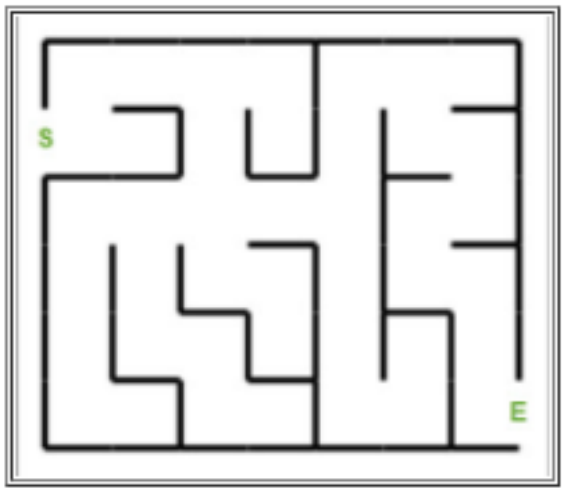
CS455 WEEK 12 HOMEWORK
SHOUMYA SINGH
ID-19566

Q25. Use Prim's Minimum Spanning Tree algorithm and Kruskal's Minimum Spanning Tree algorithm to find the shortest path of a maze.

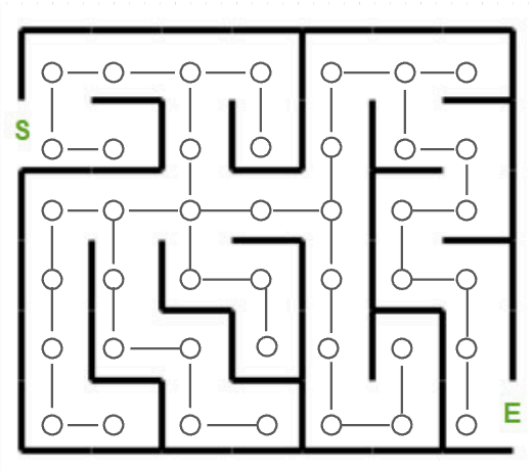
- Step 1: Similar to the previous question of finding the shortest path of the a maze. But instead of using Dijkstra's Algorithm, you will use Minimum Spanning Tree Algorithm.
- Step 2: Comparing the performance of these two algorithm in solving this question by
  - Big-O comparison
- Step 3: Update your portofolio about the Maze project

