CS 5200 Homework - 4

Instructor Avah Banerjee

April 17, 2024, 10 PM CST

Problem 1 (30+30+40 pts) Given a connected edge-weighted graph $G = (V, E, w)$, let $MST(G)$ be the set of all minimum spanning trees of G. Develop an algorithm (30 pts) that, given G, can find two minimum spanning trees T_1, T_2 in $MST(G)$ (if G has more than one minimum spanning tree) such that $|E(T_1) \cap E(T_2)|$ is minimized. That is, T_1 and T_2 are the pair of MSTs which share the least number of edges between them. Show that your algorithm is correct and identify whether your algorithm is linear time, polynomial time, or exponential time in the size of the input $(|V| = n, |E| = m)$. Justify your answer (30 pts). Implement this algorithm using Python, which takes G in the form of a weighted adjacency matrix A, such that $A_{u,v} = w(u, v)$, and returns a pair of MSTs T_1, T_2 as above, or returns that no such pair exists. Your code should take as input a `.csv` file containing the adjacency matrix as shown below, where the vertices are numbered from 0 to $n - 1$. In the above example, the weight between the pair of vertices $\{1, 3\}$ is 3 (40 pts).

Example file (*input.csv*):

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

Your output should graphically print out the two trees found, as well as the original input graph. You may use language model tools to look up simple APIs and code snippets for creating the helper functions.