

Lecture 13

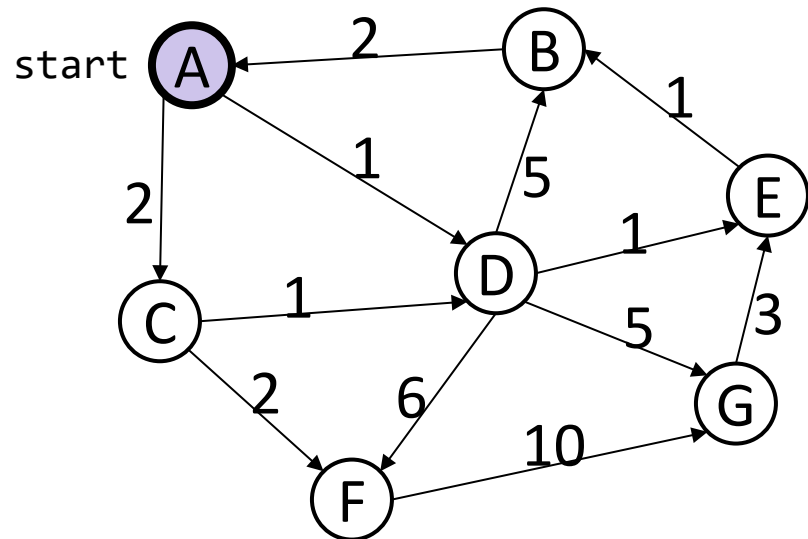
Shortest Paths

Exercises ANS

Department of Computer Science
Hofstra University

Q. Dijkstra's Algorithm

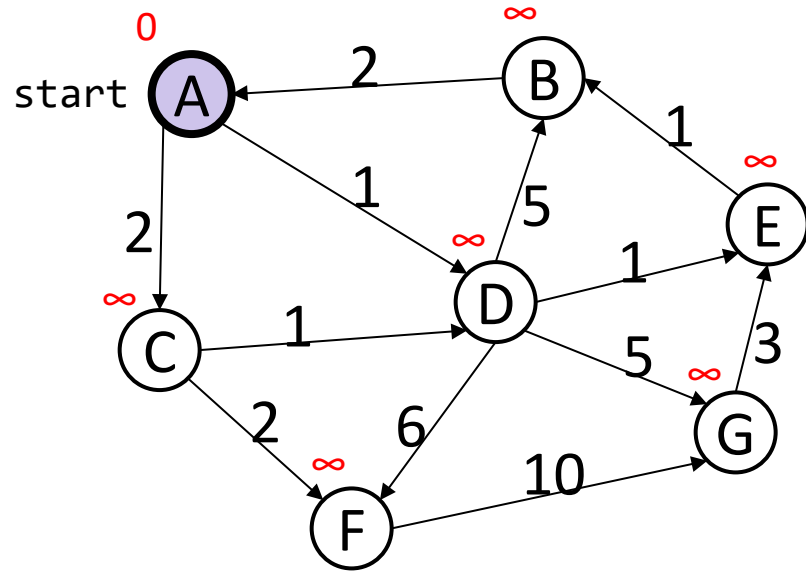
Exam question: Given this directed graph, run Dijkstra's Algo to find shortest paths starting from **source node A**. Give the node visit order, and fill in this table of SN (Shortest Distance) and PN (Previous Node), crossing out old SD and PN as you find a shortcut path with smaller SD



