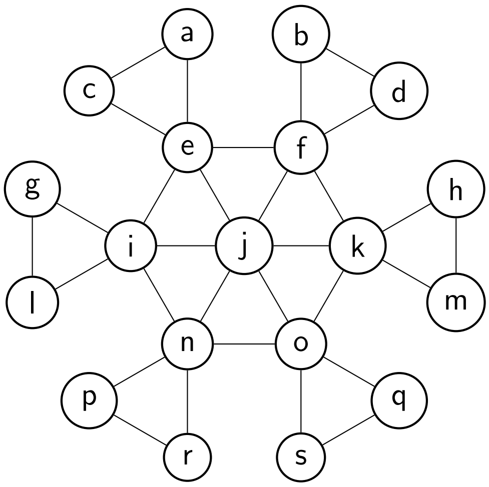
Worksheet BFS and DFS

COMP2721 Algorithms and Data Structures II

1. Execute breadth-first search and depth-first search on the following graph. Start at vertex a and handle neighbours in alphabetical order.



   Mark the edges of the BFS-tree in a drawing of the graph. For each vertex v give its BFS-number $\sigma(v)$ and the vertices in the queue at the time v is dequeued.

   Mark the edges of the DFS-tree in a drawing of the graph. For each vertex v give its DFS-number $\sigma(v)$ and the vertices in the stack, that is, the vertices u for which `DFS-visit`(u) starts before and terminates after `DFS-visit`(v).

2. BFS and DFS produce spanning trees of connected graphs, but these trees depend on the ordering of the vertices and the ordering in which their neighbours are considered. A spanning tree $T = (V, F)$ of a connected graph $G = (V, E)$ is called *BFS-tree* of G if there is an ordering of the vertices in V such that BFS produces that tree T. Similarly, $T = (V, F)$ is called *DFS-tree* of G if V can be ordered such that DFS produces T.

   For every integer $n \geq 4$ describe a connected graph G on n vertices such that there is no spanning tree of G that is both an BFS-tree and a DFS-tree of G, and describe another graph G such that every spanning tree of G is a BFS-tree of G if and only if it is a DFS-tree of G.