Prim's algorithm

It is a greedy algorithm that is used to find the minimum spanning tree from a graph. prim's algorithm finds the subset of edges that includes every vertex of the graph such as that the sum of weights of the edges can be minimized.

Prim's algorithm starts with the single node and explores all the adjacent nodes with all the connecting edges at every step. The edges with the minimal weights causing no cycle in the graph got selected.

how to work this algorithm?

=) prim's algorithm is a greedy algorithm that starts from one vertex and continue to add the edges with the smallest weight until the goal is reached.

The steps are ——

i) First to initialize an MST with the randomly choosen vertex.

ii) Now to find all the edges that connect the tree in the above step with new vertices. From the edges found select the minimum edge and add it to the tree.

iii) Repeat step 2 until minimum spanning tree is formed.

application: i) prim's algorithm can be used in network designing. ~~It can be used~~

ii) It can be used to make network cycles.

iii) It can be used to lay down electrical wiring cables.

complexity of prim's algorithm

Data structure used for the minimum edge weight	Time complexity
Adjacent matrix, linear searching	$O(\|V\|^2)$
Adjacency list and binary heap	$O(\|E\| \log \|V\|)$
Adjacency list and fibonacci heap	$O(\|E\|) + \|V\| \log \|V\|$

Algorithm for prim's :-

step 1 = Select a starting vertex

step 2 = Repeat steps 3 and 4 until there are fringe vertices

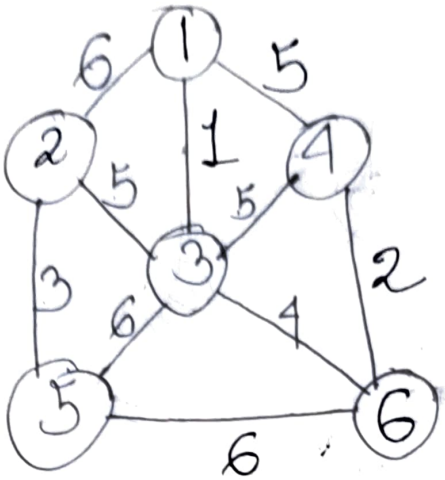
step 3 = Select an edge 'e' connecting the tree vertex and fringe vertex that has minimum weight

step 4 = Add the selected edge the vertex to the minimum spanning tree.

[END OF LOOP]