Visit Order

Node	SD	PN
A		
B		
C		
D		
E		
F		
G		

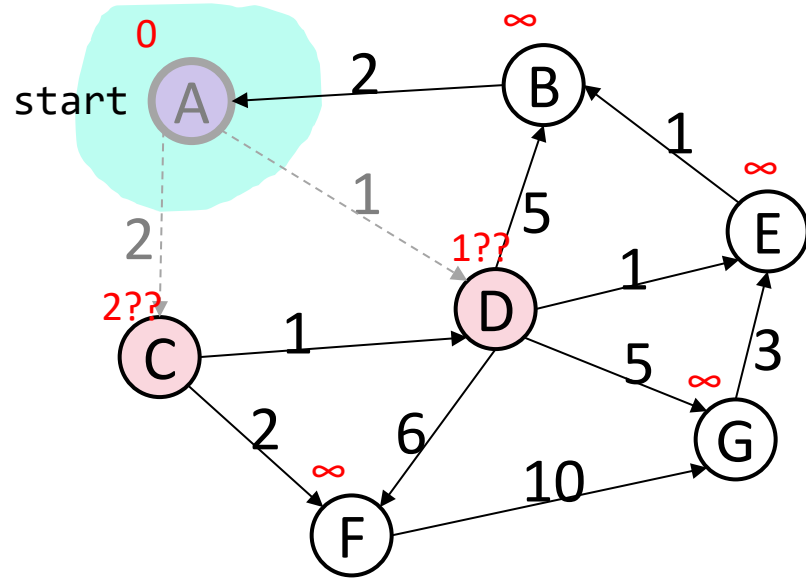
Q. Dijkstra's Algorithm



Visit Order

Node	SD	PN
A	∞	
B	∞	
C	∞	
D	∞	
E	∞	
F	∞	
G	∞	

Q. Dijkstra's Algorithm

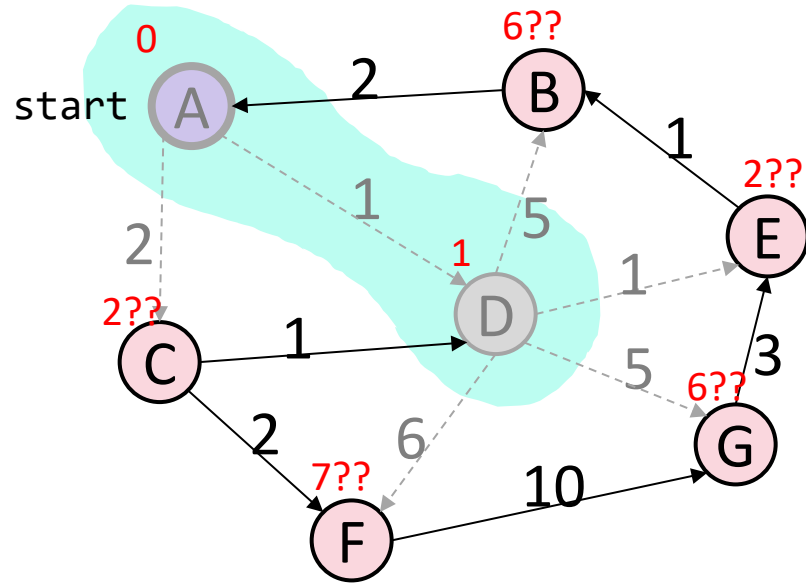


Visit Order

A

Node	SD	PN
A	0	/
B	∞	
C	2	A
D	1	A
E	∞	
F	∞	
G	∞	

Q. Dijkstra's Algorithm



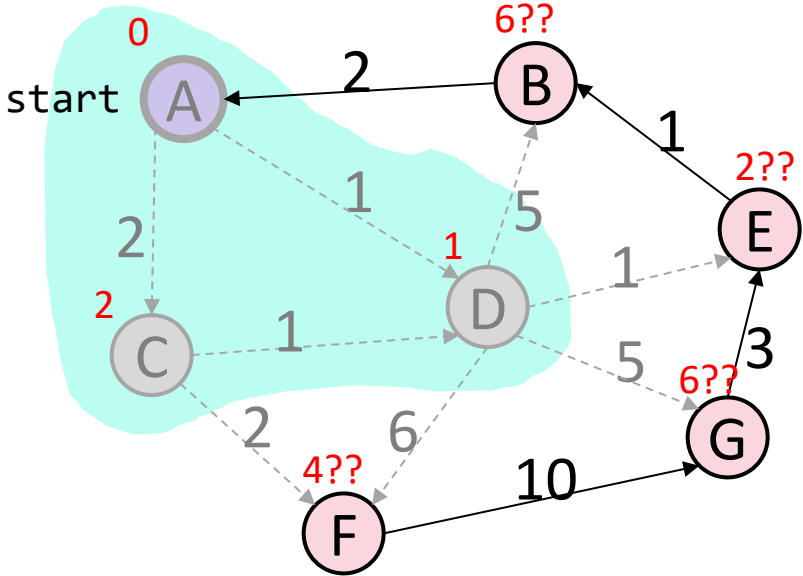
Visit Order

A, D

We can choose to visit either C or E next, since they have equal smallest SD of 2 among all unvisited nodes. Let's visit C in alphabetical order

