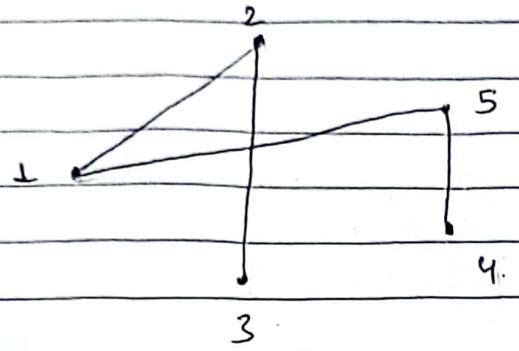
Assignment 8

1

→ Vertices {1, 2, 3, 4, 5}



Using BFS

Let, '1' be the initial vertex

Visited	①1				
Queue	①1				

Visiting adjacent vertices, 2 & 5, push them to queue and dequeue 1

Visited	1	2	5	4	
Queue	2	5			

Visiting adjacent vertices, 3 and enqueing, dequeing 2

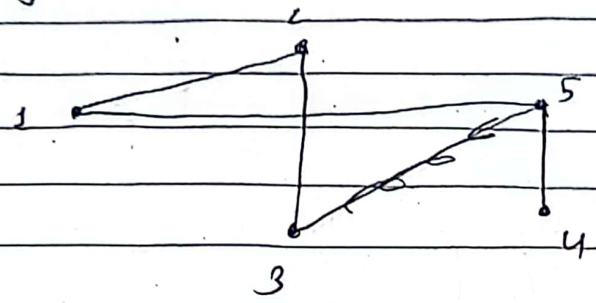
Visited 4	1	2	5	3	
Queue	5	3			

Visiting adjacent vertices, 4 and pushing to queue, dequeing 5.

Visited	1	2	5	3	4	
Queue	3					

All vertices have been visited, dequeing 3.

- Using DFS,



Let '1' be initial vertex, and push unvisited adjace nodes to st

Visited	1					
Stack	2	5				

Visit adjacent vertex and push to stack

all adjacent vertex

Visited	1	2

Top of stack is 5, so visit 5 and pop.
Push all unvisited adjacent nodes to stack

Visited	1	5				
Stack	2	4				

Top of stack is 4, so visit 4 and pop
Push all its unvisited adjacent nodes to stack

Visited	1	5	4			
Stack	2					

Top of stack is 2, so visit 2 and pop.
Push - - - -.

Visited	1	5	4	2		
Stack	3					

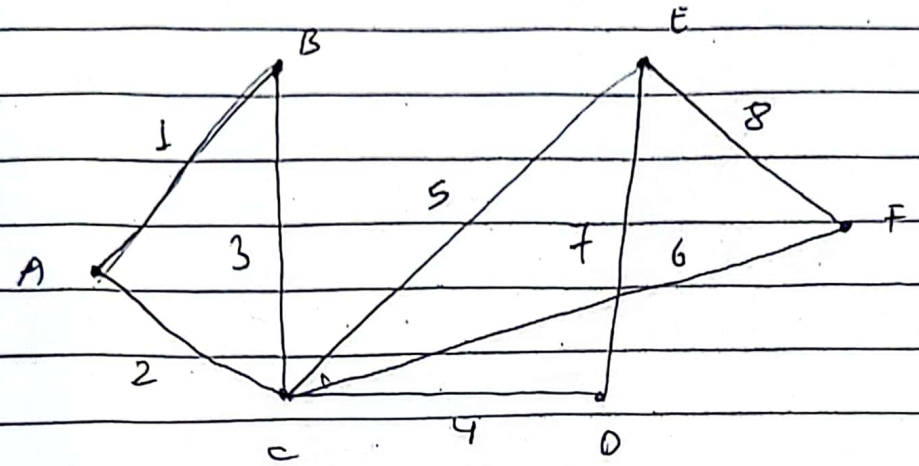
Visit top of stack and pop.