STEP 1

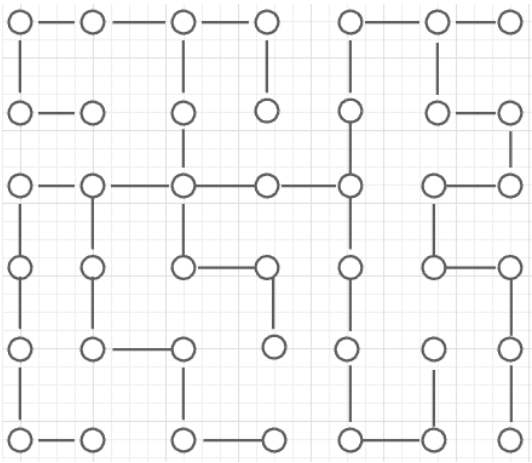


STEP 2

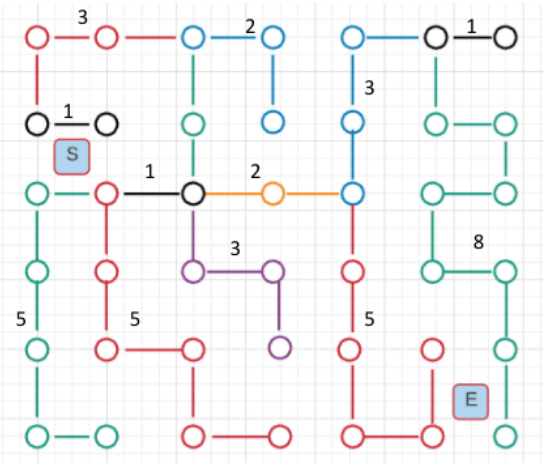
STEP 3

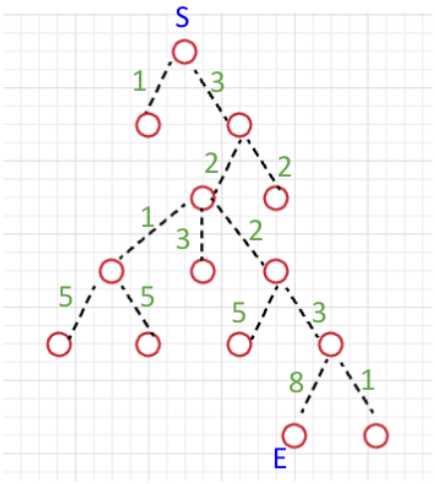


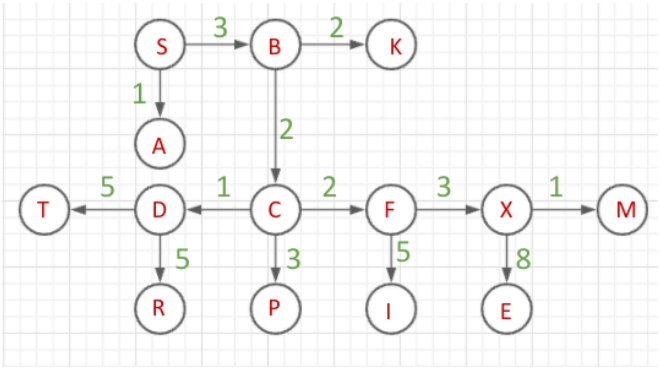
STEP 4



STEP 5

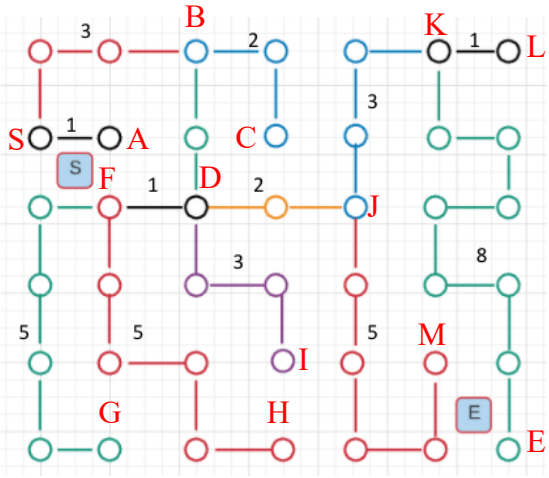




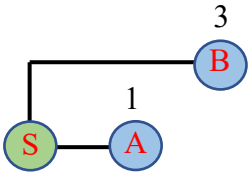


Prim's Minimum Spanning Tree (MST)

A. Create a set mstSet that keeps track of vertices already included in MST.
B. Assign a key value to all vertices in the input graph.
    1. Initialize all key values as INFINITE.
    2. Assign key value as 0 for the first vertex so that it is picked first.
C. While mstSet doesn't include all vertices
    1. Pick a vertex u which is not there in mstSet and has minimum key value.
    2. Include u to mstSet.
    3. Update key value of all of u's adjacent vertices which are not in mstSet.

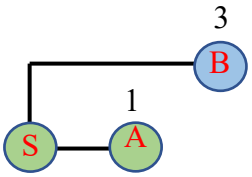


STEP 1 :



S

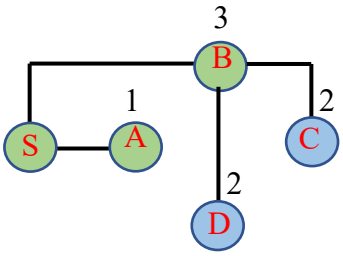
STEP 2 :



S->A
1

Starting from S, the tree can link to two nodes then choose the one with the smallest weight which is A and then go to STEP 2.

