

Fundamental Algorithm Techniques

Problem Set #8

Review on November 29

Problem 1 (SCC and reversal, 4/10 pts). Let G a directed graph:

1. Describe an algorithm to compute the reversal $\text{rev}(G)$ of a directed graph G in $O(V + E)$ time.
2. Prove that for every directed graph G , the strong component graph $\text{scc}(G)$ ¹ is acyclic.
3. Prove that $\text{scc}(\text{rev}(G)) = \text{rev}(\text{scc}(G))$ for every directed graph G .
4. Fix an arbitrary directed graph G . For any vertex v of G , let $S(v)$ denote the strong component of G that contains v . For all vertices u and v of G , prove that u can reach v in G if and only if $S(u)$ can reach $S(v)$ in $\text{scc}(G)$.

Problem 2 (Euler Tour, 2/10 pts). An Euler tour of a strongly connected, directed graph $G = (V, E)$ is a cycle that traverses each edge of G exactly once (independantly of starting point), although it may visit a vertex more than once.

1. Show that G has an Euler tour if and only if $\text{in-degree}(v) = \text{out-degree}(v)$ ² for each vertex $v \in V$.
2. Describe an $O(E)$ -time algorithm to find an Euler tour of G if one exists. (Hint: Merge edge-disjoint cycles.)

Problem 3 (Topological Search, 4/10 pts). Perform Topological Sort on course example:
 $\{A \rightarrow B, A \rightarrow C, B \rightarrow C, B \rightarrow D, C \rightarrow E, D \rightarrow E, D \rightarrow F, G \rightarrow F, G \rightarrow E\}$.

Starting from A , find orders of the course. Start from another node at random and find one or two sortings³.

¹ $\text{scc}(G)$ merges every strong component into a single point

²in-/out-degree number or in/out-flowing arrows respectively.

³video topological sort...