COMP20007 Design of Algorithms

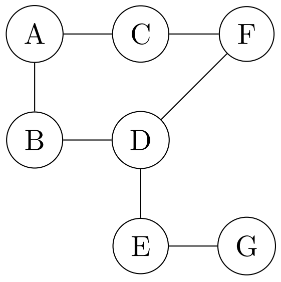
Week 6 Workshop

Tutorial

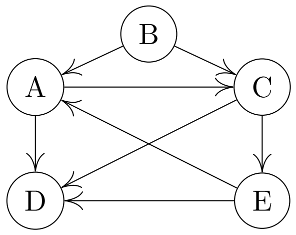
1. Depth First Search and Breadth First Search List the order of the nodes visited in the following graph when the following search algorithms are run:

(i) depth first search (ii) breadth first search

Start at A and break ties by choosing nodes in alphabetic order.



2. Tree, Back, Forward and Cross Edges A depth-first search of a directed graph can be represented as a tree (or a collection of trees, *i.e.*, a forest). Each edge of the graph can then be classified as a *tree edge*, a *back edge*, a *forward edge*, or a *cross edge*. A tree edge is an edge to a previously un-visited node, a back edge is an edge from a node to an ancestor, a forward edge is an edge to a non-child descendent and a cross edge is an edge to a node in a different sub-tree (*i.e.*, neither a descendent nor an ancestor). Draw a depth-first search tree based on the following graph, and classify its edges into these categories. Begin the depth-first search at A.

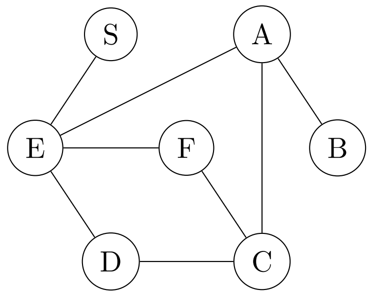


In an undirected graph, you wont find any forward edges or cross edges. Why is this true? You might like to consider the graph above, with each of its edges replaced by undirected edges.

3. Finding Cycles Explain how one can also use breadth-first search to see whether an undirected graph is cyclic. Which of the two traversals, depth-first and breadth-first, will be able to find cycles faster? (If there is no clear winner, give an example where one is better, and another example where the other is better.)

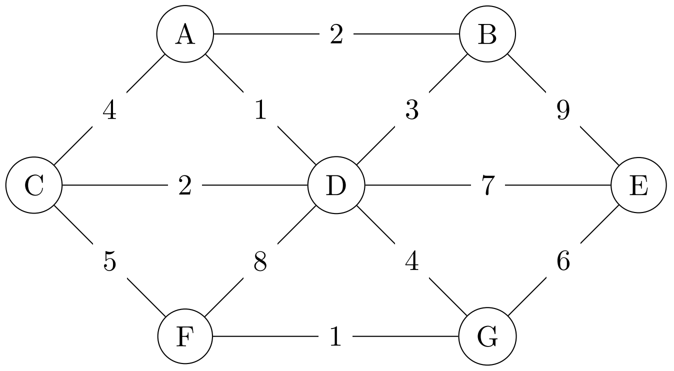
4. 2-Colourability Design an algorithm to check whether an undirected graph is 2-colourable, that is, whether its nodes can be coloured with just 2 colours in such a way that no edge connects two nodes of the same colour.

To get a feel for the problem, try to 2-colour the following graph.



Do you expect we could extend such an algorithm to check if a graph is 3-Colourable, or in general: k-Colourable?

5. Single Source Shortest Path with Dijkstra's Algorithm Dijkstra's algorithm computes the shortest path to each node in a graph from a single starting node (the 'source'). Trace Dijkstra's algorithm on the following graph, with node E as the source. Repeat the algorithm with node A as the source. How long is the shortest path from E to A? How about A to F?



6. Minimum Spanning Tree with Prim's Algorithm Prim's algorithm finds a minimum spanning tree for a weighted graph. Discuss what is meant by the terms 'tree', 'spanning tree', and 'minimum spanning tree'.

Run Prim's algorithm on the graph from Question 5, using A as the starting node. What is the resulting minimum spanning tree for this graph? What is the cost of this minimum spanning tree?

Computer Lab

Use today's lab time to work on Assignment 1. Make sure you can log in to `dimefox`, copy your code over, compile and run your program on the server.

If you've completed the assignment already feel free to use this time to complete previous lab exercises.