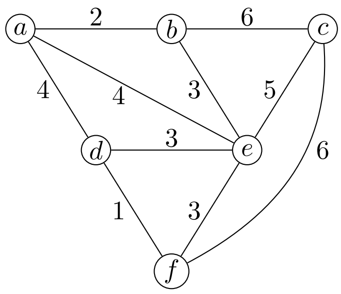
Problem 1 (20 points). Run Kruskal's algorithm on the following undirected graph: give the order of edges that are added to the MST (whenever you have a choice, always choose the smallest edge in lexicographic order); for each edge added, give a cut (i.e., the certificate) that justifies its addition does not break optimality.



Problem 2 (20 points). Run Prim's algorithm on the above undirected graph: give the order of vertices that are added to the MST (whenever you have a choice, always choose the smallest vertex in alphabetic order); before adding each vertex to the MST, give the *key* (i.e., priority) value for all vertices in the priority queue.

Problem 3 (20 points). Design an efficient algorithm for the *maximum spanning tree* problem, i.e., given an undirected graph $G = (V, E)$ with edge weight $w(e)$ for any $e \in E$, to compute a spanning tree T of G such that $\sum_{e \in T} w(e)$ is maximized.

Problem 4 (20 points). Give a counter-example or prove the following statement: Let $G = (V, E)$ be an undirected graph. Let C be one cycle in G and let e be an edge in C. If the weight of e is strictly larger than any other edge in C, then e is not in any minimum spanning tree of G.

Problem 5 (20 points). Give a counter-example or prove the following statement: Let $G = (V, E)$ be an undirected graph with edge weight $w(e)$ for any $e \in E$. Let T be an MST of G. Let X be a connected subgraph of G. Then $T \cap X$ is contained in some MST of X.