Name: Michael Rogers

ID: 105667404

CSCI 3104, Algorithms   **Profs. Hoenigman & Agrawal**
Problem Set 5a (11 points)   **Fall 2019, CU-Boulder**

Instructions for submitting your solution:

- The solutions **should be typed** and we cannot accept hand-written solutions. Here's a short intro to Latex.

- You should submit your work through **Gradescope** only.

- If you don't have an account on it, sign up for one using your CU email. You should have gotten an email to sign up. If your name based CU email doesn't work, try the iden-tikey@colorado.edu version.

- Gradescope will only accept **.pdf** files (except for code files that should be submitted sep-arately on Gradescope if a problem set has them) and **try to fit your work in the box provided**.

- You cannot submit a pdf which has less pages than what we provided you as Gradescope won't allow it.

- Verbal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

- For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

- You may work with other students. However, **all solutions must be written indepen-dently and in your own words.** Referencing solutions of any sort is strictly prohibited. You must explicitly cite any sources, as well as any collaborators.
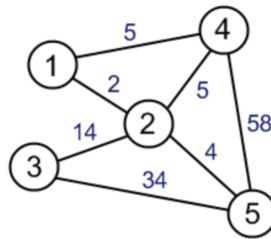
CSCI 3104, Algorithms **Profs. Hoenigman & Agrawal**
Problem Set 5a (11 points) **Fall 2019, CU-Boulder**

1. (7 pts) Consider the following weighted graph.

(a) (2 pts) Run Dijkstra's algorithm on this graph to obtain a tree of shortest paths. Use vertex 1 as the source vertex.

Solution. Table of Solved Verticies

Verticy	Shortest Path from 1
1	0
2	2
4	5
5	6
3	16

Selected Edges = {(1,2), (2,5), (1,4), (2,3)}

CSCI 3104, Algorithms   **Profs. Hoenigman & Agrawal**

Problem Set 5a (11 points)   **Fall 2019, CU-Boulder**

(b) (2 pts) Run Kruskal's algorithm on this graph to obtain a minimum spanning tree.

Solution. Table of Selected Edges for MST

Edge	Select/Disregard
(1,2)	Select
(2,5)	Select
(2,4)	Select
(1,4)	Diregard
(2,3)	Select
(3,5)	Disregard
(4,5)	Disregard

Selected Edges={(1,2), (2,5), (2,4), (2,3)}

(c) (1 pts) Is the tree of shortest paths produced by Dijkstra's algorithm a minimum spanning tree? Justify your answer.

Solution. Yes, the tree of shortest path produced by Dijsktra's Algorithm is also a minimum spanning tree of the graph. If we look at the MST produced by kruskals, the overall weight is 25. The overall weight of the tree produced by Dijsktra's also has a weight of 25.

(d) (2 pts) Find two vertices u and v, where the $u - v$ path in the Kruskal tree is not a shortest $u - v$ path.

Solution. Kruskal's selected (2,4) when Dijkstra's selected (1,4). The path from 1 to 4 in Dijkstra's is equal to 5 while in Kruskal's it is equal to 7.

Name: Michael Rogers

ID: 105667404

CSCI 3104, Algorithms   **Profs. Hoenigman & Agrawal**
**Problem Set 5a (11 points)**   **Fall 2019, CU-Boulder**

2. (1 pt) Provide a brief description of what the *find(v)* and *union(A,B)* features of the union-find algorithm produce.

   Solution. find(v) - The *find(v)* function returns the name of the leader of the cluster that contains v when working with disjoint sets with Union Find.

   union(A,B) - The *union(A,B)* function is used in Union Find and is used to fuse 2 clusters, A and B, together to create one when working with disjoint sets.

3. (3 pts) Identify three edges in the following graph G that won't be included in any MST of G. Provide a 3-4 sentence explanation of your answer.
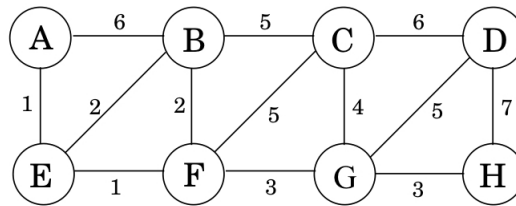
   Solution. NoMST = {(D,H), (C,D), (A,B)}

   These 3 edges cannot be apart of the MST because of the cut property. There is no cut that you can make that will make (D,H), (C,D), or (A,B) the shortest path going into any of the verticies B,C,D. This is one of the properties that makes Kruskal's algorithm optimal, so if that property is broken, then there's no way any MST can contain any of these edges.