1. Let G be an undirected graph.

   (a) Suppose we start with two coins on two arbitrarily chosen nodes. At every step, each coin must move to an adjacent node. Describe an algorithm to compute the minimum number of steps to reach a configuration that two coins are on the same node.

   (b) Now suppose there are three coins, numbered 0, 1, and 2. Again we start with an arbitrary coin placement with all three coins facing up. At each step, we move each coin to an adjacent node at each step. Moreover, for every integer i, we flip coin i mod 3 at the ith step. Describe an algorithm to compute the minimum number of steps to reach a configuration that all three coins are on the same node and all facing up. What is the running time of your algorithm?

2. Let G be a directed acyclic graph with a unique source s and a unique sink t.

   (a) A *Hamiltonian path* in G is a directed path in G that contains every vertex in G. Describe an algorithm to determine whether G has a Hamiltonian path.

   (b) Suppose several nodes in G are marked to be *important*; also an integer k is given. Design an algorithm which computes all the nodes that can reach t through at least k important nodes.

   (c) Suppose the edges in G have real weights. Describe an algorithm to find a path from s to t with maximum total weight.

   (d) Suppose the vertices of G have labels from a fixed finite alphabet, and let $A[1..\ell]$ be a string over the same alphabet. Any directed path in G has a label, which is obtained by concatenating the labels of its vertices. Describe an algorithm to find the longest path in G whose labels are a subsequence of A.

3. Let G be a directed graph with a special source that has an edge to each other node in graph, and denote $scc(G)$ as the strong component graph of G. Let S and S' be two strongly connected components in G with $S \to S'$ an arc in $scc(G)$. (That is, if there is an arc between node $u \in S$ and $v \in S'$, then it must be $u \to v$.) Consider a fixed depth-first search performed on G starting at s; we define $\text{post}(\cdot)$ as the post-order numbering of the search.

   (a) Prove or disprove that we have $\text{post}(u) > \text{post}(u')$ for any $u \in S$ and $u' \in S'$.

   (b) Prove or disprove that we have $\max_{u \in S} \text{post}(u) > \max_{u' \in S'} \text{post}(u')$.