Spring 2023 CSDS 410 Analysis of Algorithms
Prof. Dianne Foreback Assignment 5

Due Sunday, Nov 19th before 11:59 p.m. in Canvas
Possible 100 points

Directions:
All work should be your own. You may not copy information from website or use AI tools. This is considered a violation of academic integrity, see https://case.edu/gradstudies/current-students/policies-procedures. If this is violated, you will, at minimum, receive a score of zero on the assignment, it will NOT be dropped as your lowest grade out of the remaining graded assignments, you will be reported per the policy and possible receive a failing grade in the class.

Please upload your Assignment 5 into Canvas and make certain that the quality of the upload is clear. Also, enumerate your answers; e.g. before your answer for problem 1, preceded it with 1a, 1b, 1c, etc.

Please double check that your assignment is properly submitted into Canvas and is visible. Since file dates can be modified, and out of fairness to all, assignments not uploaded into Canvas by the final due date will not be accepted. That is, if your file is on a Google Drive but you forgot to place it into Canvas, it will not be accepted even if the file date shows an acceptable modify date.

Note that this assignment **may** be handwritten providing your writing is easily read.

Assignment Material:
This assignment covers Minimum Spanning Trees and Shortest Paths.

Submit your work on canvas. Always check canvas for updates and corrections. Unless otherwise stated, whenever a question asks you to **describe an algorithm**, you should:

a. Explain the main concept of your algorithm.
b. Give pseudo-code.
c. Present an example of running your algorithm.
d. Prove/justify its correctness and its running time.

Please label your problem answers with 1a. 1b. 1c. 1d as appropriate.

If asked to prove an algorithm is incorrect, this can be done by giving an example where the algorithm does not provide a correct solution.

Point values for a problem are placed in brackets next to the problem number; e.g. 1 [8] means problem 1 is worth 8 points.

Problems:

1. [8] Draw a simple, connected, undirected, weighted graph with 8 vertices and 16 edges, each with unique edge weights. Illustrate the execution of Kruskal's MST algorithm on this graph. (Note that there is only one minimum spanning tree for this graph.)

2. [8] Why does Kruskal's MST algorithm still work correctly even if the graph has negative-weight edges, and even negative-weight cycles?

3. [8] Suppose G is a weighted, connected, undirected, simple graph and e is a largest weight edge in G. Prove or disprove the claim that there is no minimum spanning tree of G that contains e.

4. [8] Suppose G is an undirected, connected, weighted graph such that the edges in G have distinct edge weights. Show that the minimum spanning tree for G is unique.

5. [8] Suppose *G* is an undirected, connected, weighted graph such that the edges in *G* have distinct positive edge weights. Prove that the minimum spanning tree for *G* is unchanged even if we square all the edge weights in *G*, that is, *G* has the same set of edges in its minimum spanning tree even if we replace the weight, *w(e),* of each edge e in *G*, with $w(e)^2$.

6. [8] Suppose *G* is an undirected, connected, weighted graph such that the edges in *G* have distinct edge weights, which may be positive or negative. Prove that the minimum spanning tree for *G* may be changed if we square all the edge weights in *G*, that is, *G* may have a different set of edges in its minimum spanning tree if we replace the weight, *w(e),* of each edge e in *G*, with $w(e)^2$.

7. [8] Suppose Joseph Kruskal had an evil twin, named Peter, who designed an algorithm that takes the exact opposite approach from his brother's algorithm for finding an MST in an undirected, weighted, connected graph, G. Also, for the sake of simplicity, suppose the edges of G have distinct weights. Peter's algorithm is as follows: consider the edges of G by decreasing weights. For each edge, e, in this order, if the removal of e from G would not disconnect G, then remove e from G. Peter claims that the graph that remains at the end of his algorithm is a minimum spanning tree. Prove or disprove Peter's claim.

8. [8] Draw a simple, connected, weighted, undirected graph with 8 vertices and 16 edges, and with distinct edge weights. Identify one vertex as a "start" vertex and illustrate a running of Dijkstra's algorithm on this graph.

9. [20] **Describe a modification** to Dijkstra's algorithm for the case when the graph is directed and we want to compute shortest directed paths from the source vertex to all the other vertices. Also, this algorithm, in addition to output the distance from v to each vertex in G (as in the original Dijkstra's algorithm), it should also to output a tree T rooted at v, such that the path in T from v to a vertex u is actually a shortest path in G from v to u. Please see "Assignment Material" section for parts a-d on "**describing an algorithm**".

10. [16] Problem on Dijkstra's algorithm with negative edge weights.
a. Prove that Dijkstra's algorithm will not work for an undirected graph that has a negative edge weight.
b. Prove that Dijkstra's algorithm will not work for a directed graph that has a negative edge weight even if there is not a negative edge weight cycle.