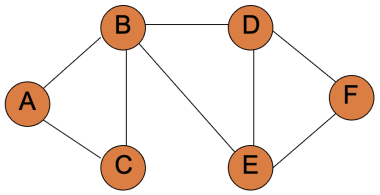
CS62: Spring 2026: Lecture #21 worksheet

Fill in the values below for running BFS

```
q.addLast(start);
visited[start] = true;
distTo[start] = 0;

while( !q.isEmpty() ) {
    int v = q.removeFirst();

    for( int adj: g.adj(v) ) {
        if( !visited[adj] ) {
            visited[adj] = true;
            edgeTo[adj] = v;
            distTo[adj] = distTo[v] + 1;
            q.addLast(adj);
        }
    }
}
```



Queue:

	A	B	C	D	E	F
visited:						
edgeTo						
distTo						

What is the shortest path from A to F?

Fill in the value below for running Dijkstra's

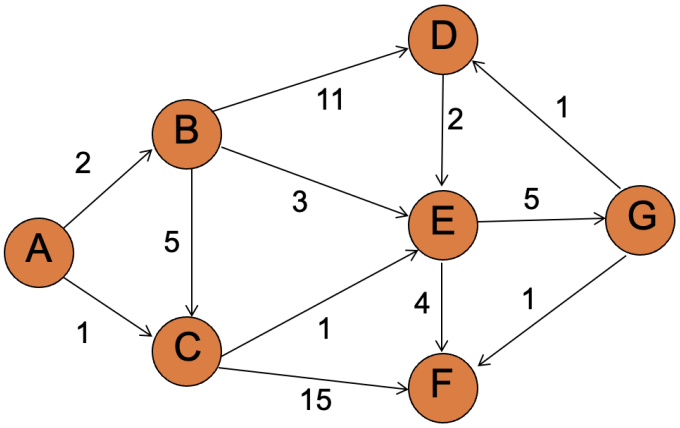
PQ

```
distTo[start] = 0.0;
pq.decreaseKey(start, 0.0);

while( !pq.isEmpty() ) {
    int v = pq.delMin();

    for (WeightedEdge e : g.adj(v)) {
        int adj = e.to();

        if( distTo[v] + e.weight() < distTo[adj] ) {
            distTo[adj] = distTo[v] + e.weight();
            edgeTo[adj] = v;
            pq.decreaseKey(adj, distTo[adj]);
        }
    }
}
```



visited:

A	B	C	D	E	F	G

edgeTo

A	B	C	D	E	F	G

distTo

What is the shortest path from A to D?