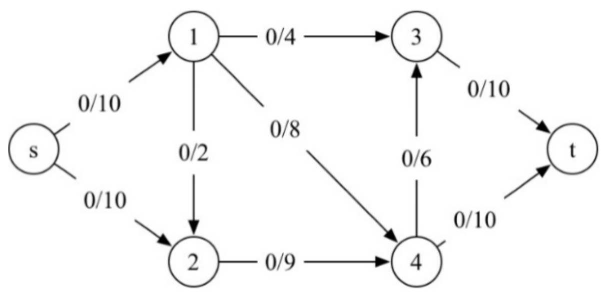
# Algorithm Design and Analysis: Homework #3

Due: 10:15am on November 2, 2017

*Grader: Zhou Yang and Meng YaZhou*

## Problem 1

Run the Ford-Fulkerson algorithm on the flow network in the figure below, and show the residual network after each flow augmentation. For each iteration, pick the augmenting path that is lexicographically smallest. ( e.g., if you have two augmenting path $1 \to 3 \to t$ and $1 \to 4 \to t$, then you should choose $1 \to 3 \to t$, which is lexicographically smaller than $1 \to 4 \to t$)




## Problem 2

There are $n$ staff members in a company. Each staff member belongs to a department. There are $k$ departments. Now we have to choose one staff member from each department to a company-level committee. In this committee, there must be $m_1$ number of A-class staff members, $m_2$ number of B-class staff members, $m_3$ number of C-class staff members, and $m_4$ number of D-class staff members (so we have $m_1 + m_2 + m_3 + m_4 = k$). We are given the list of staff members, their home departments, and their classes (A, B, C, D). Describe an efficient algorithm to determine who should be on the committee such that the constraints are satisfied.

## Problem 3

Let $G = (V, E)$ be a flow network with source $s$, sink $t$, and integer capacities. Suppose that we are given a maximum flow in $G$.

1. Suppose that we increase the capacity of a single edge $(v, u) \in E$ by 1. Give an $O(V + E)$ time algorithm to update the maximum flow.

2. Suppose that we decrease the capacity of a single edge $(v, u) \in E$ by 1. Give an $O(V + E)$ time algorithm to update the maximum flow.

## Problem 4

Suppose we have an $n$-by-$m$ matrix in which each element is either 0 or 1. We know the row sums $r = (r_1, ..., r_n)$ and the column sums $c = (c_1, ..., c_m)$ (it is guaranteed that $\sum_i r_i = \sum_j c_j$). Design a polynomial-time algorithm based network flow to decide whether such a matrix exists or not.

**Examples:** $n = m = 3$ with $c = (1, 2, 0)$ and $r = (1, 1, 1)$ (answer=yes) or $r = (3, 0, 0)$ (answer=no):

Yes

| | | | |
|---|---|---|---|
| 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 2 | 0 | |

No

| | | | |
|---|---|---|---|
| 1 | 1 | 1 | 3 |
| | | | 0 |
| | | | 0 |
| 1 | 2 | 0 | |

## Problem 5

Consider a network $G = (V, E)$. In addition to having a capacity $c(u, v)$ for every edge $(u, v)$, we also have a capacity $c(u)$ for every vertex $u$. A flow $f$ is feasible only if, in addition to the conservation constraint and the edge capacity constraint, it also satisfies the vertex capacity constraint: $\forall v \in V, \sum_{u:(u,v)\in E} f(u, v) \le c(v)$. Give an efficient algorithm to solve the maximum flow problem on this network.

Hint: You can reduce the problem to the standard maximum flow problem.

## Problem 6

Given a network $G = (V, E)$, design a polynomial time algorithm to determine whether $G$ has a unique minimum s-t cut.