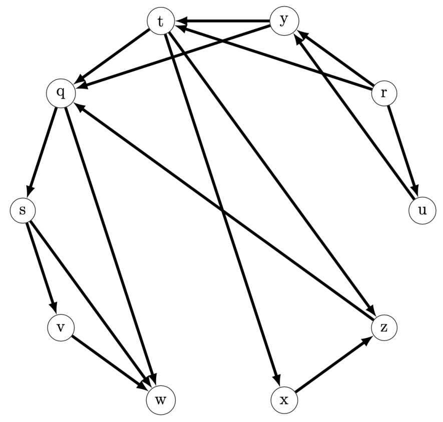
CS251 - Data Structures and Algorithms Fall 2024

PSO 9, Week 11

Question 1

(Topological Ordering)

1. Draw a directed acyclic graph $G = (V, E)$ with $|V| = 5$ that has exactly two topological orderings.
2. Execute the topological sort algorithm on the following directed graph.

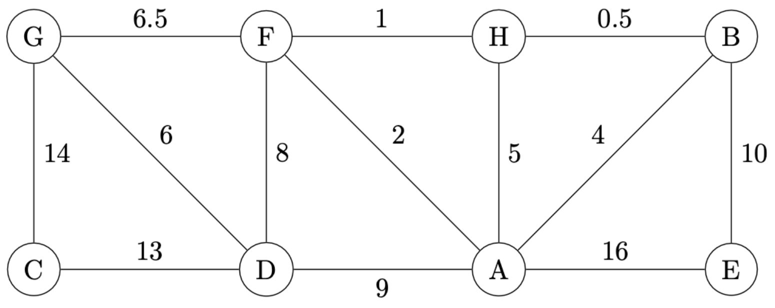


3. (optional) Prove that G has a topological ordering if and only if G is a DAG.

Question 2

(Dijkstra's algorithm)

1. List the order that edges will be added if we run the Dijkstra's algorithm starting at node A in the following graph.



2. Give a simple example of a directed graph with negative-weighted edges for which Dijkstra's algorithm produces an incorrect answer.

3. Given a weighted, directed graph $G = (V, E)$ and a source vertex $s \in V$, is it true that the Dijkstra's algorithm can never correctly find the shortest paths starting from s if G has negative-weighted edges?

Question 3

(More on the Dijkstra's algorithm)

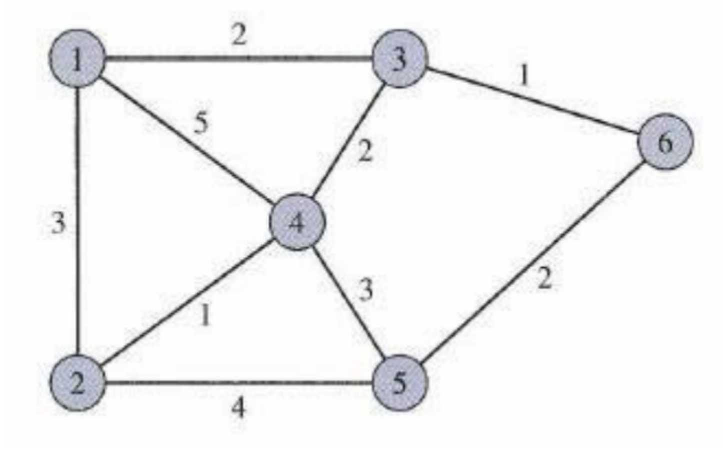
Suppose you are a city planner, and you are worried that it takes too long for an ambulance to get from the university to the hospital. You've decided to build one new street to improve this particular route. You and your team have researched and gathered a list of potential streets that could be built; the task now is to choose one street that will improve this route the most.

To model this formally, let $G = (V, E)$ be a directed graph with positive edge weights (representing the lengths of streets), and let $s, t \in V$ be two vertices in the graph. We assume that E represents the edges/streets that are"already built". Let $E^{'}$ represents the edges/streets that you can add to the graph. Then the problem is to identify the single edge $e \in E^{'}$ so that, when you add e to G, the distance from s to t in the augmented graph is as short as possible. Design an algorithm for this problem and and analyze its time complexity.

Question 4

(Bellman-Ford algorithm)

1. Execute the Bellman-Ford algorithm on the following weighted, undirected graph.



2. Can you briefly explain why $(|V| - 1)$ number of iterations in the Bellman-Ford algorithm is enough to determine if there are negative weight cycles?