

# Lecture 13

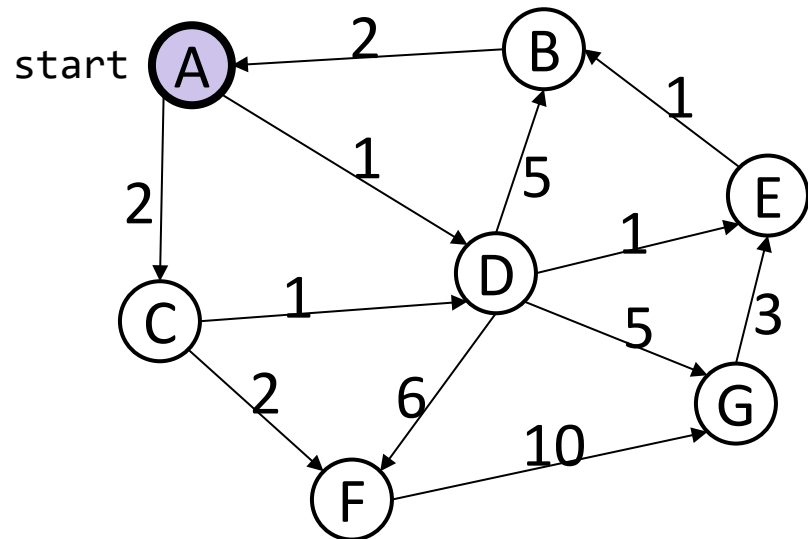
## Shortest Paths

### Exercises

Department of Computer Science  
Hofstra University

# Q. Dijkstra's Algorithm

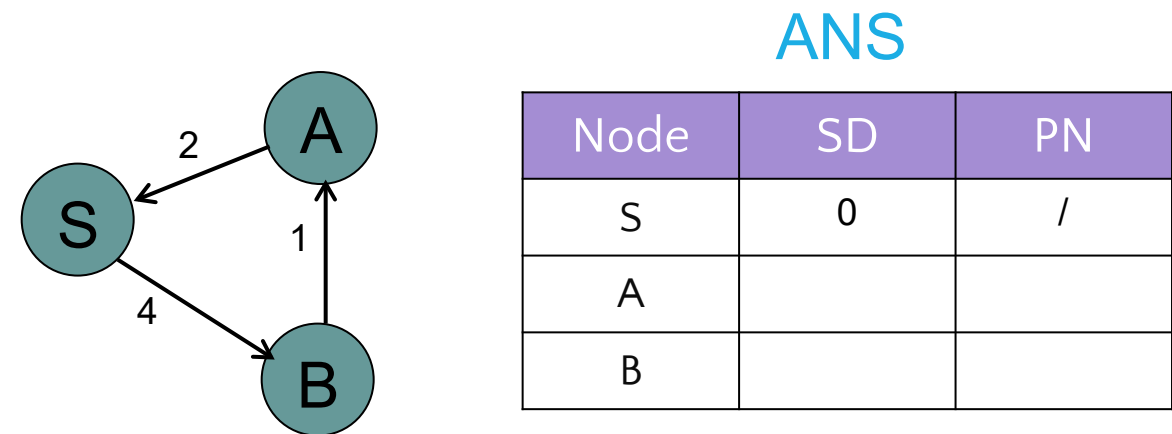
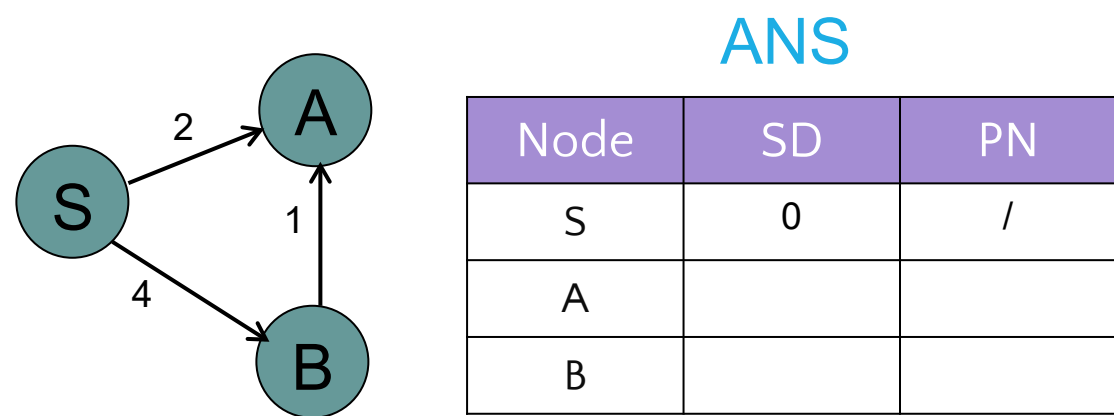
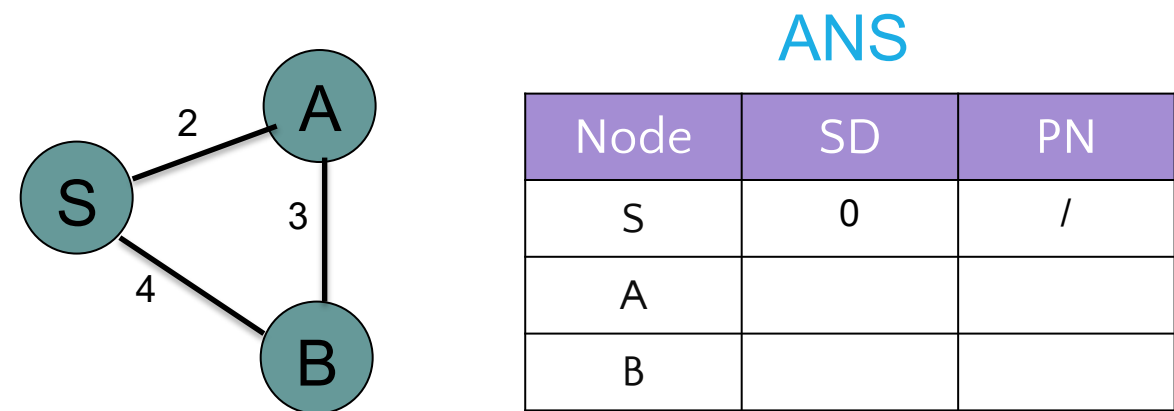
