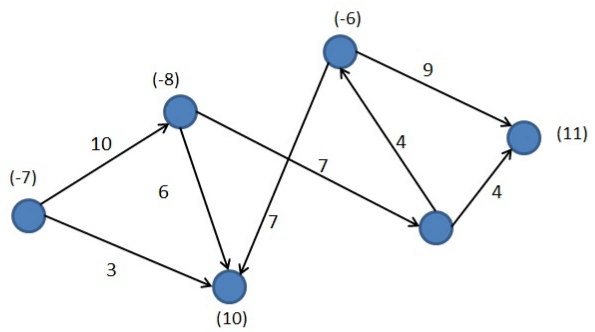
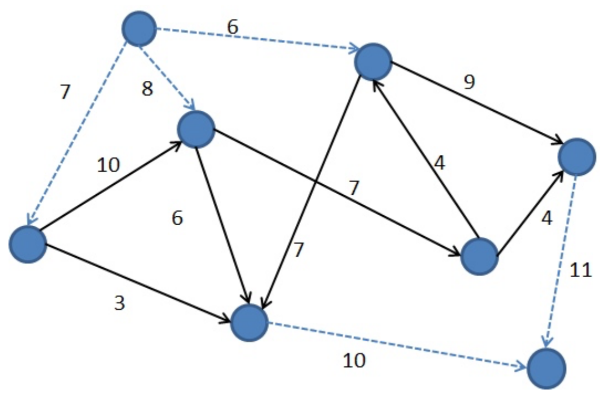
Homework 9

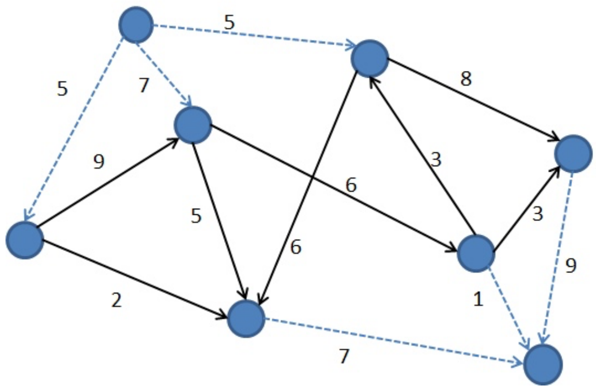
Problem 1. (10pts)
The following graph G is an instance of a circulation problem with demands. The edge weights represent capacities and the node weights (in parentheses) represent demands. A negative demand implies source.



a) Transform this graph into an instance of max-flow problem.
Solution: (5pts, no partial credits)

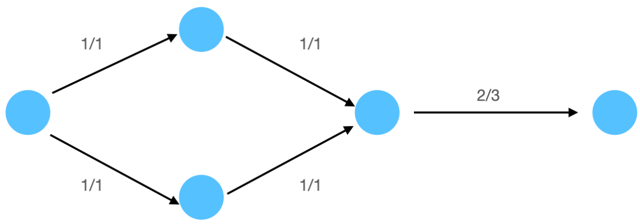


b) Now, assume that each edge of G has a constraint of lower bound of 1 unit, i.e., one unit must flow along all edges. Find the new instance of max-flow problem that includes the lower bound information. (Find G' in lecture slides)
Solution: (5pts, no partial credits)

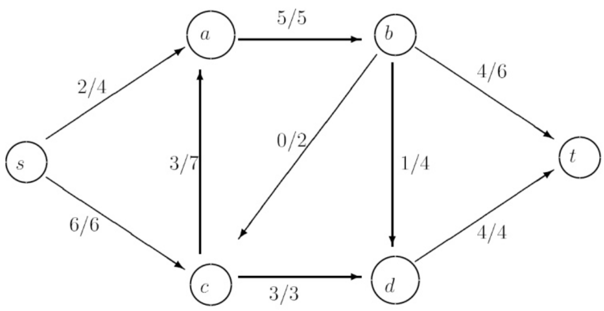


Problem 2. (9pts)
Determine if these statements are correct.

a) In a flow network, if the capacity of every edge is odd, then there is a maximum flow in which the flow on each edge is odd. (3pts)
Solution: False
Counter example:



b) The following flow is a maximal flow. (3pts)
Note: The notation a/b describes a units of flow on an edge of capacity b.



Solution: True

c) Given the min-cut, we can find the value of max flow in $O(|E|)$. (3pts)
Solution: True, Value of max flow = capacity of min-cut (iterate over all edges of the min-cut)

Problem 3. (11pts)
A company sells k different products, and it maintains a database which stores which customers have bought which products recently. We want to send a survey to a subset of n customers. We will tailor each survey so it is appropriate for the particular customer it is sent to. Here are some guidelines that we want to satisfy:

- The survey sent to a customer will ask questions only about the products this customer has purchased.

