Documentation
of
Minimum Spanning Tree Using Prims and Kruskal Algorithm

By ROHAN CHOPRA

This has been project created as a Programming project submission for CSE -5311-001 Design and analysis of algorithm. From the given project choices, I have chosen to implement and compare Minimum Spanning Trees Algorithm. This is a GUI python-based application that allows a user to perform create a Minimum Spanning Tree using either prims or Kruskal algorithm.

Tools used:

- Python(3.7)
- PyCharm Community
- Tkinter python framework for GUI

Features of this program:

- Calculate a minimum spanning tree using Prim's algorithm for a text file selected by the user.
- Calculate a minimum spanning tree using Kruskal's algorithm for a text file selected by the user.
- Assists the user in comparing and deciding between Prim's algorithm and Kruskal's algorithm depending upon the number of vertices and edges.
- A GUI driven program .

Files Included:
- **DataStructures.py**: This file contains the classes and methods used to implement the data structures used in both the algorithms.
- **PrimsAlgo.py:** This python file contains the implementation of prims algorithm.
- **KruskalAlgo.py:** This python file contains the implementation of Kruskal algorithm.
- **View.py**: This python file uses all the above files and links it with view or GUI components of the program.
- **Run.py:** This python file is used to run the project.
- **dataSet1.text :** First data set.
- **dataSet2.text :** Second data set.
- **random.py :** Generate random graph data.

How To Run:
1. Extract the files in a folder, make sure everything is the same directory.
2. Install Python on your computer (ver 3.x) is needed.
3. You can use any python IDE or command line to run "Run.py" file .
4. You need to create a dataset file , you can follow the instructions below to create one.

How to create a data set:
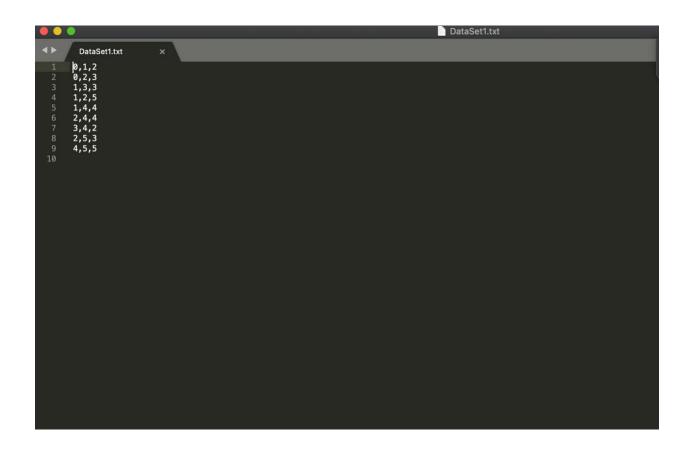1. Create a new text file , you can name it anything you like.
2. Every line in your text file is an edge.
3. The format of each line should be –
   StartVertex,EndVertex,Weight
4. No need to enter any number of edges or vertices.
5. Make sure there is no mistakes in the file.

You can use random.py file to create a random list of values and the format it according to the requirement.

```
0,1,2
0,2,3
1,3,3
1,2,5
1,4,4
2,4,4
3,4,2
2,5,3
4,5,5
```

Screenshots

Prims Algorithm

Please select a text file with graph data: Select File

No file selected

GO

Or

Back

Created by Rohan Chopra

Compare

Enter number of vertices:

Enter number of Edges:

Kruskal= O(|E*LogV|)   Prims= O(|V*V|)

Go

Back

Created by Rohan Chopra

Kruskal Algorithm

Please select a text file with gra

Graph extracted wit

GO

Or

Back

Minimum Spanning Tree is-

[V1,V2,Weight]
[0, 1, 2]
[3, 4, 2]
[0, 2, 3]
[1, 3, 3]
[2, 5, 3]
time :0.0799ms

OK

your website.

zing value

Created by Rohan Chopra

Compare

Enter number of vertices: 10

Enter number of Edges: 20

Kruskal= O(|E*LogV|)   Prims= O(|V*V|)

Go

Back

You should use

You should use Kruskal. Since graph is sparse.

OK