COMP 271 Design and Analysis of Algorithms
2003 Spring Semester
Written Assignment # 2
Distributed: March 18, 2003
Due: April 1, 2003 *at start of class*

Note: Please follow the guidelines on doing your own work and avoiding plagiarism given on the class home page.

Problem 1. (25 pts) Given a connected undirected graph G described by an adjacency list representation design an efficient algorithm to find a path in G that goes through each edge exactly once in each direction. Your algorithm should print out the edges in the direction and order in which they are traversed, i.e., printing out

$$(u, v), (v, w), (w, v), (v, u)$$

means going from u to v to w to v to u.

Argue the correctness of your algorithm and analyze its running time.

Problem 2. (25 pts) An undirected graph $G = (V, E)$ is called *injective* if the edges can be directed such that each vertex has in-degree 1.

1. Prove that each connected component of an injective graph must contain exactly one cycle.
2. Give a linear-time algorithm to orient the edges of an injective graph such that each vertex has in-degree 1.

Problem 3. (50 pts) Let $G = (V, E)$ be an undirected weighted graph with no negative edge weights. The *bottleneck* value of a path

$$(u_0, u_1), (u_1, u_2), \ldots, (u_{n-2}, u_{n-1}), (u_{n-1}, u_n)$$

is $\min_{1 \leq i \leq n} W(u_{i-1}, u_i)$. (think of the edges as being water pipes and the weights as the maximum-flow that can pass through a pipe every hour. The *maximum* amount that can flow through a path in an hour is the value of the *minimum-weight* edge on the path. This is the bottleneck value of the path).

Design an algorithm that, given s and t, finds a path from s to t with *maximum* bottleneck value among all $s - t$ paths.

Argue the correctness of your algorithm and analyze its running time.