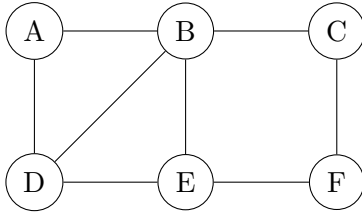


## CS 170 DIS 03

Released on 2018-02-07

### 1 Breadth-First Search



Run breadth-first search on the above graph, breaking ties in alphabetical order (so the search starts from node A). At each step, state which node is processed and the resulting state of the queue. Draw the resulting BFS tree.

### 2 Dijkstra's Algorithm Fails on Negative Edges

Draw a graph with five vertices or fewer, and indicate the source where Dijkstra's algorithm will be started from.

1. Draw a graph with at least one negative weight edge for which Dijkstra's algorithm produces the wrong answer.
2. Draw a graph with at least two negative weight edge for which Dijkstra's algorithm produces the correct answer.

### 3 Fixing Dijkstra's Algorithm with Negative Weights

Dijkstra's algorithm doesn't work on graphs with negative edge weights. Here is one attempt to fix it:

1. Add a large number  $M$  to every edge so that there are no negative weights left.
2. Run Dijkstra to find the shortest path in the new graph.
3. Return the path Dijkstra found, but with the old edge weights (i.e. subtract  $M$  from the weight of each edge).

Show that this algorithm doesn't work by finding a graph for which it must give the wrong answer.

### 4 Bounded Bellman-Ford

Modify the Bellman-Ford algorithm to find the weight of the lowest-weight path from  $s$  to  $t$  with the restriction that the path must have at most  $k$  edges.