Marcus Anderson

Homework 5

CS 6515: Introduction to Graduate Algorithms

Approach:

To find if an input edge $e = (u, v)$ is a part of some MST of graph G we can utilize the cycle property of MST. First, we make a copy of graph G, remove all edges that have a greater than or equal to weight as input edge e, and call it G^e. We then run G^e through the Explore subroutine starting at vertex u and check if *visited[v]* equals true or false. If *visited[v]* is true, then e is not a part of some MST, and we return FALSE. However, if *visited[v]* is false, e is a part of some MST, and we return TRUE.

Correctness:

Using the MST cycle property, we can see if edge e cannot be a part of any MST of G by letting e be the unique heaviest edge on a cycle of G. Creating graph G^e fulfills this condition as e will be the heaviest weighted edge within this graph. Then, utilizing the Explore subroutine starting at vertex u, tells us if vertex v is reachable from vertex u or not by checking the if *visted[v]* is true or false. If it's reachable, there's a cycle in G^e with a maximum weighted edge and therefore cannot be a part of some MST.

Runtime:

Copying the original graph takes O(n + m) time. Removing edges with a greater weight than the input edge takes O(m) time. Finally, running the Explore subroutine as a black box on the modified graph takes O(n + m) time. Making the overall runtime O(n + m).

References:

- https://en.wikipedia.org/wiki/Minimum_spanning_tree#Cycle_property

Collaborators:

- Lilley, Zachary J: zlilley3@gatech.edu
- Bertrand, James M: jbertrand9@gatech.edu
- Ramasamy, Veerajothi: vramasamy9@gatech.edu
- Acker, Joshua R: jacker7@gatech.edu
- Shah, Jeet Hemant: jshah328@gatech.edu