Node	SD	PN
A	0	/
B	6	D
C	2	A
D	1	A
E	2	D
F	7	D
G	6	D

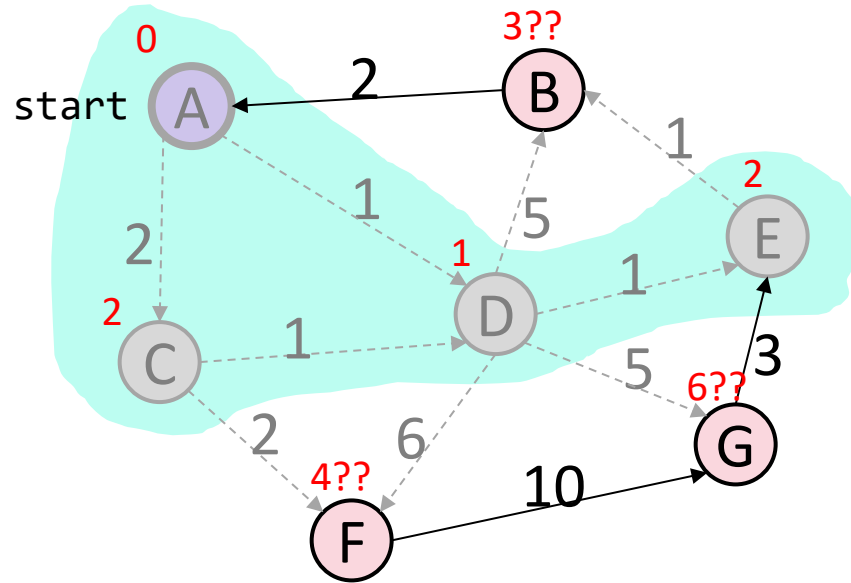
Q. Dijkstra's Algorithm



<u>Visit Order</u>
A, D, C

Node	SD	PN
A	0	/
B	6	D
C	2	A
D	1	A
E	2	D
F	7 4	D C
G	6	D

Q. Dijkstra's Algorithm

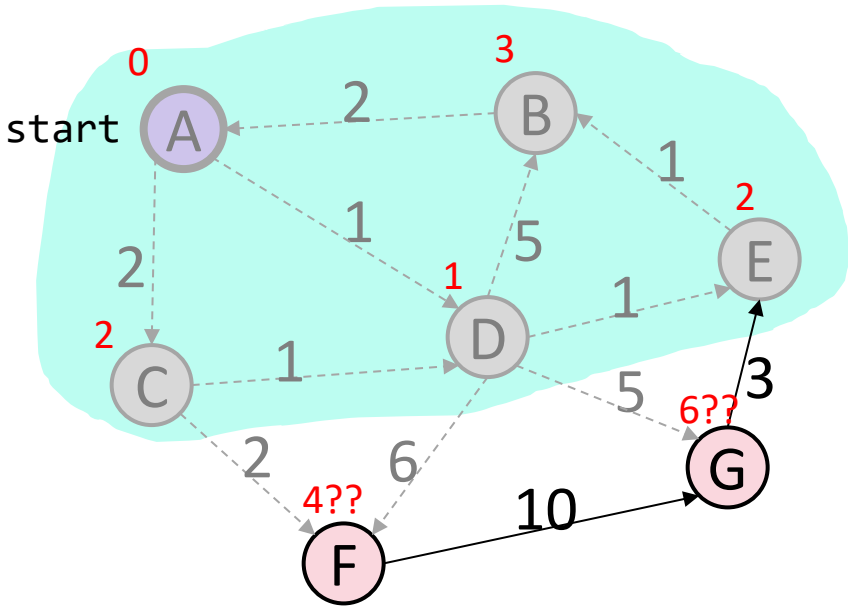


Visit Order

A, D, C, E

Node	SD	PN
A	0	/
B	6 3	D E
C	2	A
D	1	A
E	2	D
F	7 4	D C
G	6	D

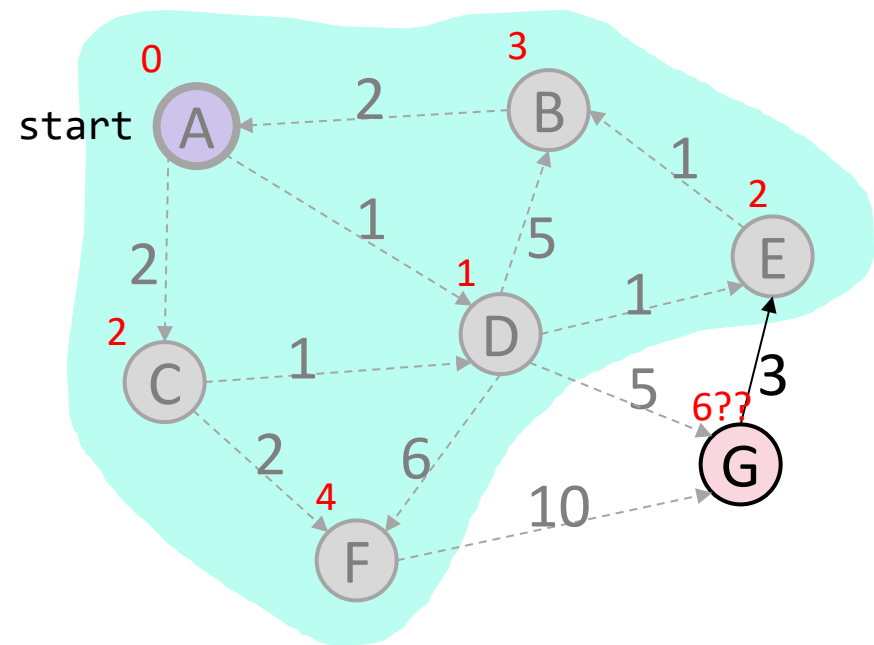
Q. Dijkstra's Algorithm



