

1. How to Run My Java Program in Eclipse

Put all my java file into eclipse and run the main class

Input the path of the graph.in.txt file

Then you can find two documents named Kruskal.out and Prim.out have already created

in the path of your input file.

2. How to Run MST.jar

1. Click Start and type cmd.

2. Type the path of the MST.jar (If your put the MST.jar into desktop, you need to type cd desktop first and then type java -jar MST.jar)

3. Type the path of the graph.in.txt (For example:
C:\\Users\\host\\Desktop\\graph.in.txt)

4. When it appears Finished. That means the application run successful. It will generate two Output files. You can check those two files.

3. How to Make My Program

I have already write the comment in my java file. And it will tell you how to make my program step by step. The word below is my thought about this project.

For Prim class:

Get the first vertex, mark it as visited. find all adjacent edges
and put them into a heap. Pop out the smallest edge and
then mark another side of vertex
as visited. Then find all adjacent edges and put them
into the same heap. Then do what the
first vertex does recursively. Until the MST size is equal
to total vertex number -1.

For Kruskal class:

Find: find a's parent (find root node) if a's parent is
itself, a is a root.

Make union: make a's root as b.

Kruskal: insert all edges into heap. Extract the
minimum edge
from heap and insert
it into the MST. If e.from and e.to is not in the same set
then union those two.

For MinHeap class: it includes swap, minheapify (if
parent is greater than its children, swap
The smaller one to upside.)

and popout (which extract min edges).

For readfile class: use hashmap to traverse vertex from a,b,c to 1,2,3.

(It can also traverse any strange signal into 1,2,3). For read edges, I use split to get all information and Store those information into edges.

For Vertex class: it includes visited and adjedges.

For Edge class: it includes from, to, and weight. It also include compareTo.