Visited	1	5	4	2	3	
Stack						

	A	B	C	D	E	F
A	0	1	2	0	0	0
B	1	0	3	0	0	0
C	2	3	0	4	5	6
D	0	0	4	0	7	0
E	0	0	5	7	0	8
F	0	0	6	0	8	0

2

→ Vertices {A, B, C, D, E, F}

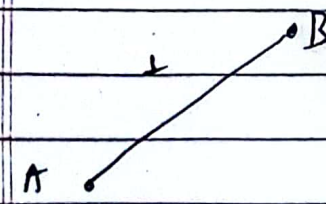


Using Kruskal's Algorithm,

Sorting all edges in ascending order,

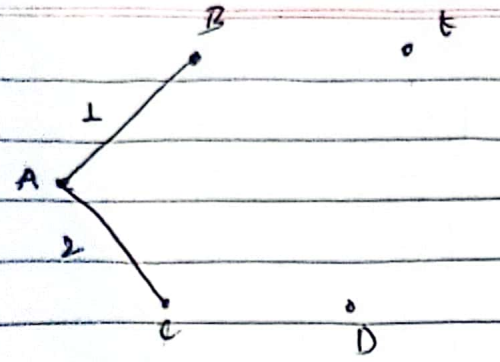
A, B = 1, A, C = 2, B, C = 3, C, D = 4, C, E = 5
C, F = 6, D, E = 7, E, F = 8

1)

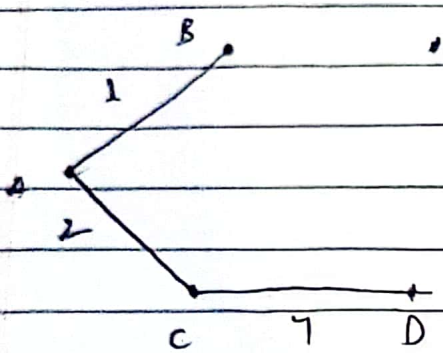


Date
Page

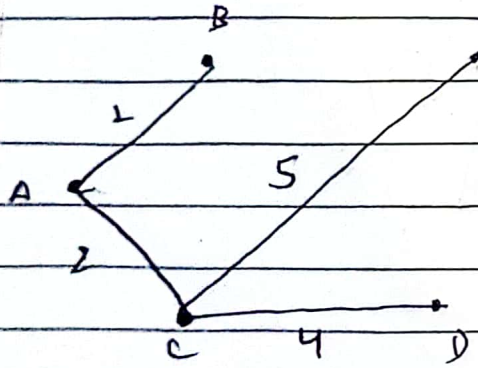
2)



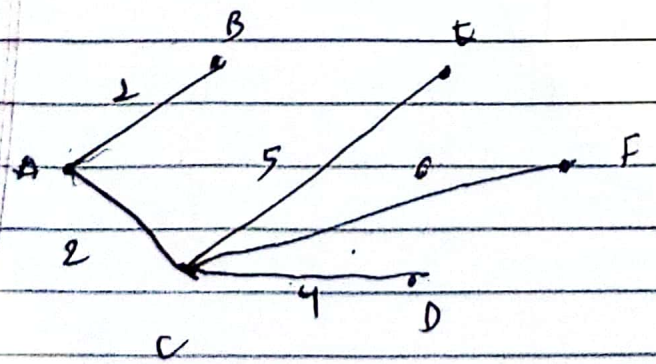
3)



4)



5)

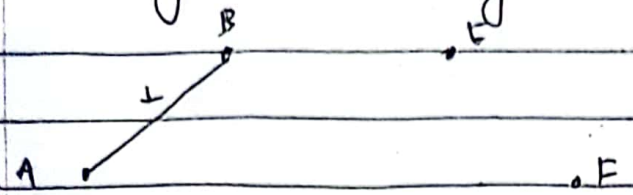


Weight of MST = 1 + 2 + 4 + 5 + 6

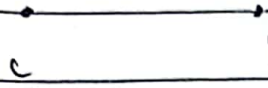
= 18

Using Prim's Algorithms,

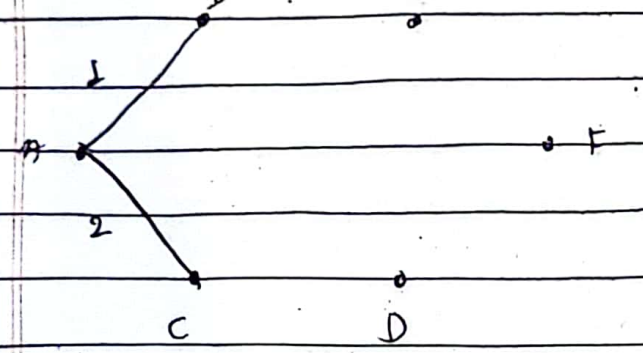
1)



