CS5002 Discrete and Data Structures	Prof Higger
Fall 2020	Nov 6, 2020

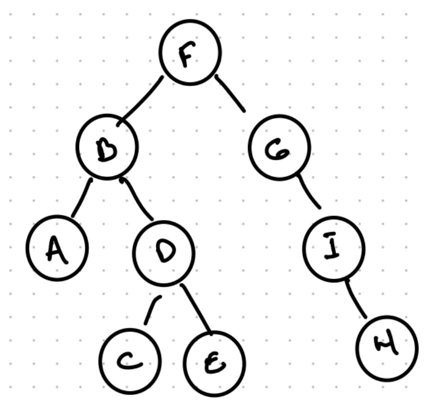
HW 08 Graphs

Due: Nov 13, 2020

Instructions:

- This homework exists to strengthen your understanding of concepts so that you may apply them elsewhere
- To get full credit, show intermediate steps leading to your answers.
- You are welcome to work on problems with classmates though you may not directly view another student's solution to a given problem while working together. Include a brief statement at the beginning of your homework which lists your homework group members: "Homework group: person A, person B". If you did not work with other students on the assignment write "Homework group: none". A 5 point penalty will be applied to all work which does not include this statement.
- Questions whose points are labelled with an addition sign are extra credit (e.g. "+4 points"). These are designed to push you, so have fun and don't worry if you're not making headway immediately: they're supposed to take some time. Excellence will come with practice.

Problem 1 [18 points (3 pts each)]: Tree Anatomy



The root of the Tree above is node F. A neighbor of node X is any node which shares an edge with X (e.g. G and I are neighbors while H and E are not). A sibling of node X is any node which shares a parent with X. An ancestor of node X is any node on the unique path from X to the root of the tree. X is a descendent of Y if Y is an ancestor of X. Assume that the sibling, neighbor, ancestor and descendent relation are reflexive while the parent and child relations are not.

Express the set all the nodes which meet each of the following criteria.

i Child of D

ii Parent of I

iii Sibling of D

iv Ancestor of H

v Descendent of B

vi the set of Neighbors of D or G which are not also neighbors of B

Problem 2 [18 points (3 pts each)]: Graph Taxonomy

If possible, draw an example of each graph as described. Otherwise, describe why such a graph does not exist. Unless otherwise specified, each graph is undirected and has exactly 5 nodes. Please use uppercase letters starting at A to index the nodes of your graph.

Remember, a path is a sequence of unique, adjacent edges.

i An acyclic graph where every pair of nodes has an edge.

ii A rooted tree where B is the root

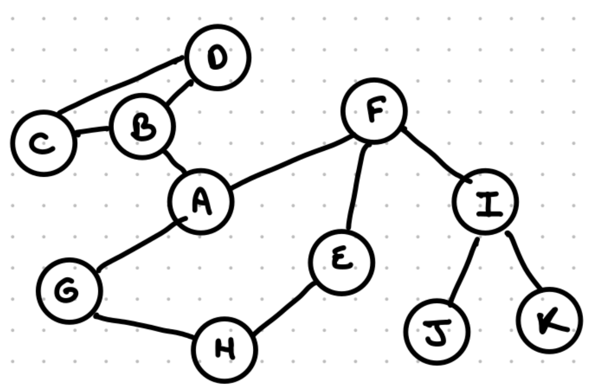
iii A graph which contains no cycles but is not a tree.

iv A weighted graph where every path from A to E has weight 2.

v A rooted tree with the minimal number of leafs.

vi A strongly connected, directed graph with the minimal number of edges. A strongly connected directed graph has a directed path between every permutation of two nodes. A directed path is a sequence of adjacent edges which moves with the direction of the edges. Note that because a directed path can't go 'backwards', we may not use the same path to connect A to B as we would to connect B to A.

Problem 3 [20 points (5 pts each)]: Breadth/Depth First Searches



Give the order of nodes in each of the searches below, starting at the indicated node. When a method may select from among many next candidate nodes, prefer the node which is alphabetically first.

