Lecture 14

Minimum Spanning Trees

Exercises ANS

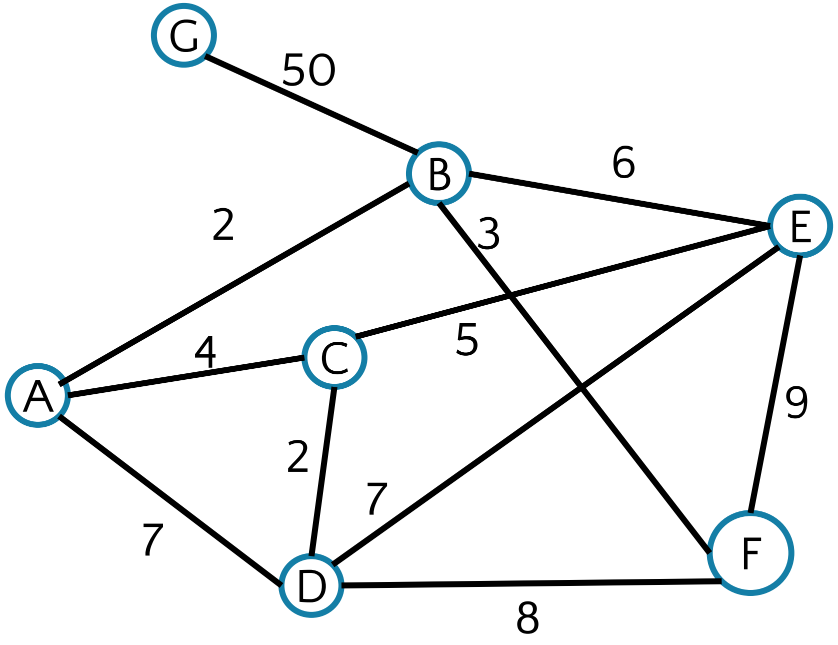
Department of Computer Science

Hofstra University

Acknowledgement: Lecture slides based on UofW Course on Data Structures

Q. Prim's Algorithm

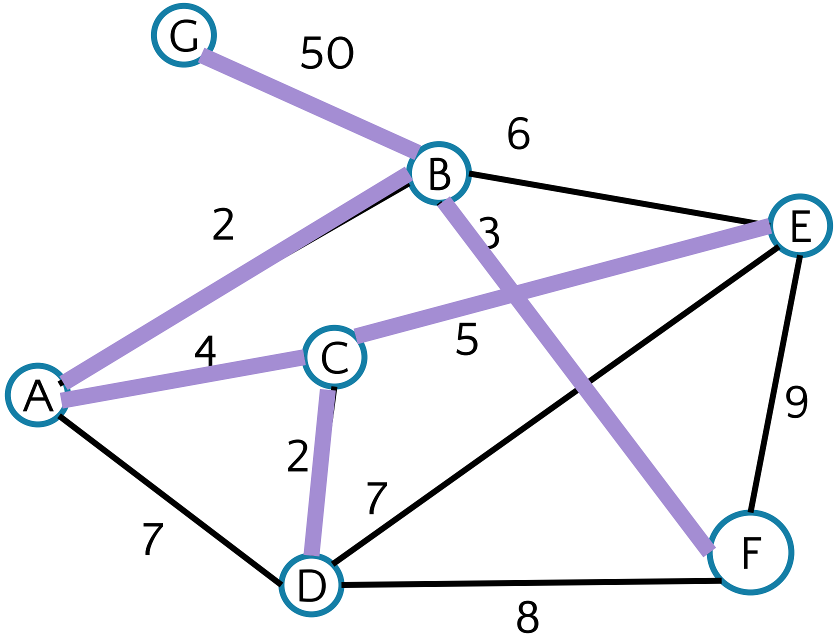
Run Prim's algorithm starting from node A. Fill in the table with the order in which each edge is added, and its weight. Break ties in alphabetical or numerical order. Highlight the final MST in the graph. For an undirected edge, write the nodes in alphabetical order, e.g., (E, F) instead of (F, E).



