1. (20 pts) Given a directed graph $G = (V, E)$ with capacity $c(u, v) > 0$ for each edge $(u, v) \in E$ and demand $r(v)$ at each vertex $v \in V$, a routing of flow is a function f such that

   - for all $(u, v) \in E$, $0 \le f(u, v) \le c(u, v)$, and
   - for all $v \in V$,
   $$\sum_{u:(u,v)\in E} f(u, v) - \sum_{u:(v,u)\in E} f(v, u) = r(v),$$
   i.e., the total incoming flow minus the total outgoing flow at vertex v is equal to $r(v)$. Notice that the demand $r(v)$ can take positive value, negative value, or zero.

   (a) Show how to find a routing or determine that one does not exist by reducing to a maximum flow problem.

   Rubric
   - 5 pts for giving the correct maximum flow problem
   - 5 pts for showing the equivalence of the two problems (2.5 pts for each direction)

   (b) Suppose that additionally there is a lower bound $l(u, v) > 0$ at each edge (u, v), and we are looking for a routing f satisfying $f(u, v) \ge l(u, v)$ for all $(u, v) \in E$. Show how to find such a routing or determine that one does not exist by reducing to a maximum flow problem.

   Rubric
   - 5 pts for giving the correct routing problem without lower bounds
   - 5 pts for showing the equivalence of the two problems (2.5 pts for each direction)

Solution: (a) Create a source s and a sink t, and add directed edges from s to every node $v \in V$ whose demand $r(v) < 0$ with capacity $-r(v)$ and add directed edges from every node $v \in V$ whose demand $r(v) > 0$ to the sink t with capacity $r(v)$.

If a flow f' in this new graph $G' = (\{V, s, t\}, E')$ saturates all new edges (notice that it has to be a maximum flow), then it yields a routing in the original graph. This can be shown by verifying the two conditions in the definition of routing.

1. f' is a flow in this graph. So for all $(u, v) \in E, 0 \leq f'(u, v) \leq c(u, v)$.
2. Since f' in this new graph saturates all new edges.

   This means:

   a.$f'(s, v) = -r(v)$ for $r(v) < 0, v \in E$

   b.$f'(v, t) = r(v)$ for $r(v) > 0, v \in E$.

   c.$\sum_{u \in E:(u,v) \in E'} f'(u, v) = \sum_{u \in E:(v,u) \in E'} f'(v, u)$.

   d.If $r(v) < 0, v \in E$, $\sum_{u \in E:(u,v) \in E} f'(u, v) + f'(s, v) = \sum_{u \in E:(v,u) \in E} f'(v, u)$.

   e.If $r(v) > 0, v \in E$, $\sum_{u \in E:(u,v) \in E'} f'(u, v) = \sum_{u \in E:(v,u) \in E} f'(v, u) + f'(v, t)$.

   So we have:

   If $r(v) < 0, v \in E$, $\sum_{u \in E:(u,v) \in E} f'(u, v) - r(v) = \sum_{u \in E:(v,u) \in E} f'(v, u)$.

   If $r(v) > 0, v \in E$, $\sum_{u \in E:(u,v) \in E'} f'(u, v) = \sum_{u \in E:(v,u) \in E} f'(v, u) + r(v)$.

   In general, these become:

$$\sum_{u:(u,v) \in E} f(u, v) - \sum_{u:(v,u) \in E} f(v, u) = r(v).$$

We also need to prove that, if there is a routing, there is a flow that saturates all new edges. This is straightforward: assign flow to each edge in the original graph according to the routing, and then there is precisely enough excess flow at vertices with $r(v) > 0$ to saturate the edge from v to t and there is precisely enough missing flow at vertices with $r(v) < 0$ to saturate the edge from s to v.

So, there is a routing iff the maximum flow saturates all the new edges.

(b) This can be solved by reducing to (a). Modify the graph and demands as follows: Replace each edge $e = (v, w)$ with three edges (v, v'), (v', w') and (w', w), and set the demands for the two new vertices as $r(v') = l(e)$ and $r(w') = -l(e)$. Now, any routing f in the original graph that satisfies the lower bounds can be extended to a routing in this new graph by making the flow on edge (v, v') equal to $f(e)$, the flow on edge (v', w') equal to $f(e) - l(e)$ and the flow on edge (w', w) equal to $f(e)$.

In the other direction, any flow on the new graph must satisfy $f(v, v') = f(v', w') + l(e)$ and $f(w', w) = f(v', w') + l(e)$. Thus the flows on edges (v, v') and (w', w) are equal, and we can set the flow on edge e in the original graph equal to this value to produce a routing in the original graph (since all the demand-equations at original vertices remain satisfied by this choice). Moreover, since $f(v', w')$ must be nonnegative, we have that $f(v, v') \geq l(e)$.

2. (20 pts extra credit) Even though in this class we focus on those greedy algorithms that generate optimal solutions, in general a greedy algorithm may not give an optimal solution. So, we are interested in those greedy algorithms that generate a good enough solution, i.e., not too far from the optimal solution. Let us consider one such problem as follows.

   Given subsets $S_1, S_2, \cdots, S_n$ of a set S of points and an integer m, a maximum $m-$cover is a collection of m of the subsets that covers the maximum number of points of S. Finding a maximum $m-$cover is a computationally hard problem. Give a greedy algorithm that achieves approximation ratio $1-1/e$; i.e., let y be the maximum number of points that can be covered by m subsets and x be the number of points that are covered by the m subsets generated by your algorithm, then give a greedy algorithm such that $x \geq (1 - 1/e)y$.

   Some useful hints:

   - You may want to use the inequality $(1 - 1/m)^m \leq 1/e$ for integer $m \geq 1$.
   - You may want to use induction at some point.

   Rubric

   - 10 pts for giving a good greedy algorithm
   - 10 pts for showing the algorithm has the desired approximation ratio

   Solution: A greedy strategy is to repeatedly pick the subset that covers the largest number of remaining uncovered points until we have selected m subsets. Let $S' \subset S$ be a subset of maximum size that can be covered by m subsets.

   Let x_i be the number of points covered after i rounds of the greedy algorithm, and let $y_i = |S'| - x_i$. We have that $x_0 = 0$ and $y_0 = |S'|$ (the maximum number of points that can be covered by m subsets). We now show that that $y_i \leq (1 - 1/m)^i y_0$, which can be done by induction. Clearly it holds for $i = 0$. Now in the ith round, there are at least y_{i-1} points of S' that remain uncovered, and yet m subsets cover all of S'. Thus there exits some subset that covers at least $1/m$ fraction of the remaining points. Thus in round i we cover at least $1/m$ of the y_{i-1} remaining points, and thus $y_i \leq (1 - 1/m)y_{i-1}$ which by induction yields $y_i \leq (1 - 1/m)^i y_0$. Thus, we have that

$y_m \leq (1 - 1/m)^m \leq y_0/e$. Now notice that $x_m = y_0 - y_m$. Thus, $x_m \geq (1 - 1/e)y_0$, i.e., the algorithm achieves an approximation ratio of $1 - 1/e$.