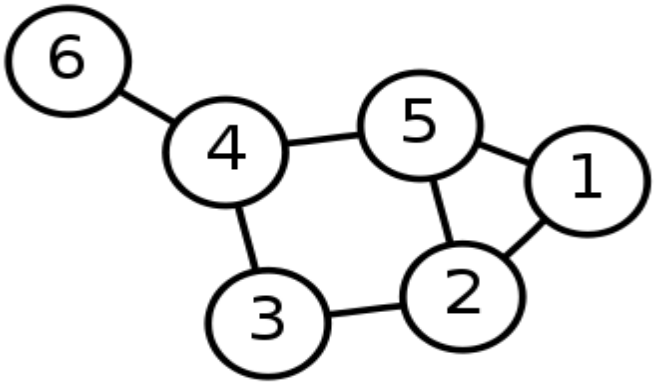
Lecture Six

Topics that must be covered in this lecture:

- Single-Source Shortest Path Problem
- Heuristic search: (Dijkstra's algorithm)
- Dijkstra's algorithm – Pseudocode
- Dijkstra algorithm Example1
- Dijkstra algorithm Example2
- Dijkstra algorithm Example3

Single-Source Shortest Path Problem: - The problem of finding shortest paths from a source vertex *v* to all other vertices in the graph.



Dijkstra's algorithm - is a solution to the single-source shortest path problem in graph theory.

Works on both directed and undirected graphs. However, all edges must have nonnegative weights.

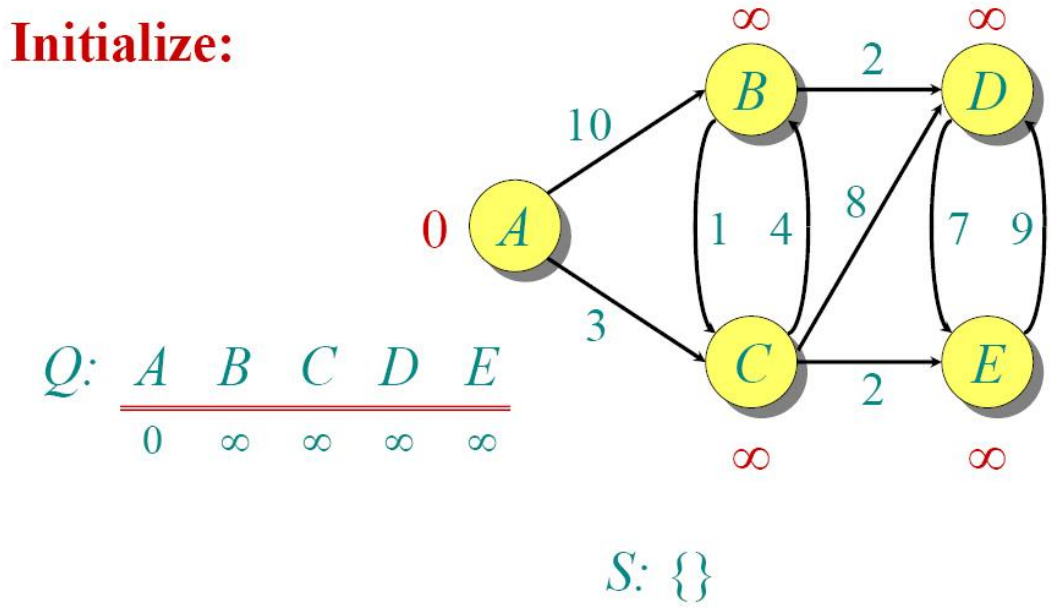
Input: A weighted graph G = {E, V} and source vertex $v \in V$, such that all edge weights are nonnegative

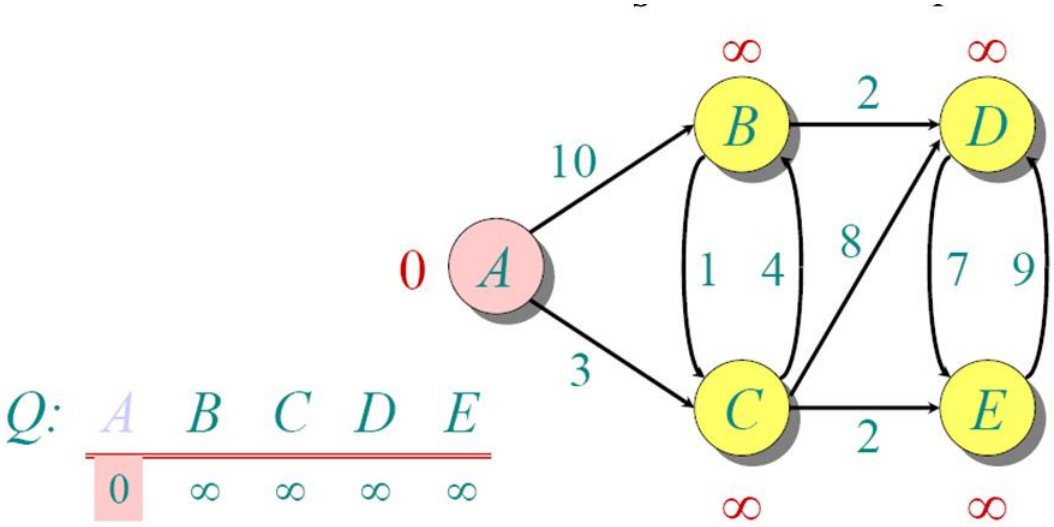
Output: Lengths of shortest paths (or the shortest paths themselves) from a given source vertex $v \in V$ to all other vertices

