project, you are required to implement the following algorithms

from Graph Analysis using given benchmarks of increasing number of nodes (from

10 nodes to 100 nodes). Basically, you need to show a very nice user interface

where user can select any input files and then graph to be displayed using x and y

coordinates provided for each node in each input file. Once displayed, then the

the user should be able to run the following algorithms. For Prims, Kruskal & Clustering

Coefficient in Graph Theory, if there is a link between two nodes, then consider this

as an edge in an undirected graph. If there are two directed link b/w edges, then consider

the edge with minimum cost.

Clustering Coefficient in Graph Theory (Only Local Clustering). The final cost

should be the average of all local clustering of all nodes

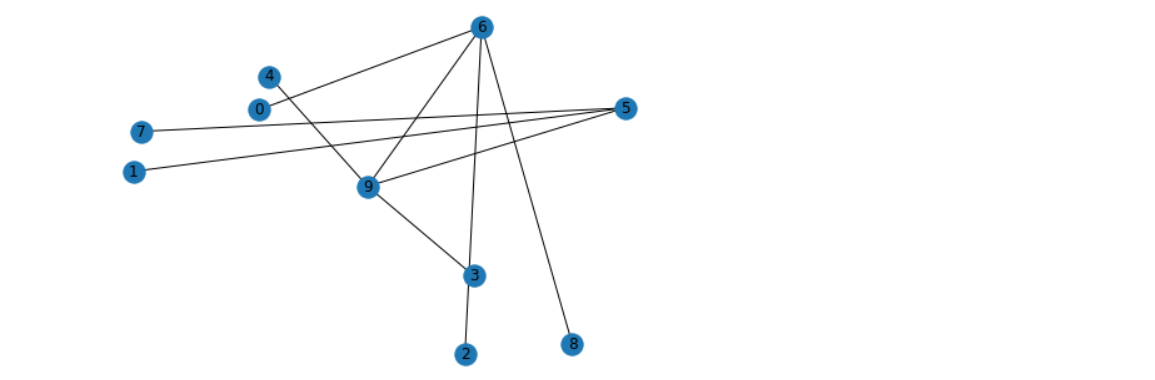
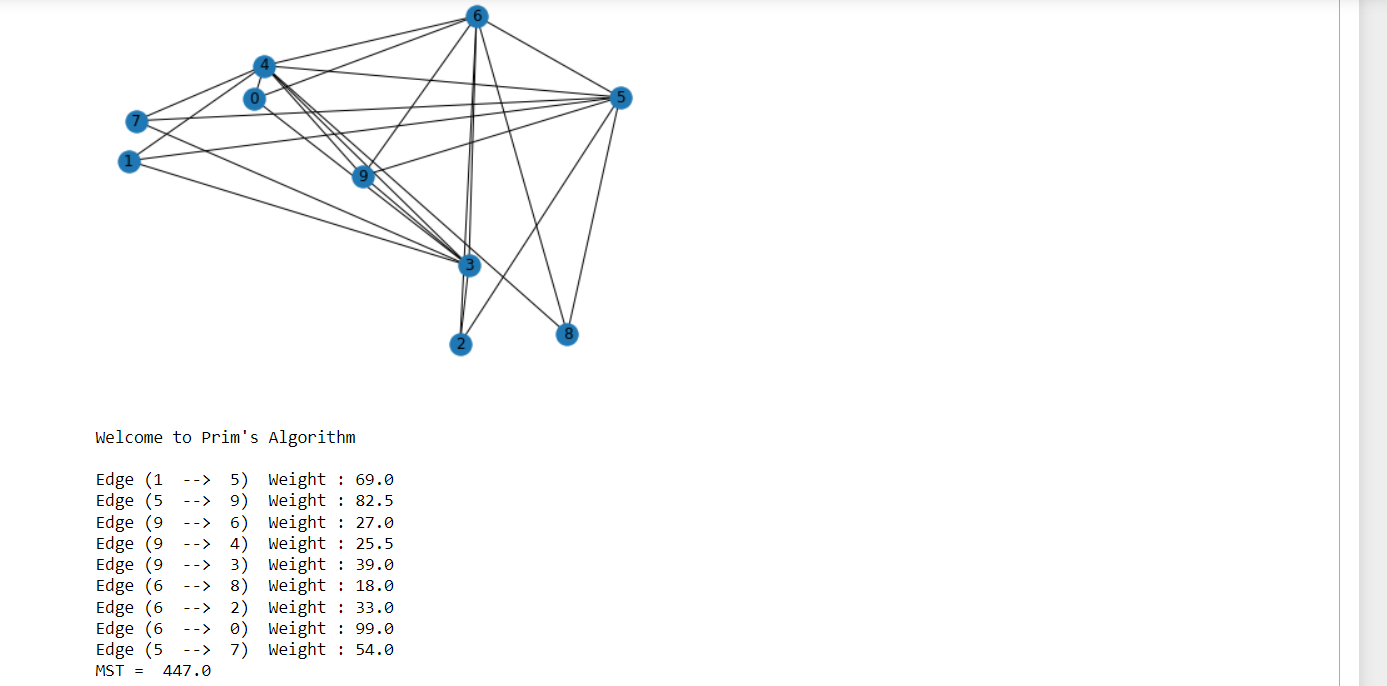
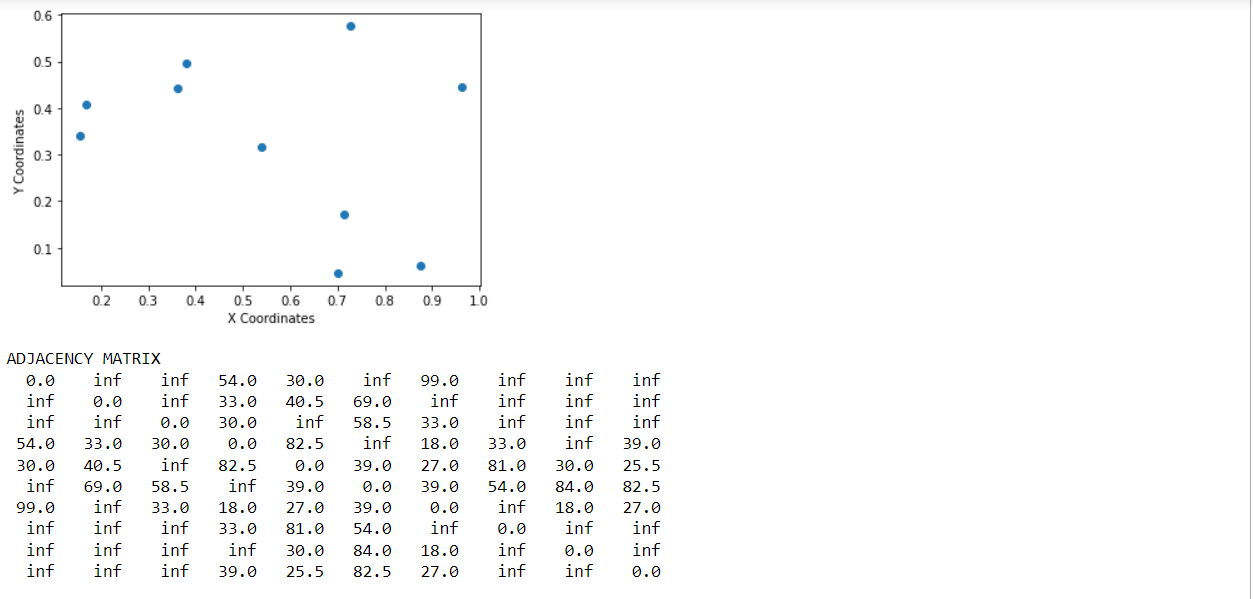
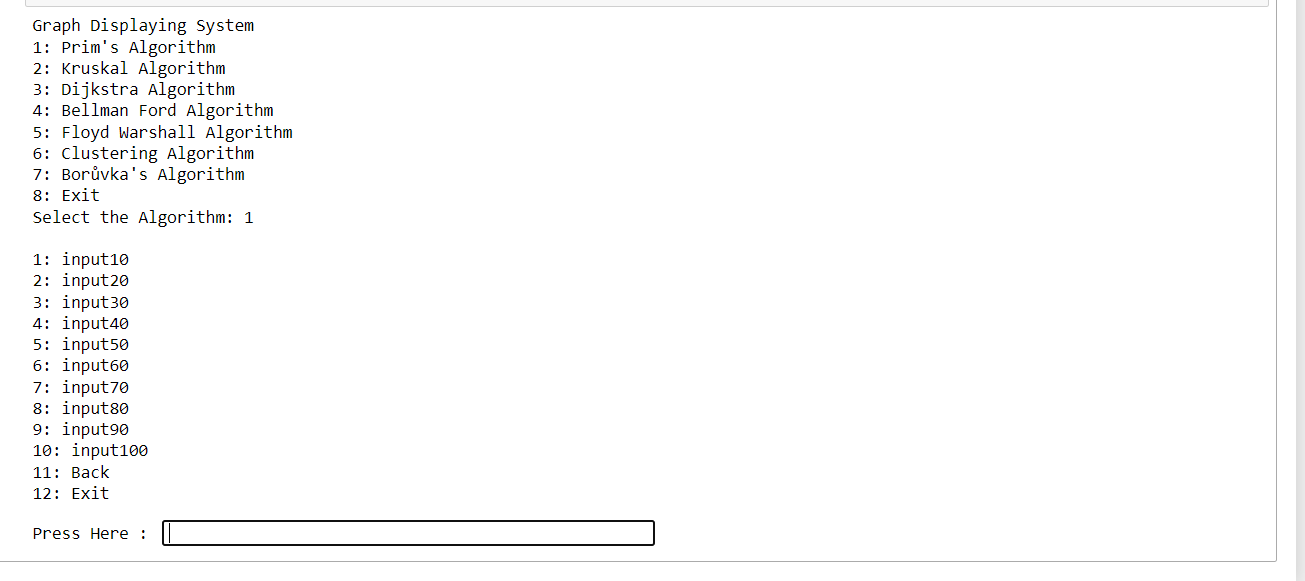
(7) Borůvka's algorithm

