CS 5633: Analysis of Algorithms

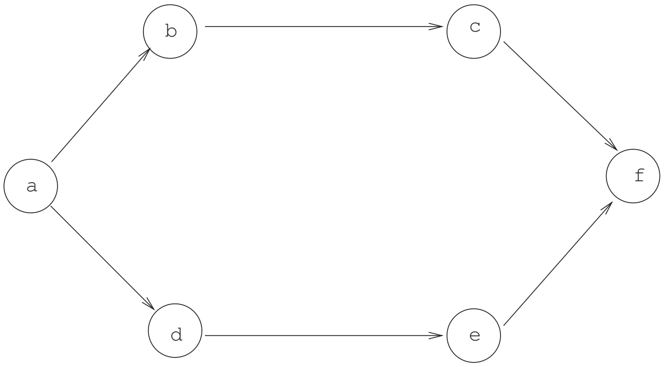
Homework 7

1. In class we considered the activity-selection problem, and we considered a greedy algorithm which iteratively took the activity with the earliest finishing times from the activities which are compatible with our current solution. We proved that the algorithm results in an optimal solution.

   (a) Another idea for a greedy algorithm would be to take the activity that will be active the least amount of time. That way we have as much of the time remaining as possible. This greedy algorithm, however, does not always result in an optimal solution. Give a counterexample which shows that this algorithm might fail.

   (b) What is another greedy algorithm (other than the earliest finishing time algorithm) which *will* always compute an optimal solution. Prove that your algorithm is correct.

2. How many valid topological sorts are there for this DAG?



3. Give an algorithm which determines whether or not an undirected graph contains a cycle. The algorithm should run in $O(n + m)$ time.

4. Prove that a connected graph G has a unique minimum spanning tree if the edge costs are all distinct.