Dijkstra's algorithm – Pseudocode

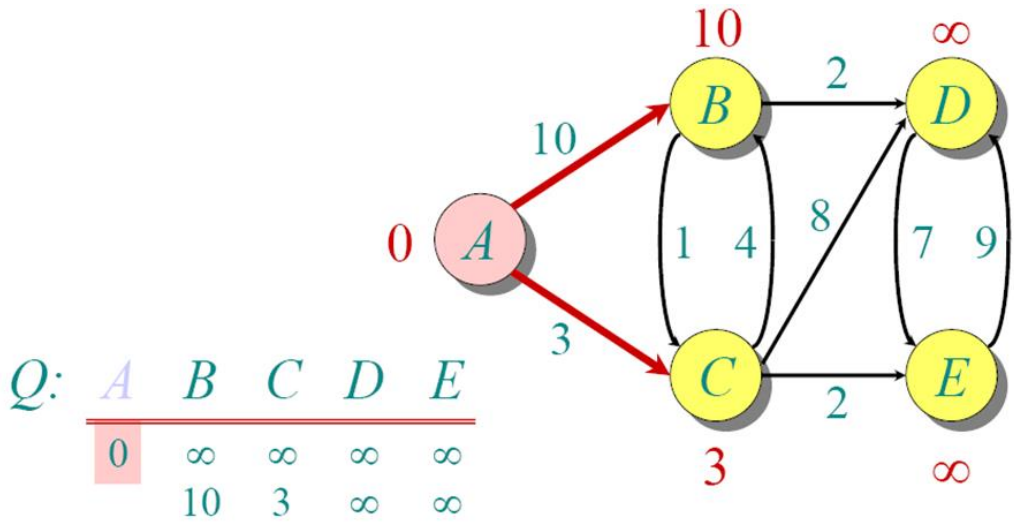
```
dist[s] ←o                          (distance to source vertex is zero)
for all v ∈ V–{s}
     do dist[v] ←∞                  (set all other distances to infinity)
S←∅                                 (S, the set of visited vertices is initially empty)
Q←V                                 (Q, the queue initially contains all vertices)
while Q ≠∅                          (while the queue is not empty)
do  u ← mindistance(Q,dist)         (select the element of Q with the min. distance)
    S←S∪{u}                         (add u to list of visited vertices)
    for all v ∈ neighbors[u]
        do if  dist[v] > dist[u] + w(u, v)        (if new shortest path found)
              then    d[v] ←d[u] + w(u, v)        (set new value of shortest path)
                (if desired, add traceback code)
return dist
```

Dijkstra algorithm Example1:



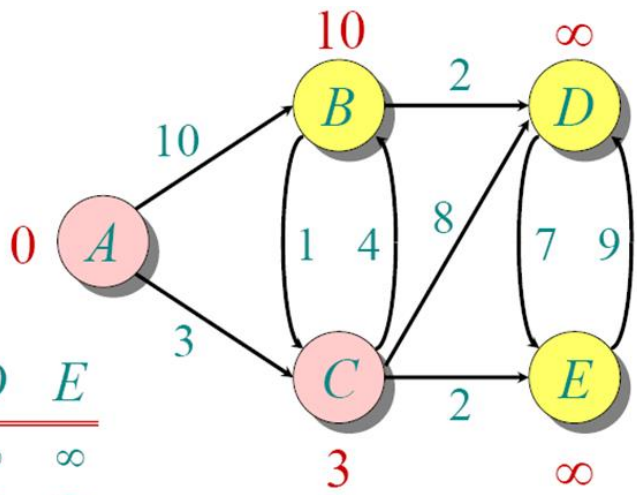


Q:	A	B	C	D	E
	0	∞	∞	∞	∞



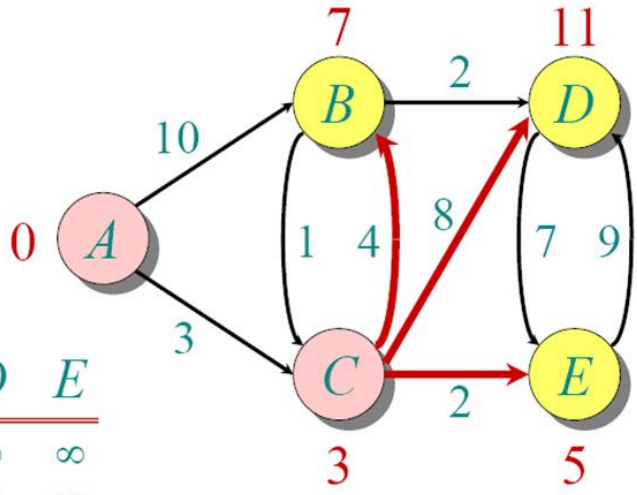
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞

S: { A }



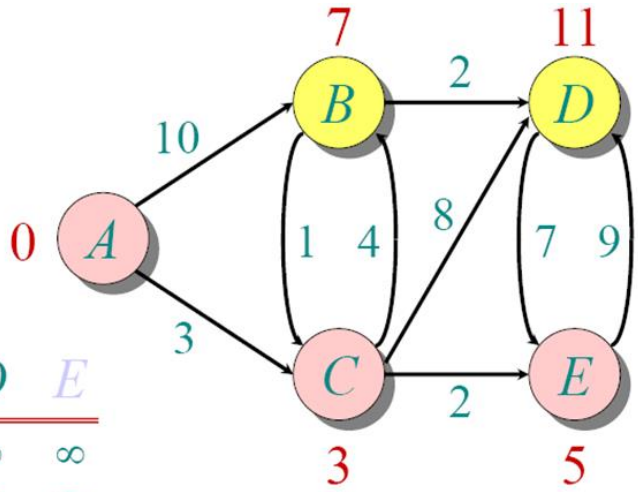
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞

S: { A, C }



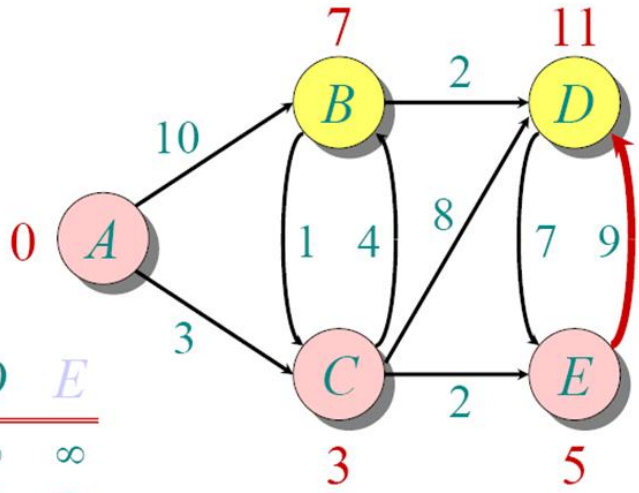
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞
		7		11	5

S: { A, C }



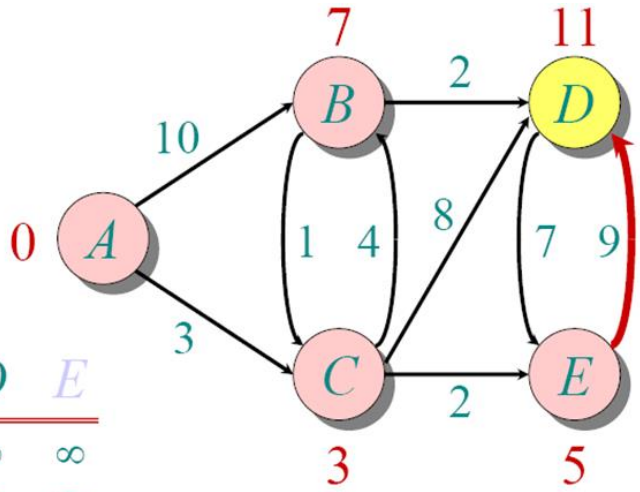
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞
		7		11	5

$S: \{ A, C, E \}$



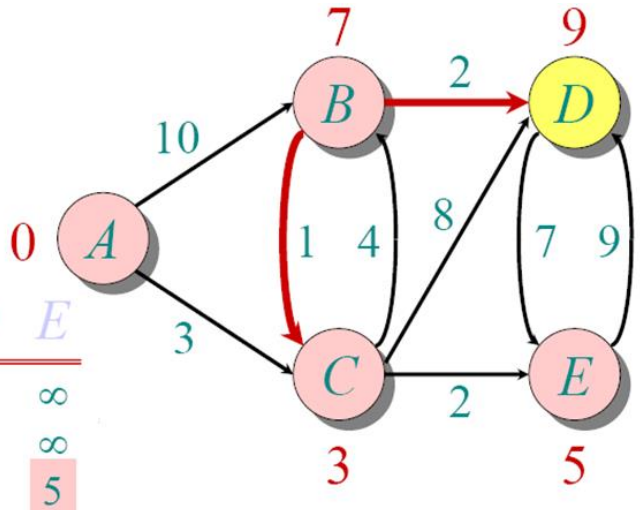
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞
		7		11	5
		7		11	