<u>Visit Order</u>
A, D, C, E, B

Node	SD	PN
A	0	/
B	3	D
C	2	A
D	1	A
E	2	D
F	4	C
G	6	D

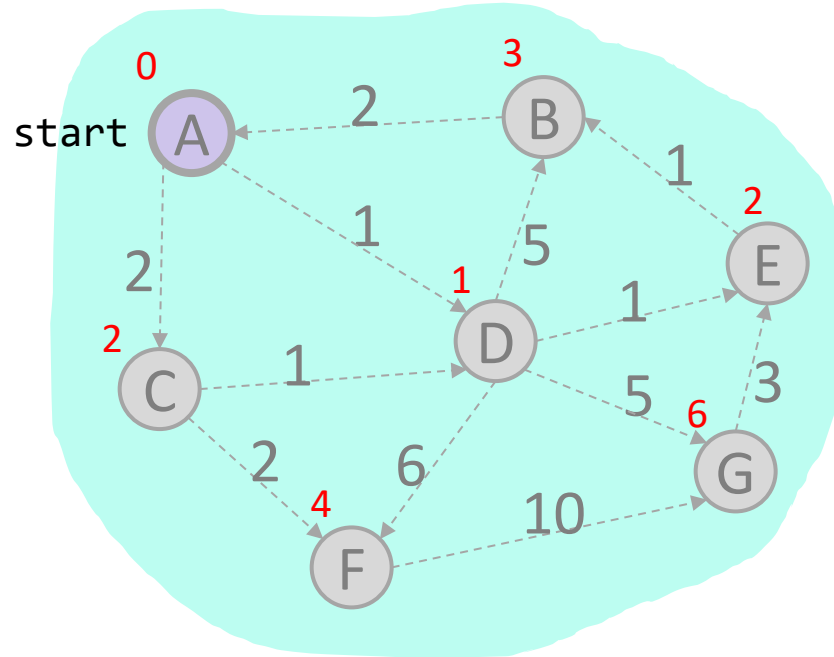
Q. Dijkstra's Algorithm



Visit Order
A, D, C, E, B, F

Node	SD	PN
A	0	/
B	3	D
C	2	A
D	1	A
E	2	D
F	4	C
G	6	D

Q. Dijkstra's Algorithm Final Answer

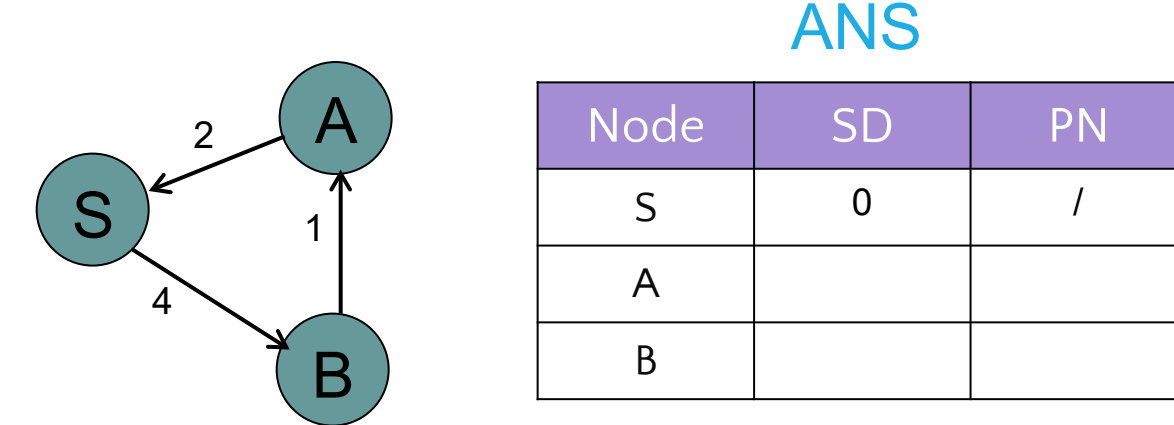
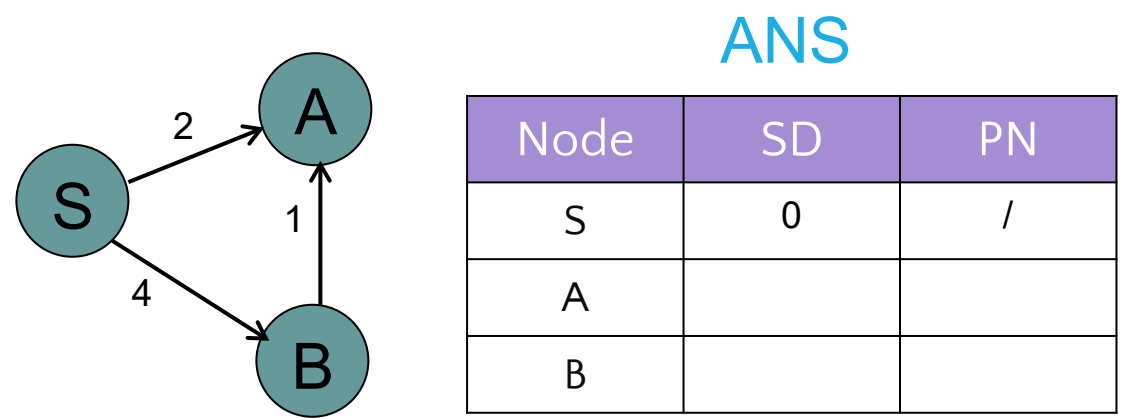
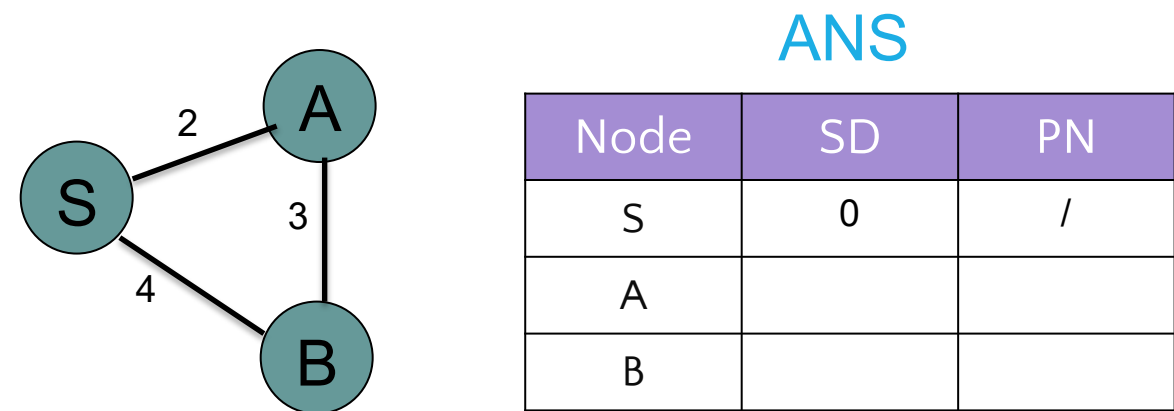
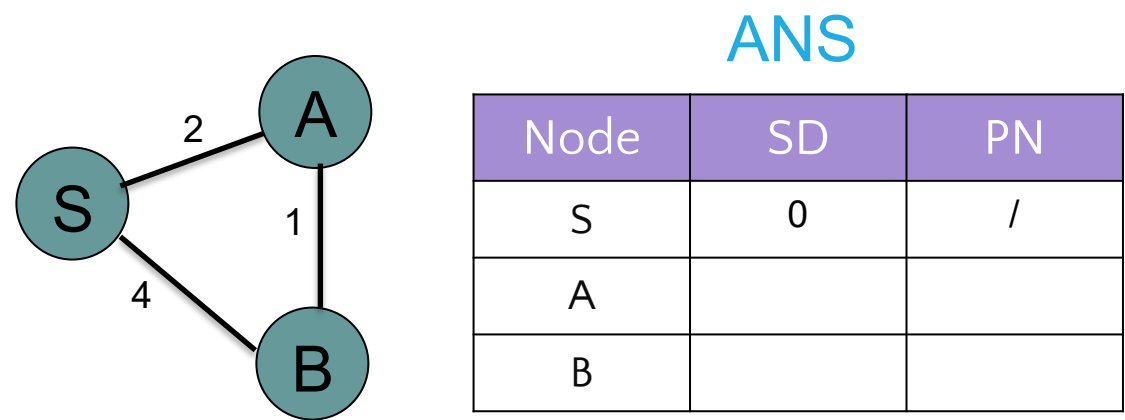


