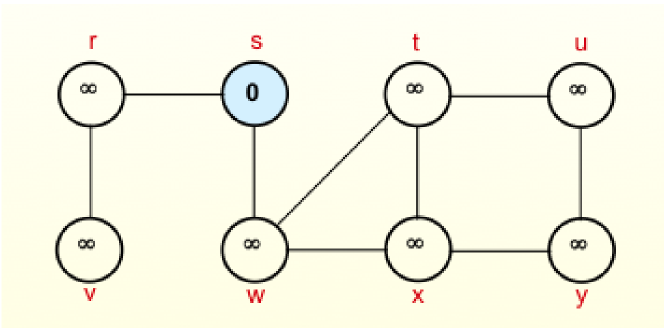
Lecture #13: Wed, 24 February 2016
Topics: Recitation Section - 8

Problem 1-1. (BFS + DFS)

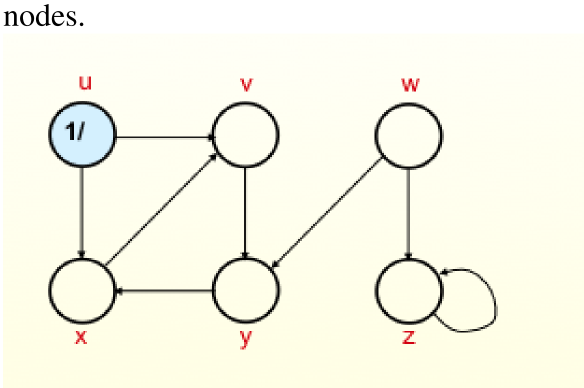
Consider this graph simple graph from lecture $G = (V, E)$:



(a) For all nodes $v \in V$, implement an efficient algorithm to find the distance from v to node s, the origin node, as well as the shortest path from v to s.

(b) Analyze the runtime of the proposed algorithm above.

(c) Given the simple example directed graph below, run DFS on the graph (assume a lexical ordering preference during DFS search). Output a topological sort of the nodes.



(d) Analyze the runtime of the algorithm from Part C.

(e) DFS trees have four type of edges: Tree edge, back edge, forward edge, cross edge. Define edge type and label each edge in the graph in part C accordingly.

(f) Your friend has run DFS on an undirected graph and has outputed the following edge type frequencies. Tree edge: 230, Back edge: 5, Forward Edge: 20. Cross Edge: 0. Your visualize the graph and see that it is acyclic. Explain why your friend's code is buggy.

(g) Your friend has run DFS on a graph and has outputed the following runtime log. for nodes v and u: $d[u] = 110, d[v] = 12, f[u] = 230, f[v] = 251$ Explain why your friend's code is buggy.

Problem 1-2. (Dynamic Programming Review: Longest Common Subsequence)

In this problem, we will build up a solution via dynamic programming to solve the Longest Common Subsequence problem.

(a) Build some intuition with toy examples. What the LCS of (GABA, AGTBA), and (GAC, AGCAT)? Remember: subsequences are not required to occupy consecutive positions within the original sequences

(b) Create a case-based rule to iterate from problem size S to S+1. (Hint: compare two sequences that end in the same character, vs ending in different characters).

(c) Use your case-based iterative step rule to fill out the 2D cache for finding the length of the longest common subsequence of the two strings: "MZJAWXU", "XMJYAUZ".

(d) How can we augment the cache above to find the actual longest subsequence, not just the length?