AB has least weight so choose AB

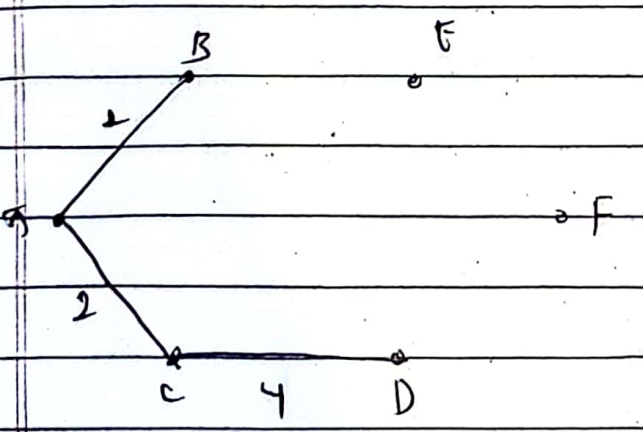


2)



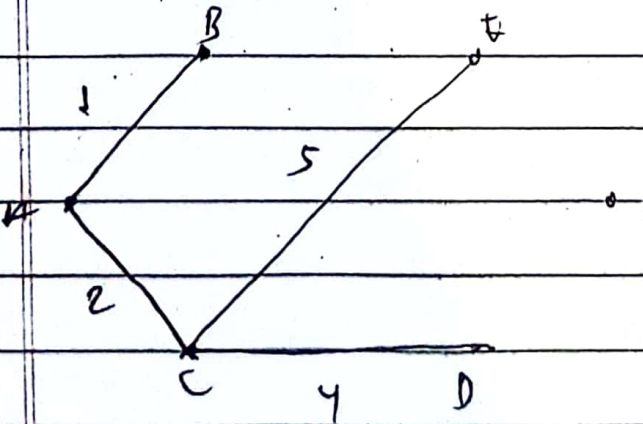
From A, and B, AC has least weight

3)



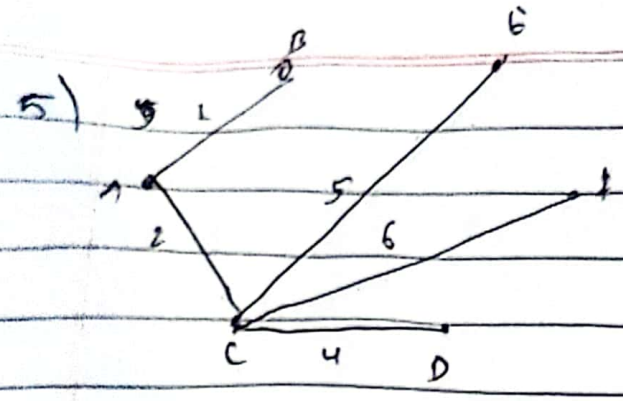
From A, B, C, BC has least weight but forms cycle. Choose next CD

4)