Visit Order

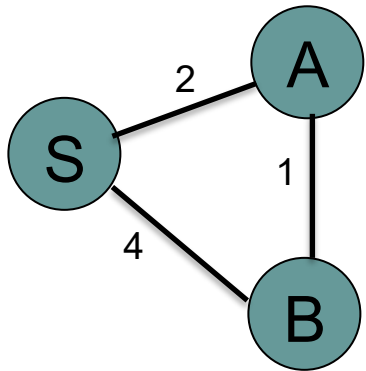
A, D, C, E, B, F, G

Node	SD	PN
A	0	/
B	3	E
C	2	A
D	1	A
E	2	D
F	4	C
G	6	D

Q. Dijkstra's Algorithm (Source Node S)

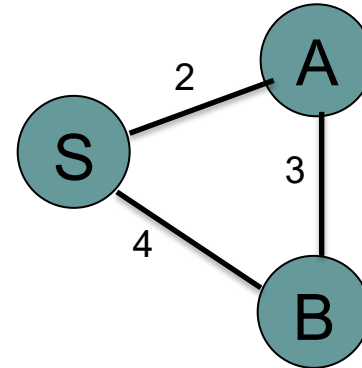


Q. Dijkstra's Algorithm (Source Node S) Final Answer



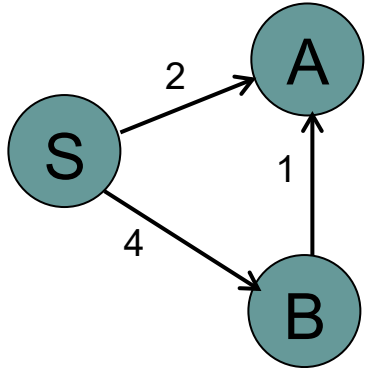
ANS

Node	SD	PN
S	0	/
A	2	S
B	4	S



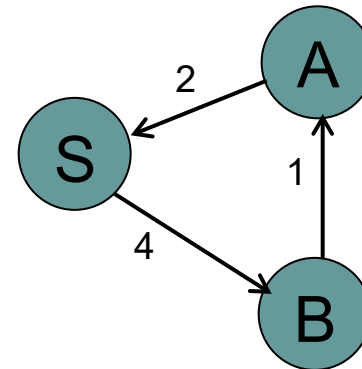
ANS

Node	SD	PN
S	0	/
A	2	S
B	4	S



ANS

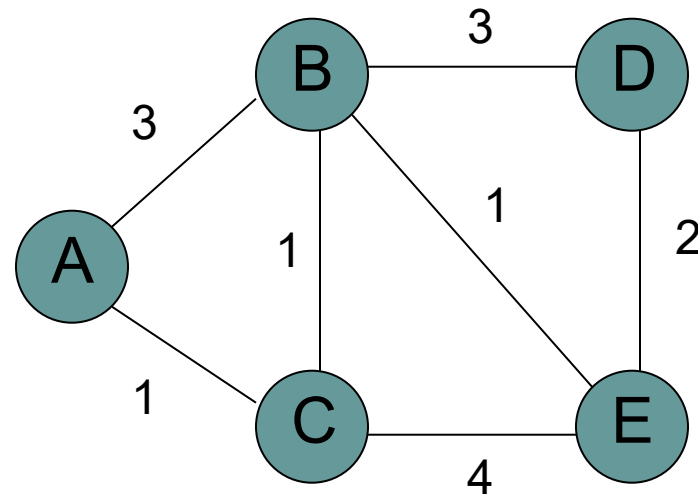
Node	SD	PN
S	0	/
A	2	S
B	4	S



ANS

Node	SD	PN
S	0	/
A	5	B
B	4	S

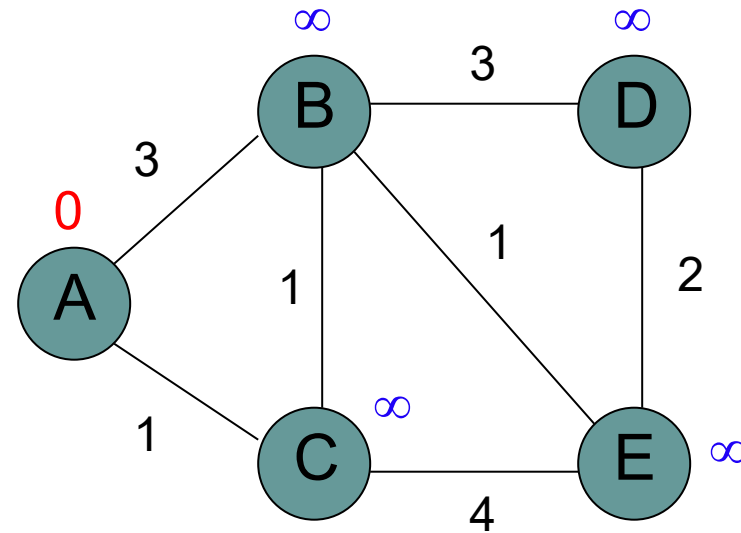
Q. Dijkstra's Algorithm (Source Node A, Undirected Graph)