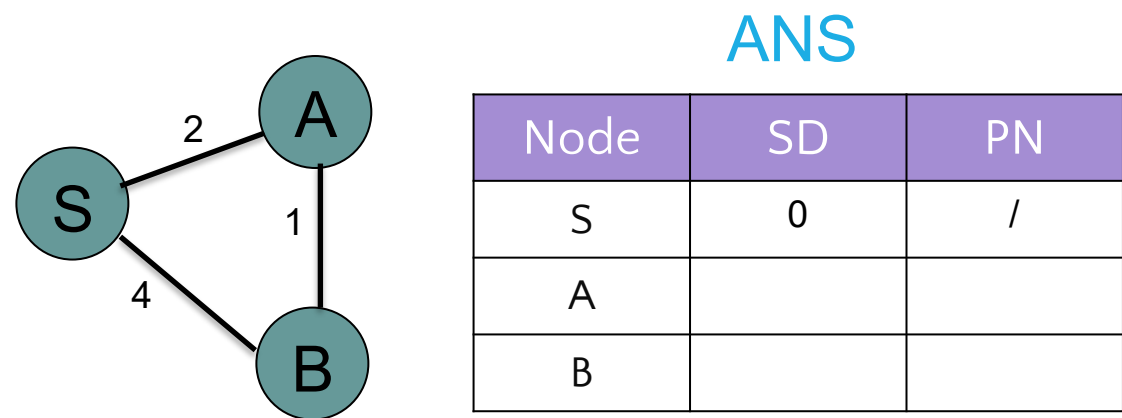
*Exam question:* Given this directed graph, run Dijkstra's Algo to find shortest paths starting from **source node A**. Give the node visit order, and fill in this table of SN (Shortest Distance) and PN (Previous Node), crossing out old SD and PN as you find a shortcut path with smaller SD