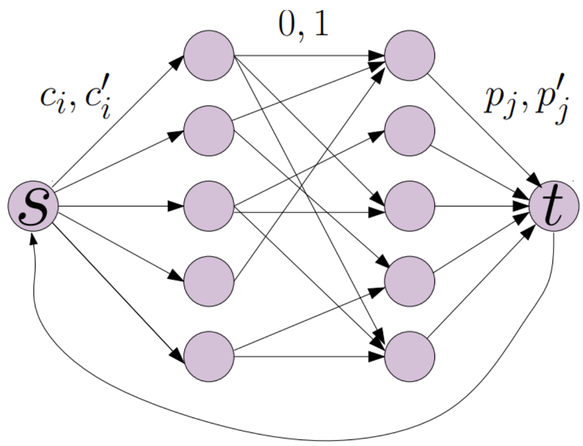
- We want to get as much information as possible, but do not want to annoy the customer by asking too many questions. (Otherwise, they will simply not respond.) Based on our knowledge of how many products customer i has purchased, and easily they are annoyed, our marketing people have come up with two bounds $0 \leq c_i \leq c'_i$. We will ask the ith customer about at least c_i products they bought, but (to avoid annoying them) at most c'_i products.

- Again, our marketing people know that we want more information about some products (e.g., new releases) and less about others. To get a balanced amount of information about each product, for the jth product we have two bounds $0 \leq p_j \leq p'_j$, and we will ask at least p_j customers about this product and at most p'_j customers.

The problem is how to assign questions to consumers, so that we get all the information we want to get, and every consumer is being asked a valid number of questions.
Solution:

First, we build a bipartite graph having consumers on one side, and products on the other side. Next, we insert the edge between consumer i and product j if the product was used by this consumer. The capacity of this edge is going to be 1. Intuitively, we are going to compute a flow in this network which is going to be an integer number. As such, every edge would be assigned either 0 or 1, where 1 is interpreted as asking the consumer about this product.

The next step, is to connect a source s to all the consumers, where the edge (s, i) has lower bound c_i and upper bound c'_i. Similarly, we connect all the products to the destination t, where (j, t) has lower bound p_j and upper bound p'_j. We would like to compute a flow from s to t in this network that comply with the constraints. However, we only know how to compute a circulation on such a network. To overcome this, we create an edge with infinite capacity between t and s. Now, we are only looking for a valid circulation in the resulting graph G which complies with the aforementioned constraints. See figure below for an example of G.



Given a circulation f in G it is straightforward to interpret it as a survey design (i.e., all middle edges with flow 1 are questions to be asked in the survey). Similarly, one can verify that given a valid survey, it can be interpreted as a valid circulation in G. Thus, computing circulation in G indeed solves our problem.

Rubric:

- Build the correct graph (6pts), -1 for each missing node/edge, or incorrect capacity

- Reduce to circulation problem and make correct conclusion (5pts). -3 if only computing max-flow. Deduct points for other incorrect statements.

Problem 4. (10pts)
Kleinberg and Tardos, Chapter 7, Exercise 7
Solution:

We build the following flow network. There is a node v_i for each client i, a node w_j for each base station j, and an edge (v_i, w_j) of capacity 1 if client i is within range of base station j. We then connect a super-source s to each of the client nodes by an edge of capacity 1, and we connect each of the base station nodes to a super-sink t by an edge of capacity L.

We claim that there is a feasible way to connect all clients to base stations if and only if there is an $s - t$ flow of value n. If there is a feasible connection, then we send one unit of flow from s to t along each of the paths s, v_i, w_j, t, where client i is connected to base station j. This does not violate the capacity conditions, in particular on the edges (w_j, t), due to the load constraints. Conversely, if there is a flow of value n, then there is one with integer values. We connect client i to base station j if the edge (v_i, w_j) carries one unit of flow, and we observe that the capacity condition ensures that no base station is overloaded.

The running is the time required to solve a max-flow problem on a graph with $O(n + k)$ nodes and $O(nk)$ edges.

Rubric:

- 4pts for constructing correct graph.
- 4pts for proving that a feasible connection is corresponding to a $s - t$ flow.
- 2pts for polynomial time complexity.

Problem 5. (practice)
Kleinberg and Tardos, Chapter 7, Exercise 9
Solution:

We build the following flow network. There is a node v_i for each patient i, a node w_j for each hospital j, and an edge (v_i, w_j) of capacity 1 if patient i is within a half hour drive of hospital j. We then connect a super-source s to each of the patient nodes by an edge of capacity 1, and we connect each of the hospital nodes to a super-sink t by an edge of capacity $\lceil n/k \rceil$.

We claim that there is a feasible way to send all patients to hospitals if and only if there is an $s - t$ flow of value n. If there is a feasible way to send patients, then we send one unit of flow from s to t along each of the paths (s, v_i, w_j, t), where patient i is sent to hospital j. This does not violate the capacity conditions on the edges (w_j, t), due to the load constraints. Conversely, if there is a flow of value n, then there is one with integer values. We send patient i to hospital j if the edge (v_i, w_j) carries one unit of flow, and we observe that the capacity condition ensures that no hospital is overloaded. The running is the time required to solve a max-flow problem on a graph with $O(n + k)$ nodes and $O(nk)$ edges.