Visit Order

Node	SD	PN
A		
B		
C		
D		
E		

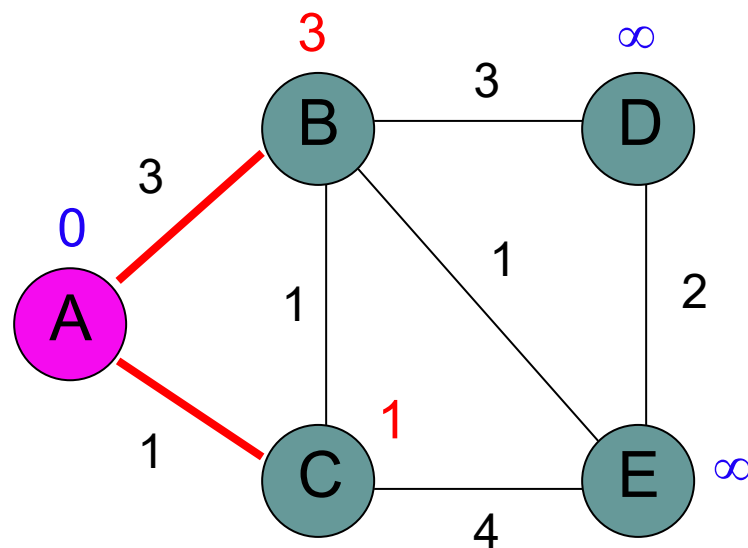
Initialize



Visit Order

Node	SD	PN
A	0	/
B	∞	
C	∞	
D	∞	
E	∞	

Visit Node A

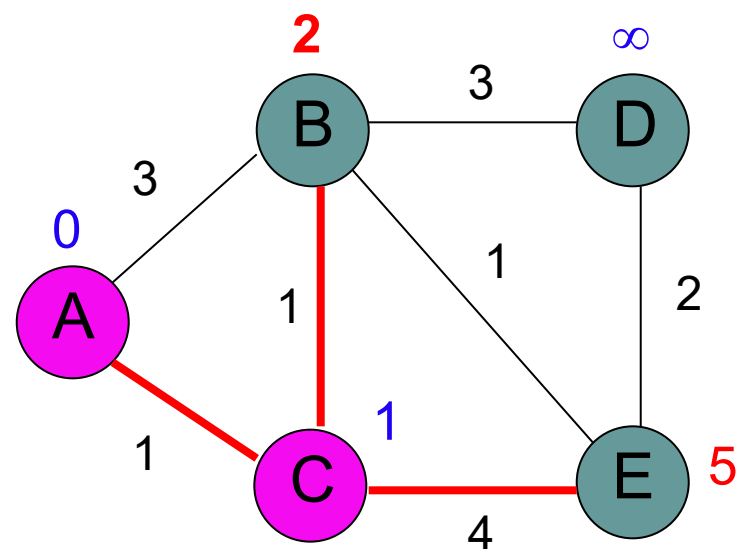


Visit Order

A

Node	SD	PN
A	0	/
B	3	A
C	1	A
D	∞	
E	∞	

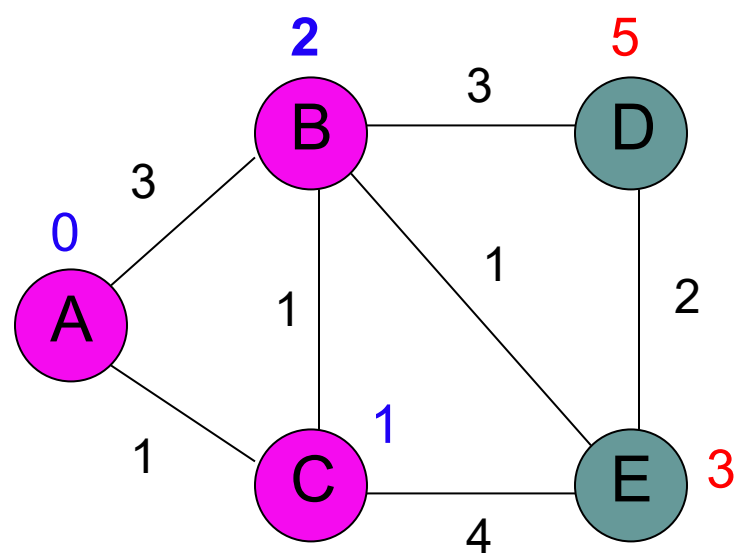
Visit Node C



Visit Order
A, C

Node	SD	PN
A	0	/
B	3 2	A C
C	1	A
D	∞	
E	5	C

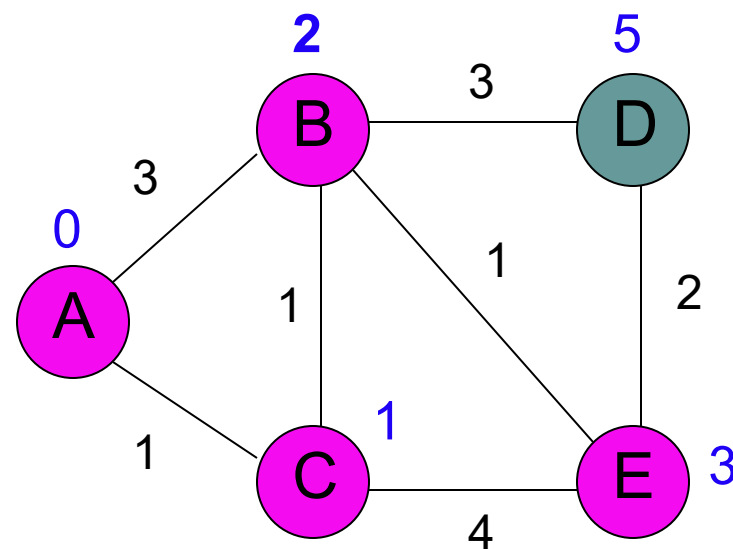
Visit Node B



Visit Order
A, C, B

Node	SD	PN
A	0	/
B	3 2	A C
C	1	A
D	5	B
E	5 3	∈ B

Visit Node E

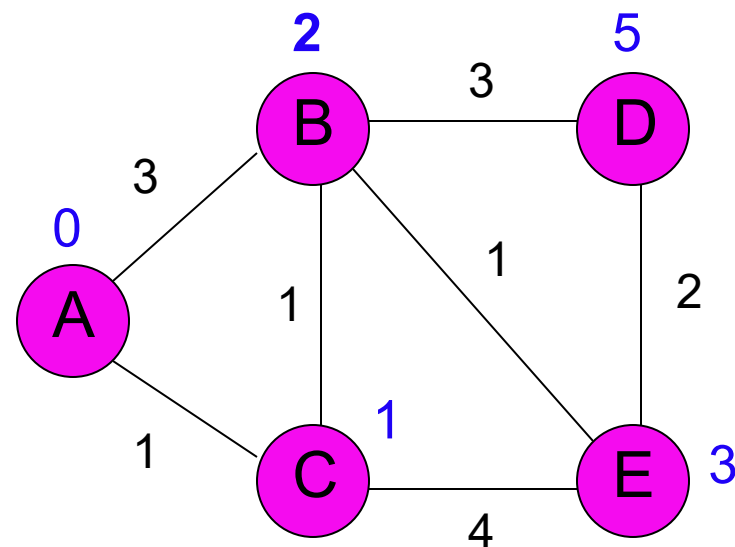


