CS466/666 Spring 2013

Assignment 5

(Due Noon, Monday June 17, 2013)

You are on your honour to present your own work and acknowledge your sources.

1. A tree is a connected graph with no cycles. As a tree has no cycles, it is bipartite (as it has no odd length cycles) and hence the polynomial time algorithm we saw in the class, to find the minimum vertex cover applies to trees.

   The purpose of this problem is to find a much simpler polynomial time algorithm to find a minimum vertex cover in a tree. Recall that a vertex cover in a graph is a subset S of vertices such that for every edge in the graph at least one of its end points is in S.

   Give a polynomial time algorithm (different from the one we saw for bipartite graphs) to find a minimum vertex cover in a tree. (Hint: Every tree has a vertex with degree 1).

2. A tournament is a directed graph with exactly one *directed* edge between every pair of vertices. I.e. take a complete undirected graph and give arbitrary orientation to each edge, what you get is a tournament. A feedback vertex set in a tournament T is a subset S of vertices such that $T \setminus S$ (T with vertices of S removed) has no directed cycles.

   (a) Prove that a tournament has a directed cycle if and only if it has a directed cycle of length 3.

   (b) Use Part (a) to give a $O(3^k n^c)$ algorithm to determine whether a given tournament on n vertices has a feedback vertex set of size at most k. Here c is some constant independent of k. Give an estimate for your c.