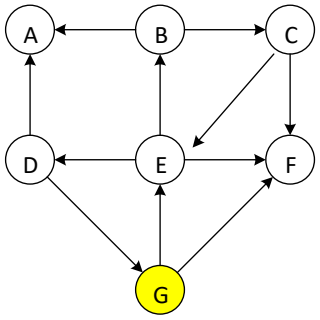
Assignment 6

Problem 1 (10 points). Run DFS on the following graph beginning at node G and show the sequence of nodes generated by the search. When you have two or more choices as the next node to visit, choose them in the alphabetical order.

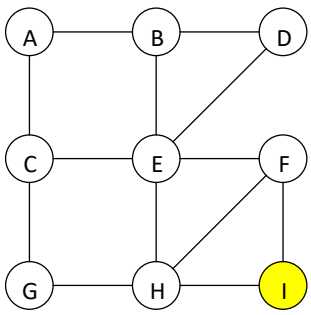


After completing the DFS, classify each edge as a *tree edge*, a *forward edge*, a *back edge*, or a *cross edge*.

Answer:
DFS path:
G => E => B => A. Backtrack to B;
B => C => F. Backtrack to F => E
E => D. Finished.

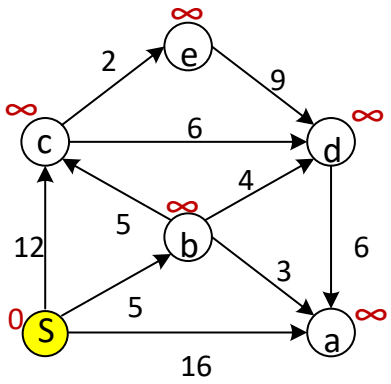
Tree Edge: G => E, E => B, B => A, B => C, C => F, E => D
Forward Edge: E => F, G => F
Back Edge: D => G, C => E
Cross Edge: E => D

Problem 2 (10 points). Run BFS on the following graph beginning at node I and show the sequence of nodes generated by the search. When you have two or more choices as the next node to visit, choose them in the alphabetical order.

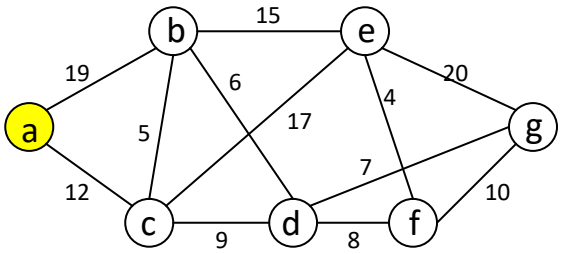


Answer:
BFS path:
I => F, I => H;
F => E, H => G;
E => B, E => D, G => C;
B => A

Problem 3 (10 points). Run Dijkstra's algorithm on the following graph beginning at node S.



Problem 3-(1). After each iteration, show the D values of all nodes (initial D values are shown above each node in red).
Answer:
Values at each node for each iteration:
1st Iteration: edge (s, a), (s, b), (s, c) are relaxed => D[a] = 16, D[b] = 5, D[c] = 12
2nd Iteration: edge(b, a), (b, c), (b, d) are relaxed => D[a] = 8, D[c] = 10, D[d] = 9
3rd Iteration: edge(c, e), (c, d) are relaxed => D[e] = 12, D[d] = 9
4th Iteration edge(d, a) is relaxed => D[a] = 8
Problem 3-(2). Show the shortest path from S to every other node generated by the algorithm
Answer:
S – a: 8
S – b: 5
S – c: 10
S – d: 9
S – e: 12
Problem 4 (10 points). Run the Prim-Jarnik algorithm on the following graph beginning at node *a*.



Problem 4-(1). Show the sequence of nodes in the order they are brought into the "cloud."
Answer:
Sequence of nodes: a, c, b, d, g, f, e

Problem 4-(2). Show the minimum spanning tree T, generated by the algorithm, as a set of edges.
Answer:
(a, c), (c, b), (b, d), (d, g), (d, f), (f, e)