COL 351

TUTORIAL SHEET 10

1. **[KT-Chapter7]** Consider the following problem. You are given a flow network with unit-capacity edges: it consists of a directed graph $G = (V, E)$, a source s and a sink t, and $u_e = 1$ for every edge e. You are also given a parameter k. The goal is delete k edges so as to reduce the maximum $s-t$ flow in G by as much as possible. In other words, you should find a set of edges $F \subseteq E$ so that $|F| = k$ and the maximum $s-t$ flow in the graph $G_0 = (V, E \setminus F)$ is as small as possible. Give a polynomial-time algorithm to solve this problem.

2. **[KT-Chapter7]** In a standard $s-t$ maximum flow problem, we assume edges have capacities, and there is no limit on how much flow is allowed to flow through a node. In this problem, we consider the variant of the maximum flow and minimum cut problems with node capacities. Let $G = (V, E)$ be a directed graph, with source s, sink t , and non-negative node capacities u_v for each $v \in V$. Given a flow f in this graph, the flow though a node v is defined as $\sum_{e \in \delta^-(v)} f_e$. where $\delta^-(v)$ denotes the edges coming into v. We say that a flow is feasible, if it satisfies the usual flow-conservation constraints and the node-capacity constraint: the flow through a node v cannot exceed c_v. Give a polynomial-time algorithm to find a $s-t$ maximum flow in such node-capacitated network. Define an $s-t$ cut for node-capacitated networks, and show that the analog of the Maximum Flow Min Cut theorem holds true.

3. Suppose in a directed graph G, there are k edge-disjoint paths from s to t and from t to u. Are there k edge disjoint paths from s to u ?

4. **[KT-Chapter7]** In sociology, one often studies a graph G in which nodes represent people, and edges represent those who are friends with each other. Let's assume for purposes of this question that friendship is symmetric, so we can consider an undirected graph. Now, suppose we want to study this graph G, looking for a close-knit group of people. One way to formalize this notion would be as follows. For a subset S of nodes let $e(S)$ denote the number of edges in S, i.e., the number of edges that have both ends in S. We define the cohesiveness of S as $e(S)/|S|$. A natural thing to search for would be the set S of people achieving the maximum cohesiveness. Give a polynomial time algorithm that takes a rational number α and determines whether there exists a set S with cohesiveness at least α. Give a polynomial time algorithm to find a set S of nodes with maximum cohesiveness.

5. You are given an $n \times n$ matrix A with real entries. You would like to round each of the entries x in the array to either $\lfloor x \rfloor$ or $\lceil x \rceil$ such that the row and column sums don't change. Give an efficient algorithm which either achieves this rounding or declares that no such rounding is possible.