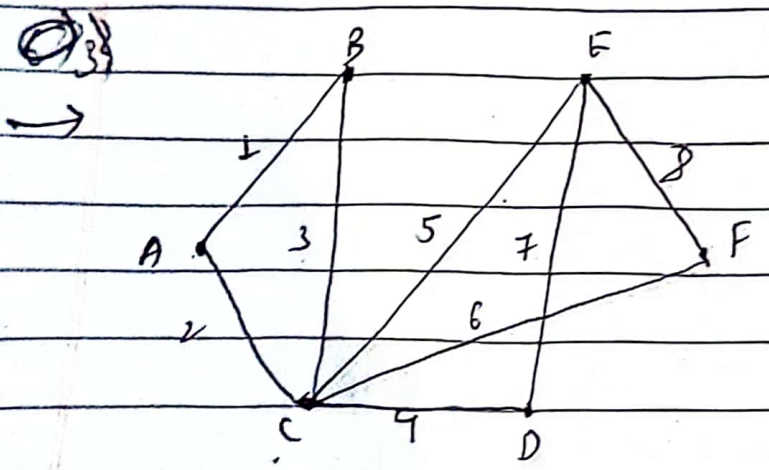
CE has least weight

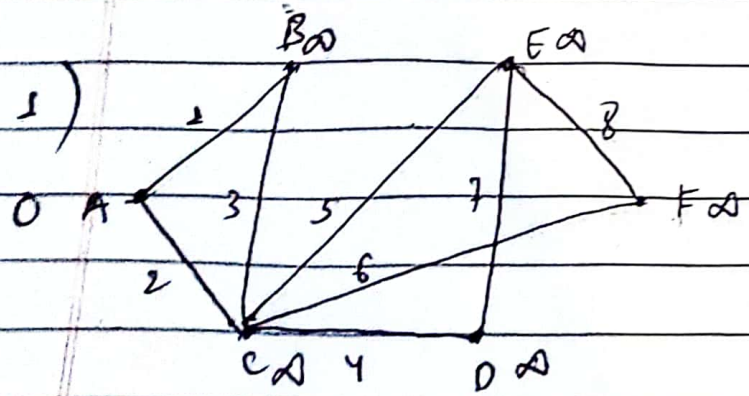
5)

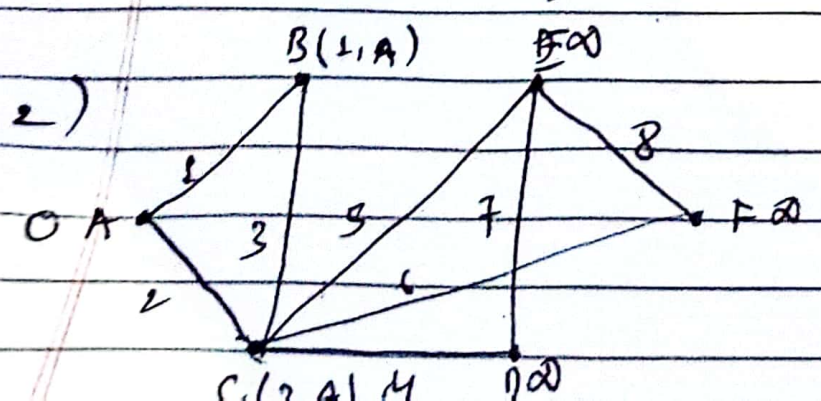


CF has least weight

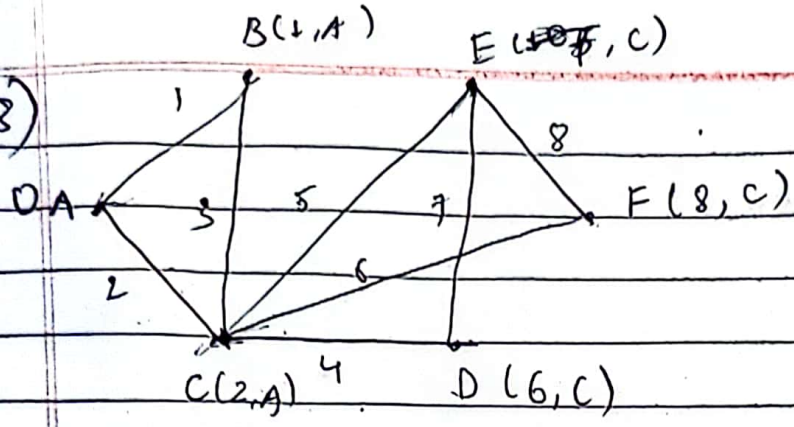
Weight of MST = 1+2+4+5+6
= 18







3)



$d(A,B) = 1(A,B)$

$d(A,C) = 2(A,C)$

$d(A,D) = 6(A,C,D)$

$d(A,E) = 5(A,C,E)$

$d(A,F) = 8(A,C,F)$