step 5 = EXIT

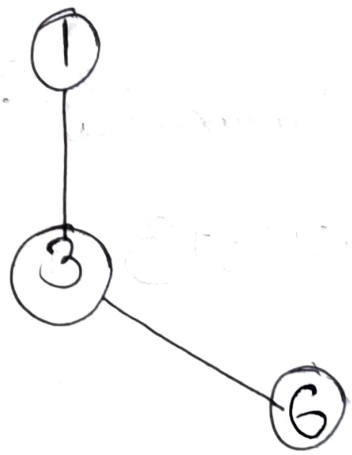
Prim's Algorithm



Step 1 -

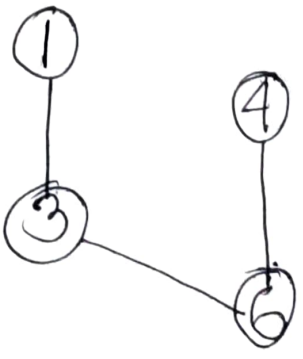
(1,3) = 1

Step 2 -



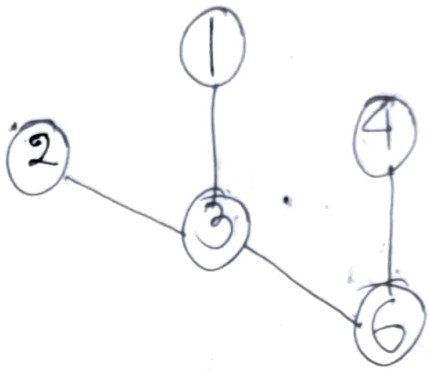
(3,6) = 4

Step 3 -



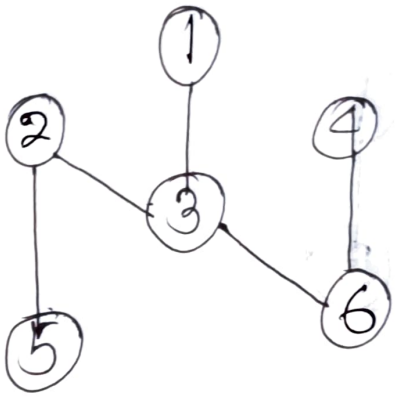
(4,6) = 2

Step 4:-



(3,2) = 5

Step 5:-



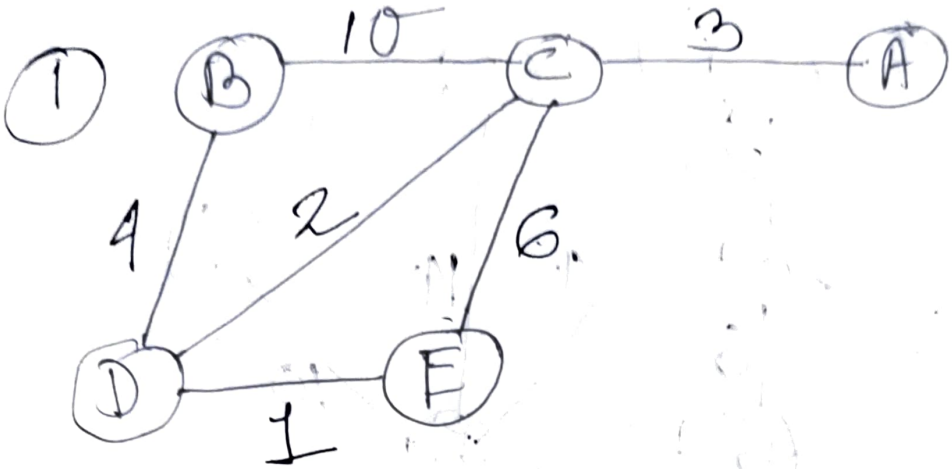
(2,5) = 3

