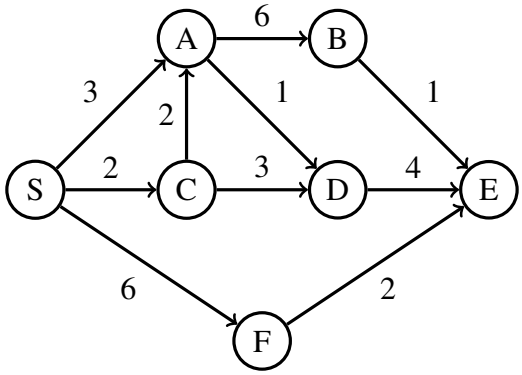
1 Q1: Shortest Path Algorithms

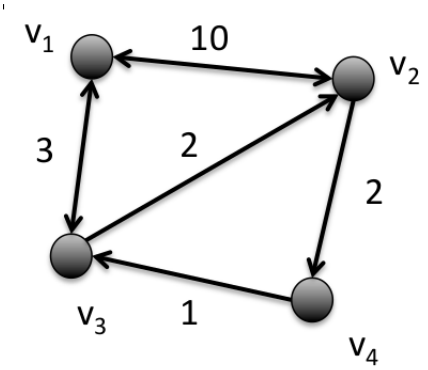
Q1.1 Consider a weighted directed graph $G = (V, E, w)$ and let X be a shortest $s - t$ path for $s, t \in V$. If we double the weight of every edge in the graph, setting $w'(e) = 2 \cdot w(e)$ for each $e \in E$, then will X be still a shortest $s - t$ path in G? Explain your answer. (**2 marks**)

Q1.2 Run Dijkstra's algorithm on the following directed graph, starting at vertex S. What is the order in which vertices get removed from the priority queue? What is the resulting shortest-path tree? (**3 marks**)



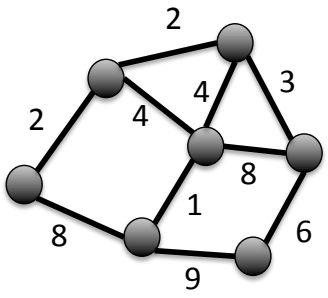
Q1.3 The time complexity Floyd-Warshalls algorithm is $O(n^3)$. Is this ever better than simply running the Dijkstra's single-source shortest path algorithm from every node? Note that the best time complexity for Dijkstra's single-source shortest path is $O(m + n \log n)$ when a Fibonacci heap are used to store distances). Briefly explain your answer. Hint - you will get most marks for your explanation. (**2 marks**)

Q1.4 Solve the all-shortest-paths problem for the following graph using Floyd-Warshall's algorithm. (**3 marks**)



2 Q2: Minimum Spanning Trees and P vs NP

Q2.1 Apply Kruskal's algorithm to the following graph



In your answer show which edges are added and skipped and draw the final spanning tree. (**3 marks**)
Q2.2 Describe the role of the Union–Find data structure in Kruskal's Algorithm. (**1 marks**)
Q2.3 Briefly describe how the *find* operation is implemented in the Union–Find data structure used in Kruskal's algorithm. (**1 marks**)
Q2.4 What does it mean to say that a problem is NP? Give an example of a problem that is NP. (**2 marks**)
Q2.5 Briefly describe how a problem can be shown to be NP-Hard. (**2 marks**)
Q2.6 Is the minimum spanning tree problem in P? Please explain your answer. (**1 marks**)