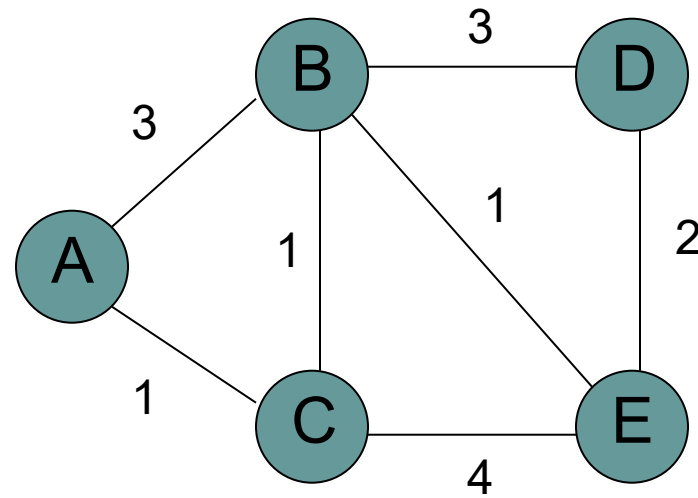
Visit Order

Node	SD	PN
A		
B		
C		
D		
E		
F		
G		

# Q. Dijkstra's Algorithm (Source Node S)



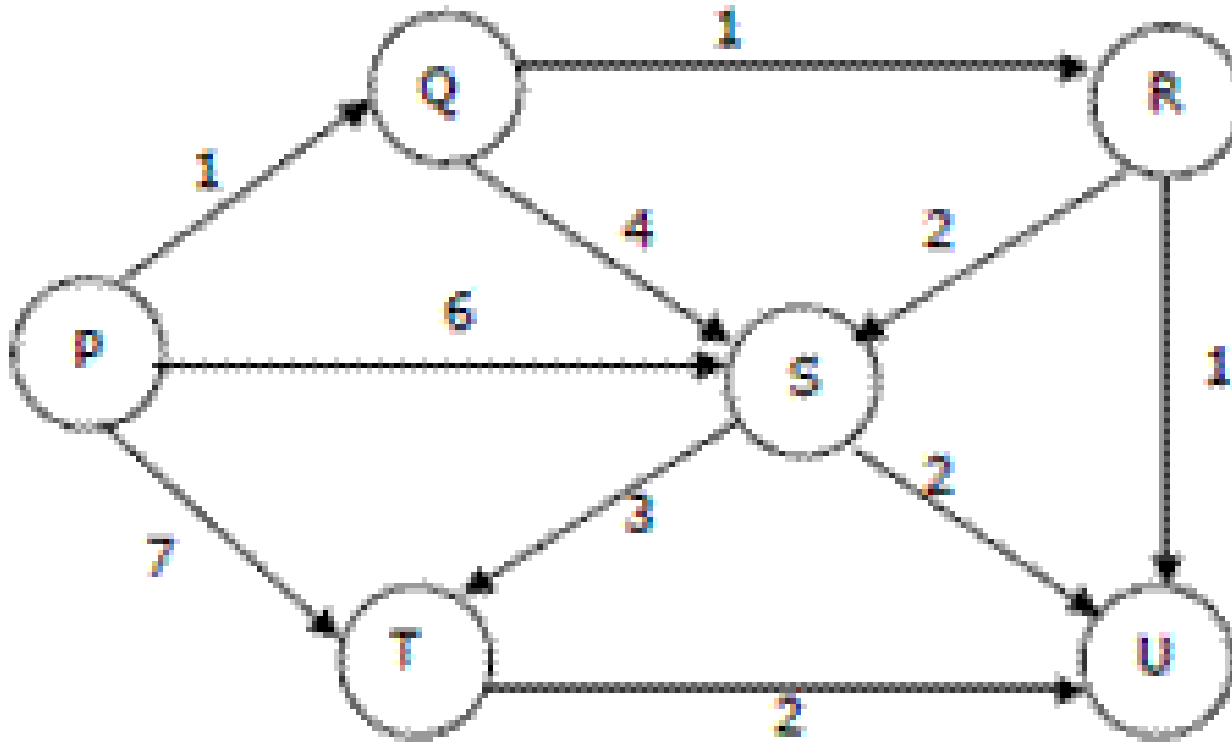
# Q. Dijkstra's Algorithm (Source Node A, Undirected Graph)