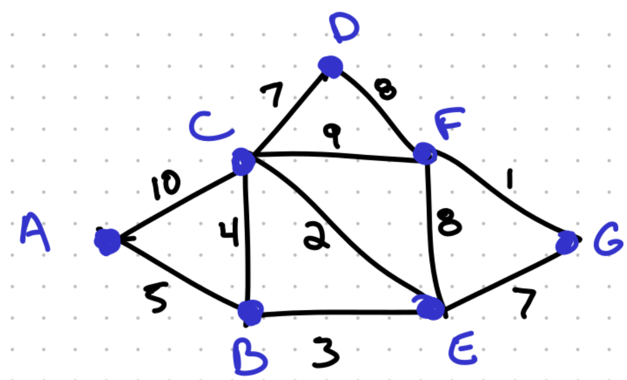
i Breadth First Search starting at node A

ii Depth First Search starting at node A

iii Breadth First Search starting at node E

iv Depth First Search starting at node E

Problem 4 [20 points]: Shortest Path (Dijkstra's Algorithm)

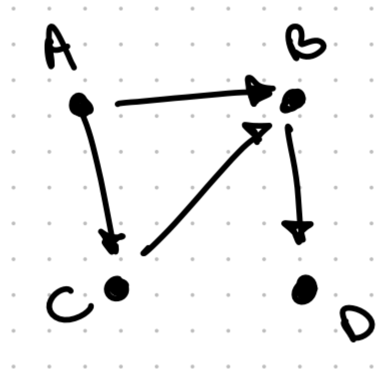


Using Dijkstra's algorithm, find the total weight of the shortest path from node a to g. You do not need to give the path itself, only its total weight. To show your work, give the vector representing the "shortest path from a" as well as the list of nodes "visited" every step. The first step of the algorithm is given as an example.

The shortest path which may visit nodes: [a] from node a to all others is:

a	b	c	d	e	f	g
0	5	10	∞	∞	∞	∞

Problem 5 [24 points (6 pts each)]: Graph Representation



The graph above is G_{directed}. Let $G_{\text{undirected}}$ be its undirected counterpart created by removing the edge direction associated with each edge. Represent each of the graphs below using the indicated method. Note that you must take special care to represent the directions of G_{directed} where necessary. Be sure to explain your representation of the direction of edges. Textbook example 16.22 in Fell and Aslam provides a scheme to encapsulate edge direction in an adjacency list representation.

i Adjacency List representation of $G_{\text{undirected}}$

ii Adjacency Matrix representation of $G_{\text{undirected}}$

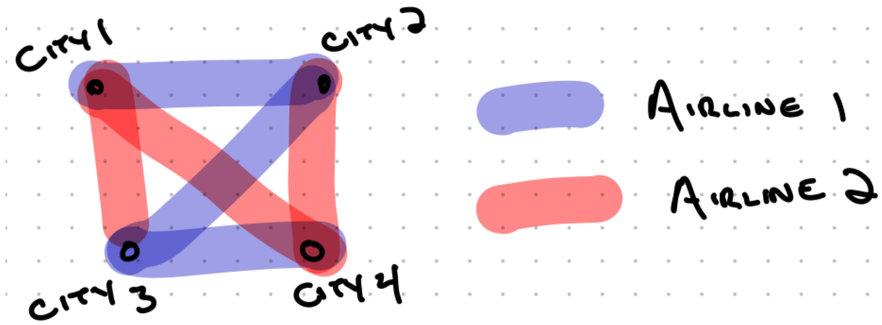
iii Adjacency List representation of G_{directed}

iv Adjacency Matrix representation of G_{directed}

Problem 6 [+3 points]: Duelling Connections

Consider a graph which represents the flights made by two airlines among N cities. Between every pair of cities exactly one airline flies the route. Show that there exists an airline which may carry a passenger from any city to any other city using only their own routes.

For example, consider the particular graph:



Airline 1 is able to carry passengers from any city to any other using only their routes. Notice that to achieve this Airline 1 may use indirect flights.