CS473: Algorithms I **21.07.2011**

Problems

1. The transpose of a directed graph G=(V, E) is the graph G^T=(V, E^T), where $E^T = \{(v,u) \in V \times V : (u, v) \in E\}$. Thus, G^T is G with all its edges reversed. Describe efficient algorithms for computing G^T from G, for both the adjacency list and adjacency matrix representations of G. Analyze the running times of your algorithms.

2. What is the running time of BFS if we represent its input graph by an adjacency matrix and modify the algorithm to handle this form of input?

3. There are two types of professional wrestlers: good guys and bad guys. Between any pair of professional wrestlers, there may or may not be a rivalry. Suppose we have n professional wrestlers and we have a list of r pairs of wrestlers for which there are rivalries. Give an O(n + r)-time algorithm that determines whether it is possible to designate some of the wrestlers as good guys and the remainder as bad guys such that each rivalry is between a good guy and a bad guy. If it is possible to perform such a designation, your algorithm should produce it.

4. Rewrite the procedure DFS, using a stack to eliminate recursion.

5. Give a counterexample to the conjecture that if a directed graph G contains a path from u to v, and if u.d < v.d in a depth-first search of G, then v is a descendant of u in the depth-first forest produced.

6. Give a counterexample to the conjecture that if a directed graph G contains a path from u to v, then any depth-first search must result in v.d < u.f.

7. Give an algorithm that determines whether or not a given undirected graph G = (V, E) contains a cycle. Your algorithm should run in O(V) time, independent of E.

8. Another way to perform topological sorting on a directed acyclic graph G = (V, E) is to repeatedly find a vertex of in-degree 0, output it, and remove it and all of its outgoing edges from the graph. Explain how to implement this idea so that it runs in time O(V + E). What happens to this algorithm if G has cycles?

9. Let G = (V, E) be a directed graph in which each vertex $u \in V$ is labeled with a unique integer L(u) from the set {1, 2, …, |V|}. For each vertex $u \in V$, let R(u) be the set of vertices that are reachable from u. Define min(u) to be the vertex in R(u) whose label is minimum. Give an O(V+E)-time algorithm that computes min(u) for all vertices $u \in V$.

10. Give an example graph which has at least two different topological orderings. Give an example graph which has exactly one topological ordering.