Table 1: Give proper caption here

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Benchmark | Prims Total cost in Mbps | Kruskal | Dijkstra (Choose any input node e.g. 5) | Bellman Ford (Choose any input  node e.g. 5) | Floyed Warshall Algorithm | Clustering Coefficient (Local Clustering) | Borůvka's Algorithm |
| Input 10 | 447.0 | 253.5 |  |  |  | 0.6583333333333333 | 253.5 |
| Input 20 | 819.0 | 592.5 |  |  |  | 0.48284685086155665 | 592.5 |
| Input 30 | 1456.5 | 1036.5 |  |  |  | 0.6980946493593552 | 1066.5 |
| Input 40 | 2223.0 | 1519.5 |  |  |  | 0.7709709665090257 | 1519.5 |
| Input 50 | 2688.0 | 1443.0 |  |  |  | 0.6114072786596008 | 1443.0 |
| Input 60 | 3750.0 | 2379.0 |  |  |  | 0.7103363831898404 | 2379.0 |
| Input 70 | 3991.5 | 2278.5 |  |  |  | 0.685452232971337 | 2307.0 |
| Input 80 | 4951.5 | 3072.0 |  |  |  | 0.7008658687965761 | 3072.0 |
| Input 90 | 5226.0 | 3343.5 |  |  |  | 0.7858374185184406 | 3343.5 |
| Input 100 | 5662.5 | 3513.0 |  |  |  | 0.699423195351583 | 3513.0 |

**SCREENSHOTS**

Some Screenshots of prim’s Algorithm

****

**Conclusion**

Graph containing all the specified nodes and their edges will be displayed.

Initial values of the edges and the edges with no weight have value infinity (inf).

MST of the graph is shown in the end, showing minimum spanning tree cost for

The particular algorithm and nodes file.

**References**

[www.programiz.com](https://www.programiz.com)

[www.geeksforgeeks.org](http://www.geeksforgeeks.org)

[www.github.com](http://www.github.com)

[www.youtube.com](http://www.youtube.com)

Algorithm Theory Slides

1. Anany V. Levitin. 2006. *Introduction to the Design and Analysis of Algorithms (2nd Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Appendix (If needed) this can be outside 4 pages