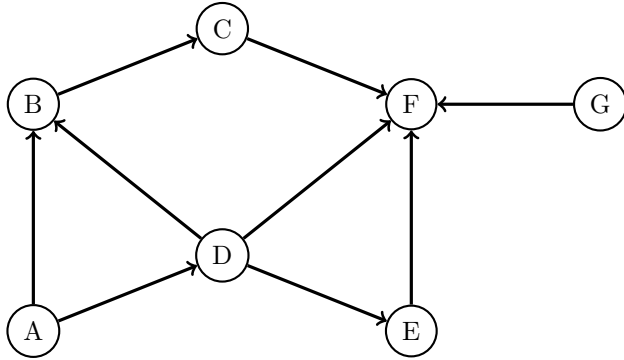


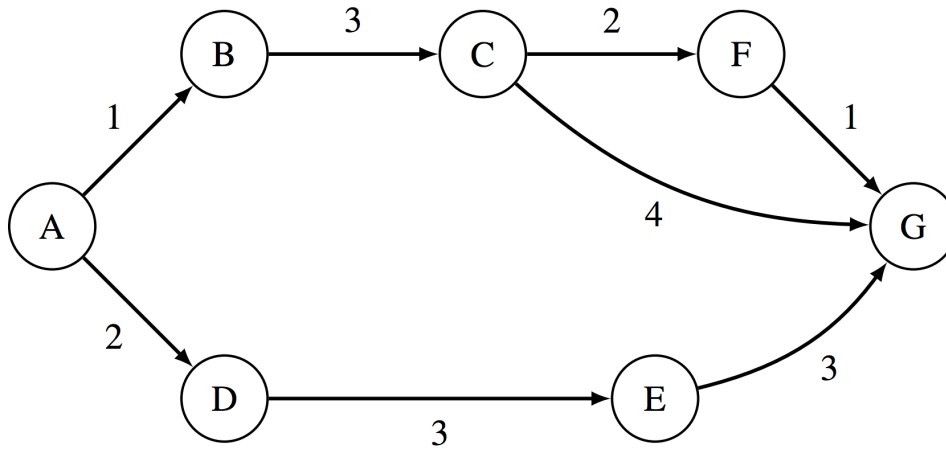
1 Graph Representations



- (a) Write the graph above as an adjacency matrix, then as an adjacency list. What would be different if the graph were undirected instead?
- (b) Give the DFS preorder, DFS postorder, and BFS order of the graph traversals starting from vertex *A*. Break ties alphabetically.

2 Dijkstra's Algorithm

For the graph below, let $g(u, v)$ be the weight of the edge between any nodes u and v . Let $h(u, v)$ be the value returned by the heuristic for any nodes u and v .



Edge weights	Heuristics
$g(A, B) = 1$	$h(A, G) = 8$
$g(B, C) = 3$	$h(B, G) = 6$
$g(C, F) = 4$	$h(C, G) = 5$
$g(C, G) = 4$	$h(F, G) = 1$
$g(F, G) = 1$	$h(D, G) = 6$
$g(A, D) = 2$	$h(E, G) = 3$
$g(D, E) = 3$	
$g(E, G) = 3$	

- (a) Run Dijkstra's algorithm to find the shortest paths from A to every other vertex. You may find it helpful to keep track of the priority queue and make a table of current distances.

Pseudocode

```

1  PQ = new PriorityQueue()
2  PQ.add(A, 0)
3  PQ.add(v, infinity) # (all nodes except A).
4
5  distTo = {} # map
6  distTo[A] = 0
7  distTo[v] = infinity # (all nodes except A).
8
9  while (not PQ.isEmpty()):
10     poppedNode, poppedPriority = PQ.pop()
11
12     for child in poppedNode.children:
13         if PQ.contains(child):
14             potentialDist = distTo[poppedNode] + edgeWeight(poppedNode, child)
15             if potentialDist < distTo[child]:
16                 distTo.put(child, potentialDist)
17                 PQ.changePriority(child, potentialDist)
  
```

- (b) Given the weights and heuristic values for the graph below, what path would A* search return, starting from A and with G as a goal?

Pseudocode

```

1 PQ = new PriorityQueue()
2 PQ.add(A, h(A))
3 PQ.add(v, infinity) # (all nodes except A).
4
5 distTo = {} # map
6 distTo[A] = 0
7 distTo[v] = infinity # (all nodes except A).
8
9 while (not PQ.isEmpty()):
10     poppedNode, poppedPriority = PQ.pop()
11     if (poppedNode == goal): terminate
12
13     for child in poppedNode.children:
14         if PQ.contains(child):
15             potentialDist = distTo[poppedNode] + edgeWeight(poppedNode, child)
16
17             if potentialDist < distTo[child]:
18                 distTo.put(child, potentialDist)
19                 PQ.changePriority(child, potentialDist + h(child))

```

- (c) Is the heuristic admissible? Why or why not?