Order added	Edge	Edge Weight
1		
2		
3		
4		
5		
6		

Q. Prim's Algorithm ANS

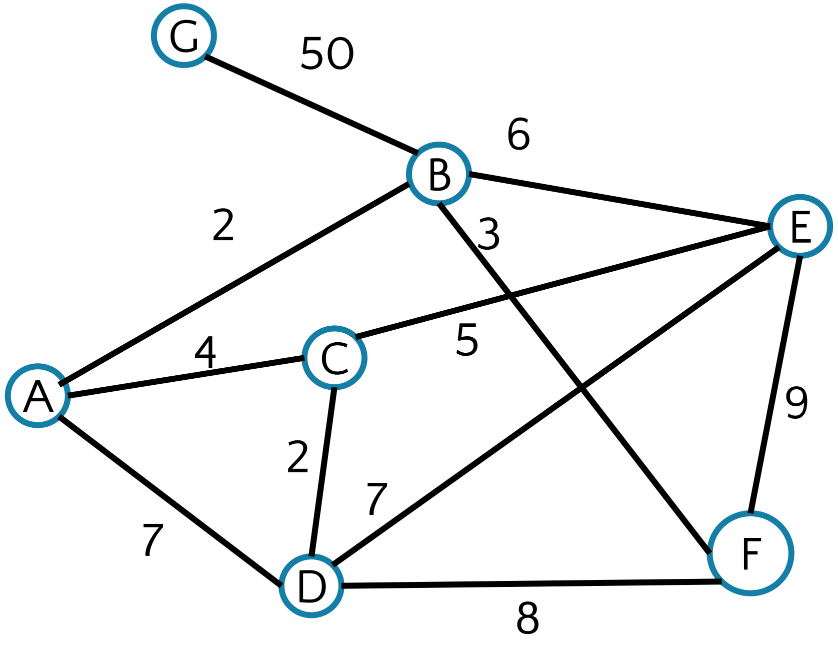
Run Prim's algorithm starting from node A. Fill in the table with the order in which each edge is added, and its weight. Break ties in alphabetical or numerical order. Highlight the final MST in the graph. For an undirected edge, write the nodes in alphabetical order, e.g., (E, F) instead of (F, E).



Order added	Edge	Edge Weight
1	(A, B)	2
2	(B, F)	3
3	(A, C)	4
4	(C, D)	2
5	(C, E)	5
6	(B, G)	50

Q. Kruskal’s Algorithm

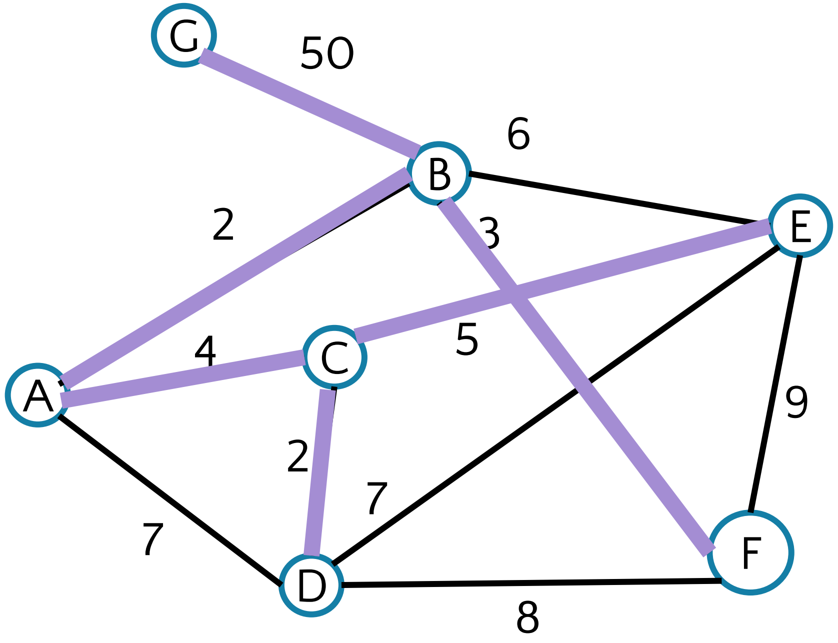
Run Kruskal’s algorithm. Fill in the table with the order in which each edge is added, and its weight. Break ties in alphabetical or numerical order. Highlight the final MST in the graph. For an undirected edge, write the nodes in alphabetical order, e.g., (E, F) instead of (F, E).



Order added	Edge	Edge Weight
1		
2		
3		
4		
5		
6		

Q. Kruskal's Algorithm ANS

Run Kruskal's algorithm. Fill in the table with the order in which each edge is added, and its weight. Break ties in alphabetical or numerical order. Highlight the final MST in the graph. For an undirected edge, write the nodes in alphabetical order, e.g., (E, F) instead of (F, E).



Order added	Edge	Edge Weight
1	(A, B)	2
2	(C, D)	2
3	(B, F)	3
4	(A, C)	4
5	(C, E)	5
6	(B, G)	50