Visit Order
A, C, B, E

Node	SD	PN
A	0	/
B	3 2	A C
C	1	A
D	5	B
E	5 3	∈ B

Nothing changes

Visit Node D

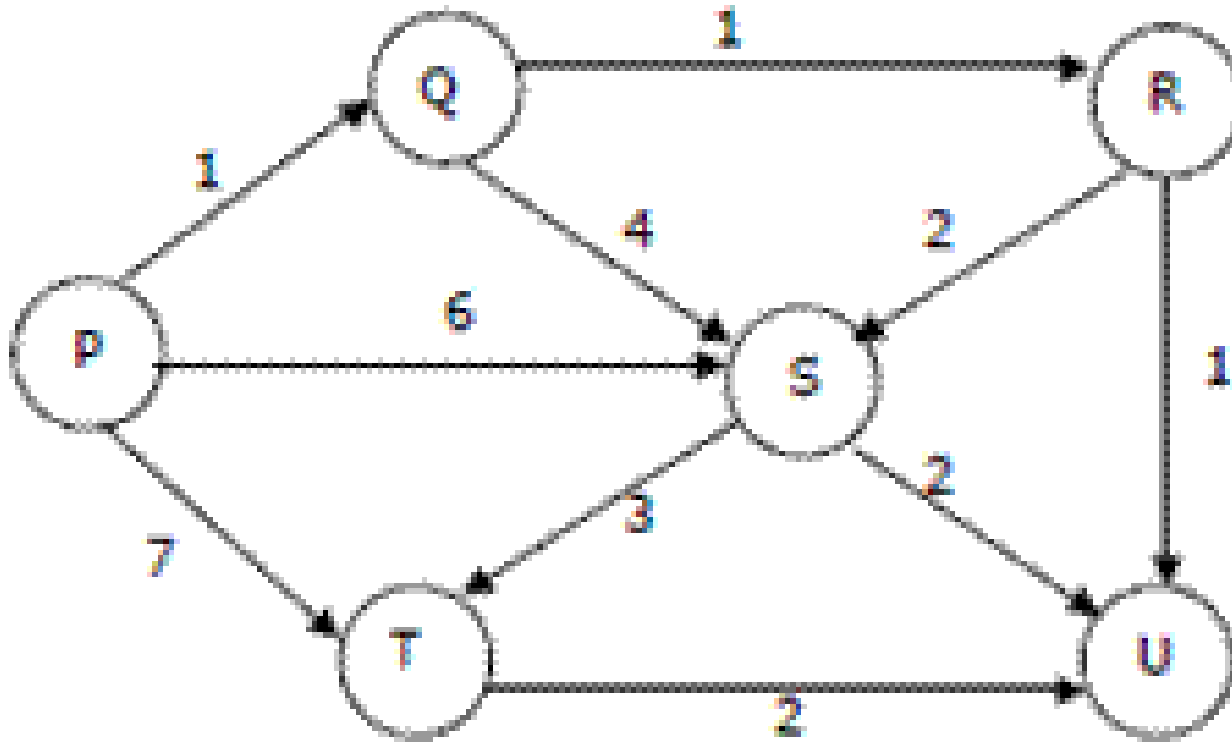


Visit Order
A, C, B, E, D

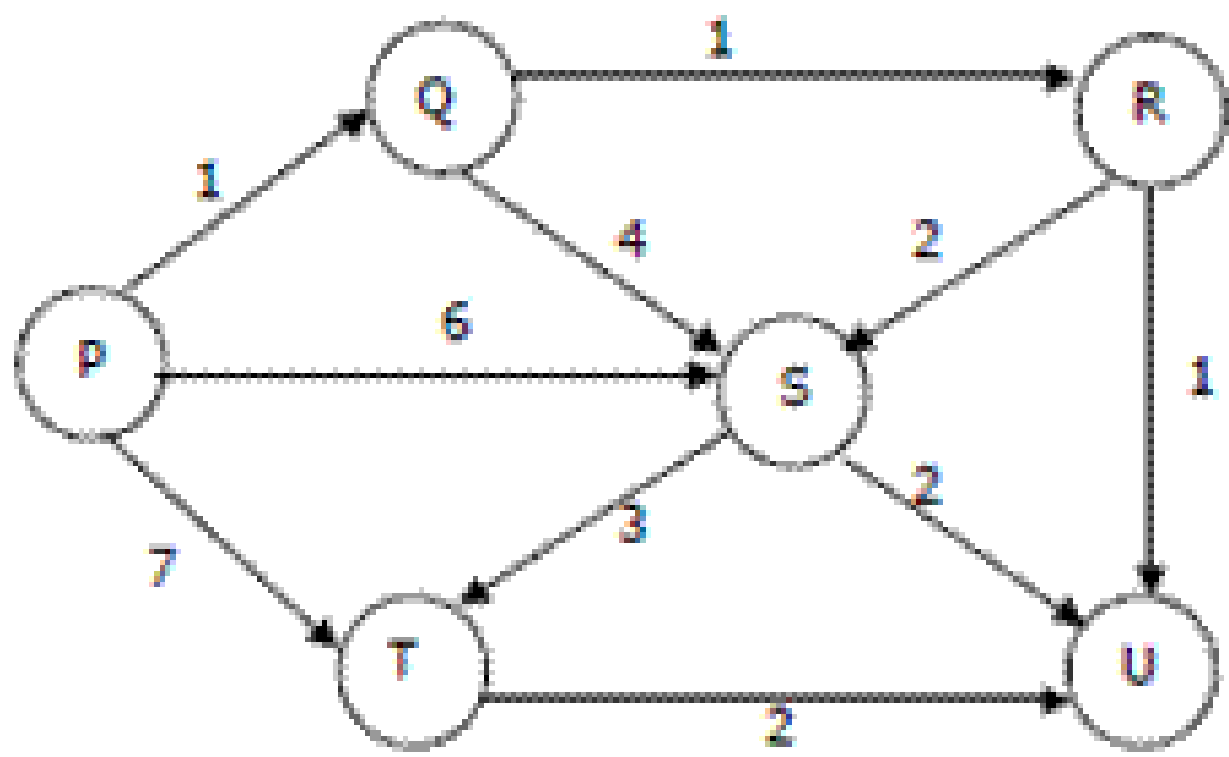
Node	SD	PN
A	0	/
B	3 2	A C
C	1	A
D	5	B
E	5 3	∈ B

Nothing changes

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



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

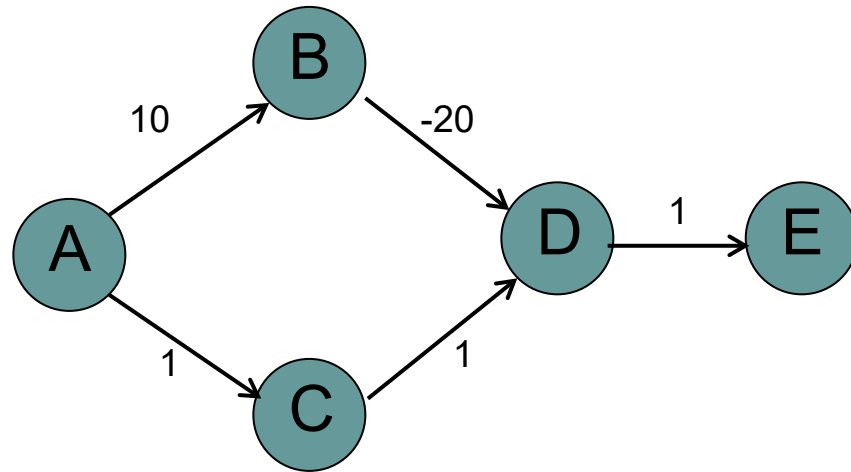


Visit order: P, Q, R, U, S, T

Node	SD	PN
P	0	
Q	1	P
R	2	Q
S	6 5 4	P Q R
T	7	P
U	3	R

