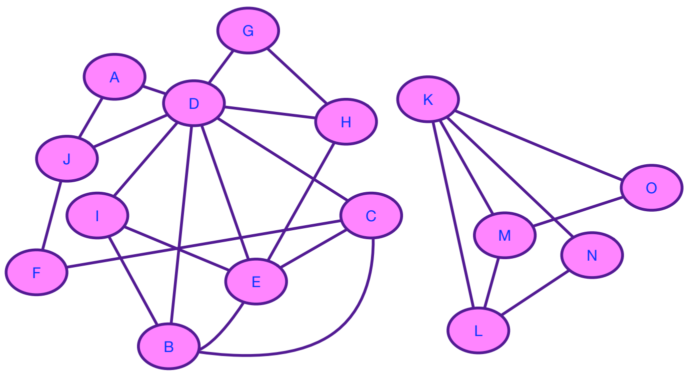
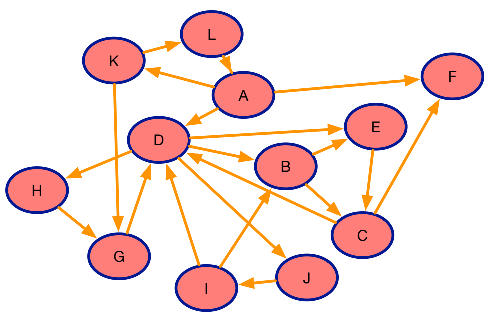
Graph Algorithms

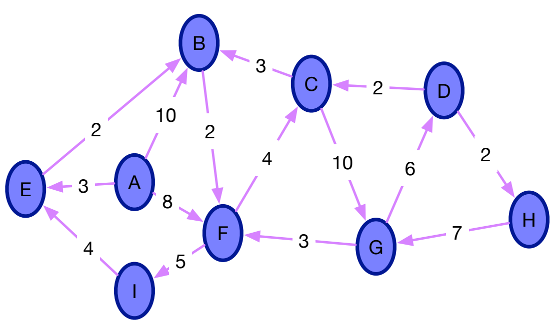
1. Perform the DFS algorithm on the following graph. Give the pre- and postorder numbers for each vertex. For consistency, break ties alphabetically.



2. Perform the DFS algorithm on the following directed graph. Label each edge as a tree edge, forward edge, back edge or cross edge. For consistency, break ties alphabetically.



3. Run Dijkstra's algorithm on the following graph, starting at vertex A.



   (a) Give a table showing the intermediate distance values for all vertices at each iteration of the algorithm.

   (b) Give the final shortest path tree.

4. A *bipartite graph* is a graph $G = (V, E)$ such that V can be partitioned into two sets ($V = V_1 \cup V_2$ and $V_1 \cap V_2 = \emptyset$) such that there are no edges between vertices in the same set.

   (a) Design and analyze an algorithm that determines whether an undirected graph is bipartite.

(b) Prove the following theorem:
An undirected graph, G, is bipartite if and only if it contains no cycles of odd length.

5. Suppose a CS curriculum consists of n courses, all of them mandatory. The prerequisite graph G has a vertex for each course, and an edge from course v to course w if and only if v is a prerequisite for w. Design and analyze an algorithm that works directly with this graph representation, and computes the minimum number of quarters necessary to complete the curriculum (assume that a student can take any number of courses in one quarter). The running time of your algorithm should be linear.

6. There are three containers: a 10 liter container, a 7 liter container, and a 4 liter container. The 7 and 4 liter containers are initially full of water, while the 10 liter container is empty. There is only one operation: pour the contents of one container into the other stopping only when the pouring container is empty or the receiving container is full.

   - Is there a sequence of pourings that leaves exactly 2 liters in the 4 liter container?
   - Model this as a graph problem:
     - Define the graph.
     - What is the problem that needs to be solved?
     - What algorithm can be used to solve this problem?

7. Suppose you are given a source word and a target word. Your goal is to transform the source word to the target word in as few valid steps as possible. The valid steps are change one character of the word to a new character, add one character to the word, or remove one character from the word. After each step, the resulting string must be a valid word. For example, we could transform *cat* to *hut* in two steps:

   cat $\rightarrow$ *hat* $\rightarrow$ *hut*

   Model this problem as a graph. What algorithm can be used to solve this problem?

8. Design and analyze an algorithm for finding an odd length cycle in a strongly connected directed graph.

9. Design and analyze an algorithm which takes as an input an undirected graph, G, and an edge, $e = (u, v)$, and determines whether G has a cycle containing e.

10. Prove that in an undirected graph, $\sum_{u \in V} degree(u) = 2|E|$. Conclude that there must be an even number of vertices with odd degree. Does a similar statement hold for the number of vertices of odd indegree in a directed graph?

11. Prove the following theorem:

    In any connected undirected graph, there is a vertex whose removal leaves the graph connected.

12. Is it always possible to make an undirected graph with two connected components connected by adding a single edge?

    - Why or why not? (Proof or counterexample)
    - Does the same hold true for directed graphs (and strongly connected components rather than connected components)? (Proof or counterexample)

13. Give an example of a strongly connected directed graph $G = (V, E)$ such that, for every $v \in V$, removing v from G leaves a directed graph that is not strongly connected.

14. True or False? (Proof or counterexample) If a depth-first search on a directed graph $G = (V, E)$ produces exactly one back edge, then it is possible to choose an edge $e \in E$ such that the graph without e ($G' = (V, E \setminus \{e\})$) is acyclic.

15. True or False? (Proof or counterexample) If a directed graph $G = (V, E)$ is cyclic but can be made acyclic by removing one edge, then a depth-first search in G will encounter exactly one back edge.

16. Suppose you had a tree with 14 vertices, $A, B, \ldots, N$. You wrote down the pre and post orderings for the vertices:

    Preorder: $A, B, D, E, H, L, M, I, F, C, G, J, K, N$

    Postorder: $D, L, M, H, I, E, F, B, J, N, K, G, C, A$

    Then you lost your graph. Can you reconstruct the tree? If so, do.

17. Can Dijkstra's algorithm handle negative weight edges? Why or why not?