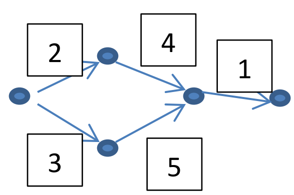
Tutorial on Network Flow

1. Suppose a graph, in addition to capacities on edges, has a capacity on every vertex such that the total inflow (and therefore the total outflow from the node) cannot exceed the capacity of the vertex. The other constraints remain the same as in the standard maximum-flow problem. Can you find the maximum flow in this graph?

   [Solution]:
   - Replace each vertex v with two vertices v1 and v2, with a directed edge from v1 to v2 with capacity equal to the vertex capacity.
   - For each edge (u, v), replace (u, v) with (u,v1) with same capacity
   - For each edge (v, w), replace (v, w) with (v2, w) with same capacity
   - Now you have a graph with only edge capacities. Find the maximum flow in it.

2. Suppose all capacities in a graph are distinct? Will the max-flow be unique?

   [Solution]: No. Consider the following graph. It is obvious that the max flow is not unique (can use either the 2-4 capacity path or the 3-5 capacity path to achieve the max flow of 1.



3. A company running a factory has to arrange for extra maintenance staff to work over vacation periods. There are K vacation periods in the year, numbered from 1 to K, with the i-th vacation period having d(i) days. There are N maintenance staff available numbered from 1 to N, with the j-th maintenance being available for a total of c(j) days. For each vacation period i, each staff can only work on a subset of the d(i) days (for example, if a vacation period spans over Friday, Saturday and Sunday, staff 1 may be available for only Friday and Saturday but not Sunday, staff 2 may be available for only Friday and Sunday but not Saturday, and staff 3 may be available for all days). What is the maximum no. of vacation days that can be covered by the company with its available staff?

   [Solution]: Create a bipartite graph with two partitions X and Y. X contains one node for each maintenance staff (say node j for staff j). Y contains one node for each vacation day of each vacation period (basically all vacation days, so if you have 3 vacation periods with 2, 4, and 3 days respectively, you will have 2+4+3 = 9 nodes in Y). Add an edge from node j in X to node i in Y if maintenance staff j can work on the vacation day represented by node i.

Now add two vertices s and t. Add an edge from s to each node j in X with capacity c(i). Add an edge from each node i in Y to t with capacity 1.
Now find the maximum flow from s to t. The value of the flow is the required answer.

4. Given an undirected graph, we want to find the smallest subset of edges that will disconnect the graph (the actual set of edges need to be found). Design an efficient algorithm for it.

   [Solution sketch] Run the algorithm for finding edge connectivity. For the pair (u,v) that gives the minimum, traverse the final residual graph from u to find the set S of all nodes reachable from u in the residual graph. Then find all edges that go between S and V-S in the original graph, where V is the set of all nodes.

5. Find a maximum matching in a bipartite graph in O(VE) time (assume E = Ω(V)).

   [Solution sketch] Use preflow push with push-relabels in a specific order. Suppose the two partitions are X and Y (the source s connected to X). Then from the initial preflow and h values, first relabel all nodes in X, then push from all nodes in X to nodes in Y, then relabel all nodes in Y, then push from all nodes in Y to t, then push back any excess flow in similar order (complete and verify the complexity).