Q. Topological Sort

Consider this DAG, use Topological Sort to find Shortest Paths in DAG, considering all possible topological orders



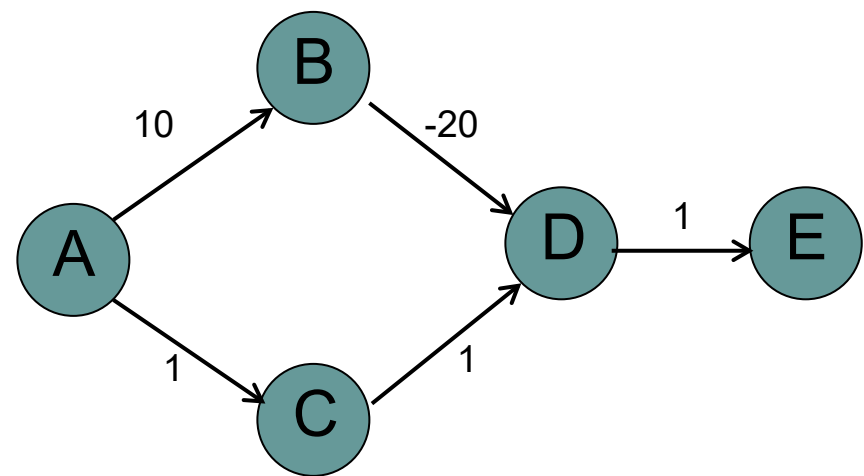
Visit Order

0, 1, 4, 7, 5, 2, 3, 6

Node	SD	PN
A	0	/
B		
C		
D		
E		

Q. Topological Sort Final Answer

Consider this DAG, use Topological Sort to find Shortest Paths in DAG, considering **all possible** topological orders



Visit Order		
A, B, C, D, E		

Node	SD	PN
A	0	
B	10	A
C	1	A
D	-10	B
E	-9	D

Visit Order		
A, C, B, D, E		

Node	SD	PN
A	0	
B	10	A
C	1	A
D	2 -10	∈ B
E	-9	D