CSL356 **Due on :** August 20, 2010

Homework I

1. We say that a directed graph is *nice* if for every pair of vertices u and v in the graph, either u is reachable from v **or** v is reachable from u. Give a linear time algorithm to check if a graph is nice.

2. Consider a directed graph G where every vertex has a *prize* associated with it. Let $p(v)$ denote the prize at vertex v. For a vertex u, define $\max(u)$ as the maximum value of $p(w)$ over all vertices w which are reachable from u. Give a linear time algorithm which computes $\max(u)$ for all the vertices u in G.

3. Let G be an undirected graph and fix a vertex s in G. For each vertex v, let $N(v)$ denote the **number** of shortest paths from s to v (i.e., the length of each of these paths must be equal to the distance of v from s). Give a linear time algorithm which computes $N(v)$ for each vertex v in G. Note that you should not output the actual paths.