Visit Order

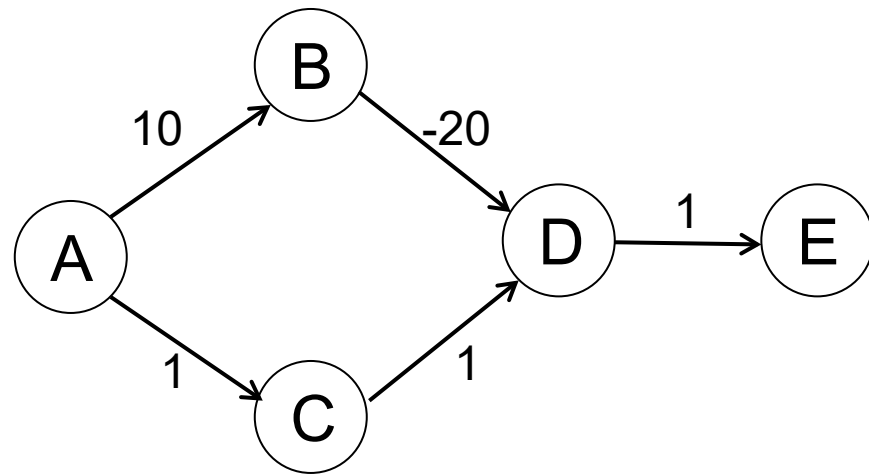
Node	SD	PN
A		
B		
C		
D		
E		

# Q. Dijkstra's Algorithm (Source Node P, Directed Graph)



# Q. Topological Sort

Given this directed graph, run Topological Sort to find shortest paths starting from **source node A**. Give the node visit order, and fill in this table of SN (Shortest Distance) and PN (Previous Node), crossing out old SD and PN as you find a shortcut path with smaller SD. Considering **all possible** topological orders.



Visit Order

Node	SD	PN
A	0	/
B		
C		
D		
E		

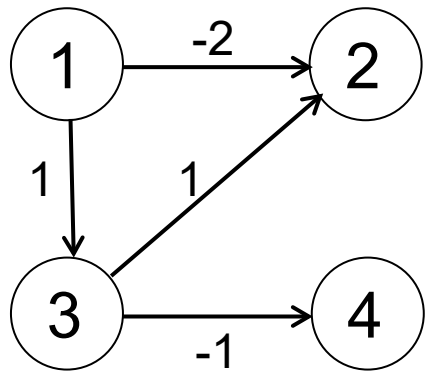
# Q. Johnson's algorithm

Consider the following weighted digraph. As part of Johnson's algorithm for All-pairs Shortest Paths, add a dummy source node  $d$ , and edges with weight 0 from  $d$  to all vertices of  $G$ . Let the modified graph be  $G'$ .

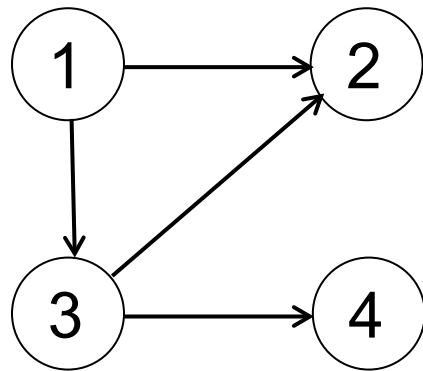
a) Compute the shortest distances from dummy source node  $d$  to each node in  $G'$  by hand:  $h[0]$ ,  $h[1]$ , ..  $h[V-1]$ , then reweight the edges of the original graph to make the edge weights greater than or equal to 0. Draw the reweighted graph  $G'$  (without the dummy node  $d$ ).

b) For the reweighted graph  $G'$ : run Dijkstra's Algo to find shortest paths starting from **source node 1**, and compute the shortest paths for the graph with updated positive or zero weights. (Do not show the intermediate steps.)

c) For the original graph  $G$ : compute the shortest paths starting from **source node 1** with negative weights.



Original graph



Reweighted graph

Node	$h()$
1	
2	
3	
4	

Shortest paths starting from dummy node

Node	SD	PN
1	0	/
2		
3		
4		

Shortest paths starting from source node 1 in reweighted graph

Node	SD	PN
1	0	/
2		
3		
4		

Shortest paths starting from source node 1 in original graph