$S: \{ A, C, E \}$



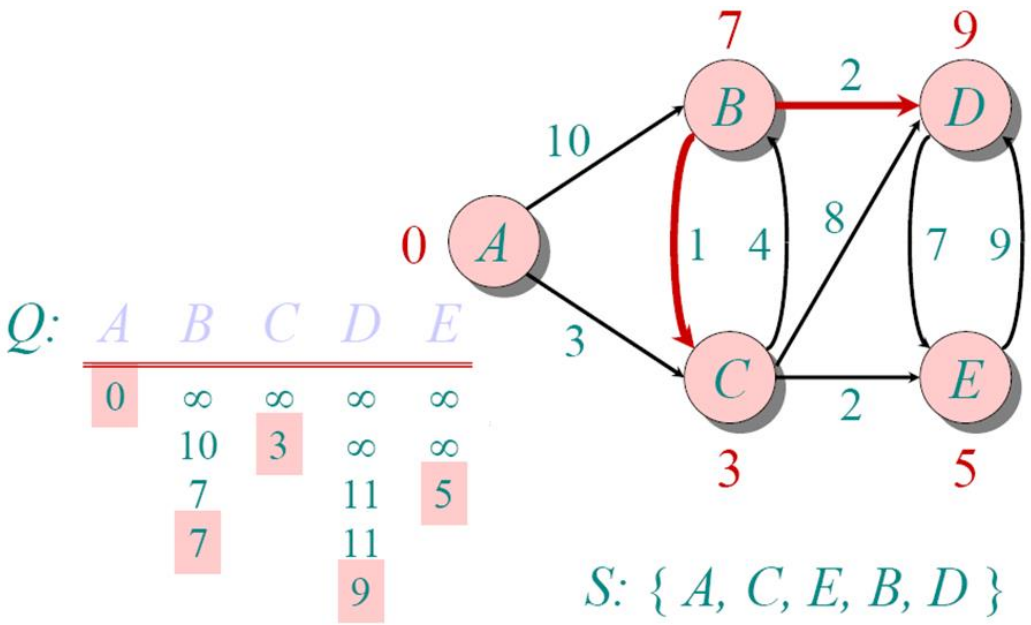
Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞
		7		11	5
		7		11	

S: { A, C, E, B }

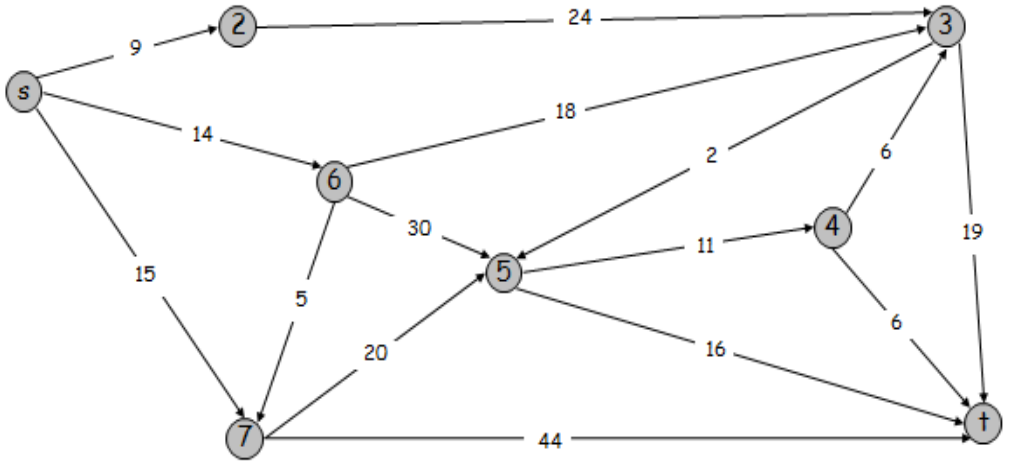


Q:	A	B	C	D	E
	0	∞	∞	∞	∞
		10	3	∞	∞
		7		11	5
		7		11	
				9	

S: { A, C, E, B }



Dijkstra algorithm Example 2: Find the shortest path from s to t.



Dijkstra algorithm Example 3: For a given source vertex (node) in the graph, the algorithm can be used to find the shortest path from a single starting vertex to a single destination vertex.

For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city (a) and the destination city (b).