Homework 2

CPSC 629, Spring 2005

The homework is due on Monday, February 21, before class.

Name (print): ________________________________ UIN ________________

Read the chapters on graph algorithms in [CLRS] carefully, before you attempt to solve the problems. Consult Aho, Hopcroft and Ullman as well.

Problem 1 Depth-First Search. Show that an edge (u, v) is
(a) a tree edge or a forward edge if and only if $d[u] < d[v] < f[v] < f[u]$;
(b) a back edge if and only if $d[v] < d[u] < f[u] < f[v]$;
(c) a cross edge if and only if $d[v] < f[v] < d[u] < f[u]$.

Problem 2 Give a counterexample to the conjecture that if there is a path from u to v in a directed graph G, and if $d[u] < d[v]$ in a depth-first search of G, then v is a descendant of u in the depth-first forest produced. Give a careful explanation.

Problem 3 Topological-Sort(G) produces an ordering of the vertices by listing the vertices in decreasing finishing times of a DFS(G). Show that there exists a directed acyclic graph G that has a topological order that cannot be produced by any run of Topological-Sort(G). Explain the main idea behind your example. Choose your example as small as possible.

Problem 4 We denote the component graph of $G = (V, E)$ by $G^{scc} = (V^{scc}, E^{scc})$, where V contains one vertex for each strongly connected component, and E^{scc} contains an edge (u, v) if and only if there is an edge in E from one vertex in the strongly connected component u to a vertex in the strongly connected component v. Roughly speaking, G^{scc} is obtained by contracting the strongly connected components of G. Prove that G^{scc} is a directed acyclic graph.

[0]Problems 1 and 2 re-enforce your understanding of DFS. Problem 3 gives you an opportunity to learn more about the algorithm Topological-Sort(G). The last problem allows give you an opportunity to exercise some proofs; make sure that your arguments are elegant and easy to understand. Read about breadth-first search in [CLRS].