Name: __________________________
ID: __________________________

Be clear and concise. Write your answers in the space provided. Use the backs of pages for scratchwork.
1. **(10 points)** Short-answer problems.

(a) Do a depth-first search of the following graph. Whenever there is a choice, process nodes in alphabetical order. Show the DFS search forest, and write in the pre and post numbers.

(b) For the graph below, (i) next to each node, write down the distance to that node from node c; and (ii) draw the corresponding shortest path tree.
2. **(10 points)** For each of the following computational tasks, say whether you think it should take polynomial time or exponential time. If exponential, give a brief but concrete reason. If polynomial, briefly (but clearly) describe an algorithm for the problem and state its running time.

(a) Find all sources of a given directed graph.

(b) Find all cycles in a given directed graph.

(c) Find all topological orderings, if any, of a directed graph.
3. **(10 points)** Think of the Web as a large directed graph $G = (V, E)$, where each node $u \in V$ corresponds to a URL and an edge $(u, v)$ means that URL $u$ points to URL $v$. We will define a *community* on the web as being a subset of nodes $S \subseteq V$ such that: for any $u, v \in S$, it is possible to reach $v$ from $u$ by following pointers *and* it is possible to reach $u$ from $v$ by following pointers.

(a) Give an algorithm for finding the largest community on the web (that is, the community consisting of the most URLs). How long will your algorithm take?

(b) We will call a URL *universally accessible* if it can be reached from all other URLs, by following pointers. Give an algorithm which finds all universally accessible URLs. How long will your algorithm take?
4. **(10 points)** You are given: a directed graph $G = (V, E)$; a set of additional candidate edges $E'$ between nodes of $V$; and two specific nodes $s, t \in V$. Assume all edges have unit length. Find the edge in $E'$ whose addition to $G$ would most decrease the distance between $s$ and $t$.

(a) State the algorithm.

(b) Analyze its running time as a function of $|V|$, $|E|$, and $|E'|$. 