Theory of Computation
Homework Set 1
February 24, 2005
http://staffweb.ncnu.edu.tw/shieng/

Problem 1 A directed graph is *acyclic* if it has no cycles. Show that any acyclic graph with n nodes has a source (a node with no incoming edge).

Problem 2 Show that a graph with n nodes is acyclic if and only if its nodes can be numbered 1 to n so that all edges go from lower to higher numbers (use the property in Problem 1 above repeatedly).

Problem 3 Describe a polynomial-time algorithm that decides whether a graph is acyclic. (Implement the idea in Problem 2 above. With care, your algorithm should not spend more than a constant amount of time per edge.)

Problem 4 Give a counterexample to show that there is an acyclic graph that has no source.

Problem 5 Show that a graph is bipartite (that is, its nodes can be partitioned into two sets, not necessarily of equal cardinality, with edges going only from one to the other) if and only if it has no odd-length cycles.

Problem 6 Describe a polynomial algorithm for testing whether a graph is bipartite.