The Minimum Cost Spanning Tree

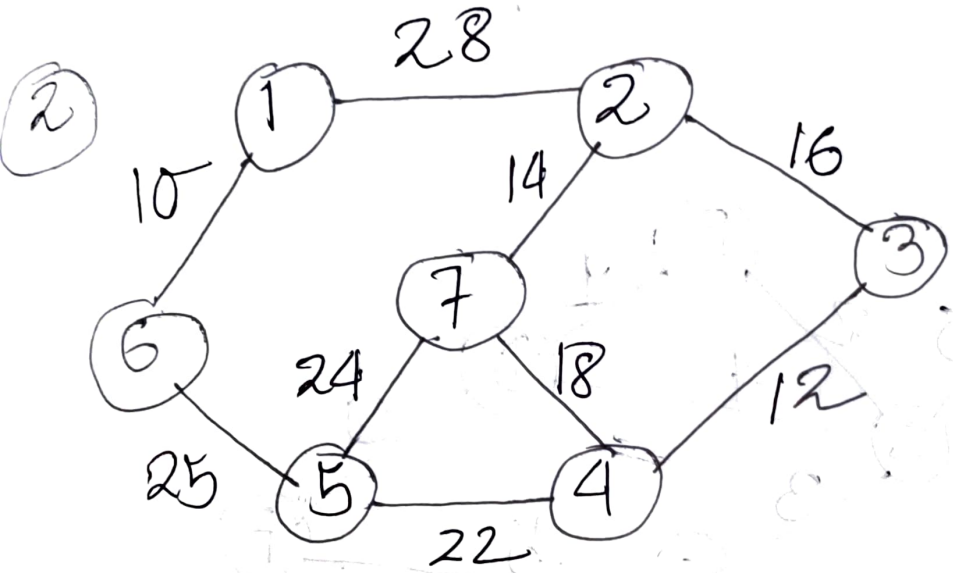
$\Rightarrow 1+4+2+5+3 = 15.$

Hw: Prim's & Kruskals

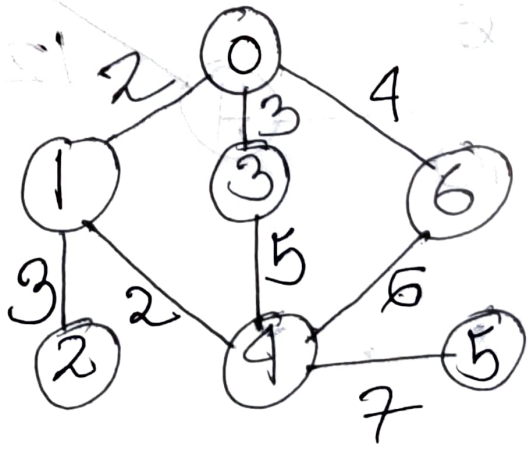
1

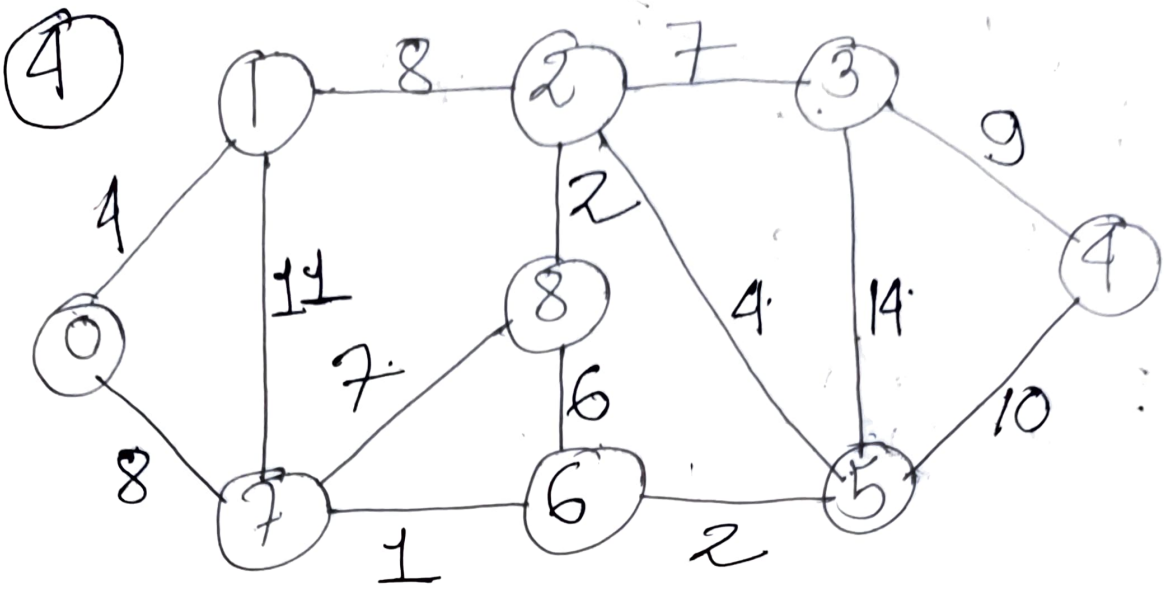


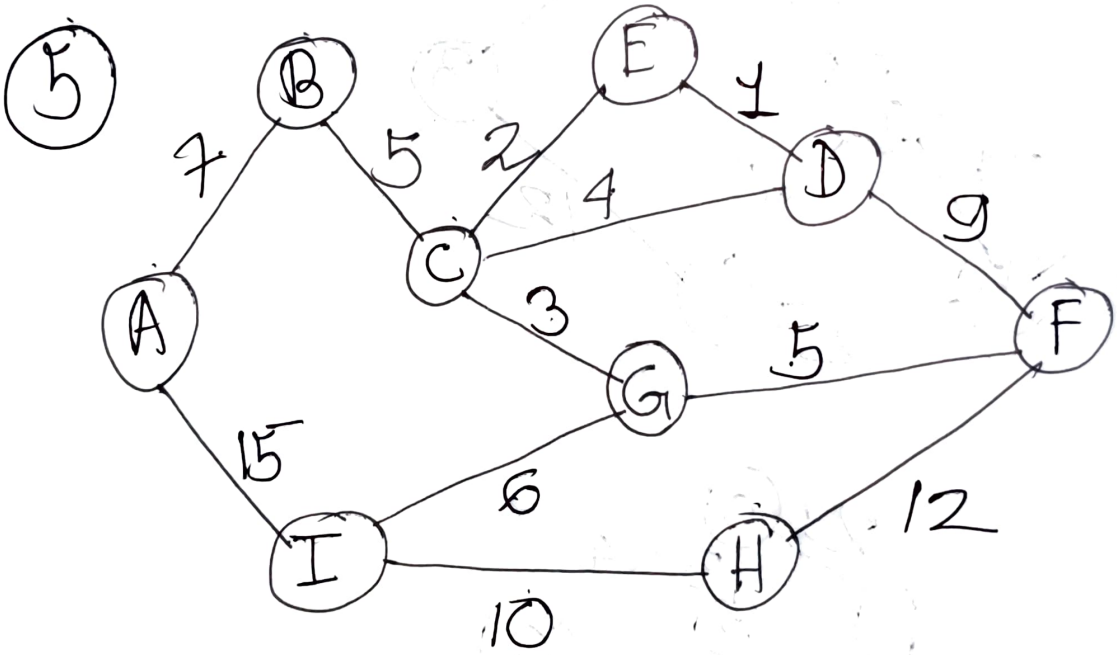
2



3




4
1
8
2
7
3
9
4
1
11
2
0
8
4
14
7
6
10
8
7
6
5
1
2


5
B
E
1
7
5
2
D
4
9
C
A
3
F
5
G
15
6
12
I
H
10