1. (10 points) Recall Johnson’s algorithm for the APSP problem.

(a) Suppose that the weight function is defined as \( \hat{w} = w(u, v) - \min_{(u,v) \in E} \{w(u, v)\} \). Does this modification yield a correct APSP algorithm? Prove your claim.

(b) Suppose we modify Johnson’s algorithms as follows: do not create a new source vertex but use \( G' = G \) and let \( s = v \) where \( v \) is any vertex in \( V(G) \). Does this modification yield a correct APSP algorithm? Prove your claim.

2. (10 points) The questions below are about flow networks.

(a) Show that splitting an edge in a flow network as described in class yields an equivalent network. More formally, suppose that flow network \( G \) contains edge \((u, v)\) and we create a new flow network \( G' \) by creating a new vertex \( x \) and replacing \((u, v)\) by new edges \((u, x)\) and \((x, v)\) with \( c(u, x) = c(x, v) = c(u, v) \). Show that a maximum flow in \( G' \) has the same value as a maximum flow in \( G \).

(b) Suppose that a flow network \( G = (V, E) \) violates the assumption that the network contains a path \( s \leadsto v \leadsto t \) for all vertices \( v \in V \). Let \( u \) be a vertex for which there is no path \( s \leadsto u \leadsto t \). Show that there must exist a maximum flow \( f \) in \( G \) such that \( f(u, v) = f(v, u) = 0 \) for all vertices \( v \in V \).

3. (10 points) Show that the maximum flow in a network \( G = (V, E) \) can be found by a sequence of at most \(|E|\) augmenting paths. (Note that this question doesn’t ask you to find these paths, but to argue that such paths exist)

4. (20 points) Let \( G = (V, E) \) be a directed graph with source \( s \) and sink \( t \) and with edge capacities given by \( c_e \geq 0 \) for all \( e \in E \). Let \( f \) be a maximum flow defined by \( f_e \) for all edges \( e \in E \).

(a) Suppose we increase the capacity of a single edge \( e \) from \( c_e \) to \( c_e + 1 \). Show how to find a maximum flow in the new graph in time \( O(|V| + |E|) \).

(b) Suppose we decrease the capacity of a single edge \( e \) from \( c_e \) to \( c_e - 1 \). Show how to find a maximum flow in the new graph in time \( O(|V| + |E|) \).

5. (10 points) Define edge-connectivity of an undirected graph to be the minimum number \( k \) of edges that need to be removed in order to disconnect the graph (trees have \( k = 1 \), cycles have \( k = 2 \), a complete graph on \( n \) vertices has \( k = n - 1 \)). Using a max-flow approach, design and analyze an algorithm to determine the edge connectivity of a given graph \( G = (V, E) \).

6. (20 points) Suppose we have a flow network with multiple sources and sinks. In particular, let \( S \) be the set of sources and let \( T \) be the set of sinks. Instead of maximizing the flow value, we would like to consider the problem where sources have fixed supply values and sinks have fixed demand values and our goal is to ship flow from nodes with available supply to those with given demands (think about factories with supply and stores with demand). Instead of maximizing particular values, we simply want to satisfy all the demand using the available supply.
Formally, we are given a graph \( G = (V, E) \) with capacities on the edges. Associated with each node \( v \in V \) is a demand \( d_v \). If \( d_v > 0 \), this indicates that node \( v \) has demand of \( d_v \) for flow; the node is a sink and wishes to receive \( d_v \) units more flow than it sends out. If \( d_v < 0 \), this indicates that \( v \) has supply of \(-d_v\); the node is a source and it wishes to send out \(-d_v\) units more flow that it receives. If \( d_v = 0 \), then node \( v \) is neither a source, nor a sink. Further, assume that all capacities and demands are integers.

Define a circulation with demands \( \{d_v\} \) to be a function \( f \) that assigns a non-negative real number to each edge and satisfies the following two conditions:

i. \( \forall e \in E : 0 \leq f(e) \leq c_e \) (capacity constraint)

ii. \( \forall v \in V : f^{in}(v) - f^{out}(v) = d_v \) (demand constraint)

The problem we want to solve is the circulation feasibility problem: does there exist a circulation that meets the above conditions?

(a) Prove that if there exists a feasible circulation with demands \( \{d_v\} \) then \( \sum_{v \in V} d_v = 0 \)

(b) Show how to reduce the problem of finding a feasible circulation with demands \( \{d_v\} \) to the problem of finding a maximum \( s-t \) flow in a different flow network. (What is the flow in this “other” network?)

7. (20 points) We now modify the circulation problem to add lower bounds as follows: We are given a graph \( G = (V, E) \) with capacities \( c_e \) and lower bounds \( l_e \) on the edges, where \( 0 \leq l_e \leq c_e \) for all edges \( e \in E \). Associated with each node \( v \in V \) is a demand \( d_v \) as before. Define a circulation with lower bounds with demands \( \{d_v\} \) to be a function \( f \) that assigns a non-negative real number to each edge and satisfies the following two conditions:

i. \( \forall e \in E : l_e \leq f(e) \leq c_e \) (capacity constraint)

ii. \( \forall v \in V : f^{in}(v) - f^{out}(v) = d_v \) (demand constraint)

As in the problem above, we want to find out whether there exist a feasible circulation that meets the above conditions.

(a) Show how to reduce the problem of feasible circulation with lower bounds to the problem of feasible circulation without lower bounds.

(b) Using the idea from part (a), design and analyze an algorithm for the following problem: KrispyKream would like to conduct a survey, sending customized questionnaires to a group of \( n \) customers, to try determining which donuts people like best. Here are the parameters of the survey:

- each customer with receive questions about a certain subset of the donuts;
- a customer can only be asked about products that they have purchased;
- each customer \( i \) should be asked about a number of products between \( c_i \) and \( c'_i \);
- to collect sufficient data about each kind of donut, there must be between \( p_j \) and \( p'_j \) distinct customers asked about each donut kind \( j \).

The question is whether there is a way to design a questionnaire for each customer, so as to satisfy all these conditions. Describe how to model this problem with a flow network, and design and analyze an algorithm for the problem.

**Extra Credit:** It is not difficult to prove that a triangle \( T \) with area 1 unit can be divided into three triangles \( T_1, T_2, T_3 \) by adding a single point inside it such that the areas of the three triangles realize any three numbers \( a_1, a_2, a_3 \) so long as \( a_1 + a_2 + a_3 = 1 \) and \( 0 \leq a_i \leq 1 \) for all \( i \). Now consider a rectangle \( R \) with area 1 unit. Can you divide it into 4 convex quadrilaterals \( R_1, R_2, R_3, R_4 \) by adding a single point inside it such that the areas of the four quadrilaterals realize any four numbers \( a_1, a_2, a_3, a_4 \) so long as \( a_1 + a_2 + a_3 + a_4 = 1 \) and \( 0 \leq a_i \leq 1 \) for all \( i \)? If so give a proof; otherwise give a counterexample.