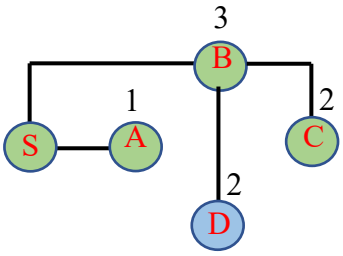
STEP 3 :



S->A->B
1+3=4

Since A doesn't connect to other nodes, go to next smallest node which is B .

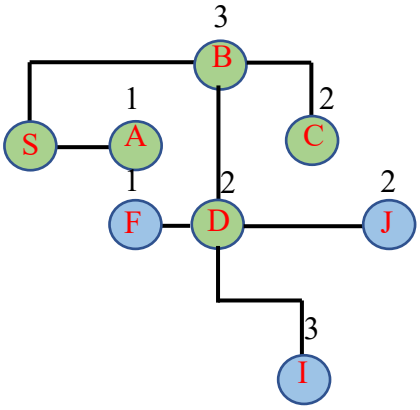
STEP 4 :



S->A->B->C
1+3+2=6

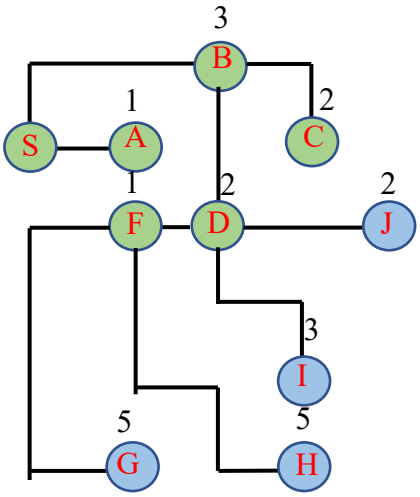
Now repeat the rule like step 1,2 and keep choosing the smallest nodes till all the vertices are visited.

STEP 5 :



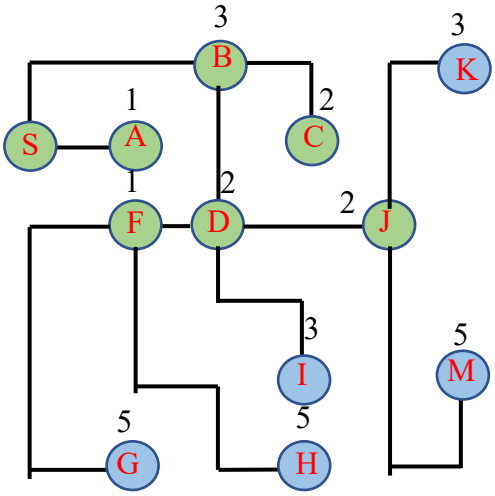
S->A->B->C->D
1+3+2+2=8

STEP 6 :



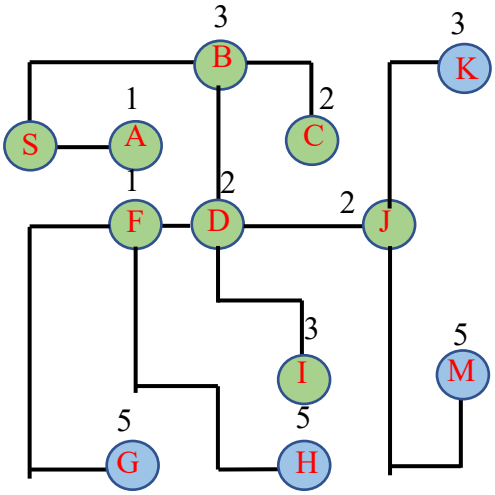
S->A->B->C->D->F
1+3+2+2+1=9

STEP 7 :



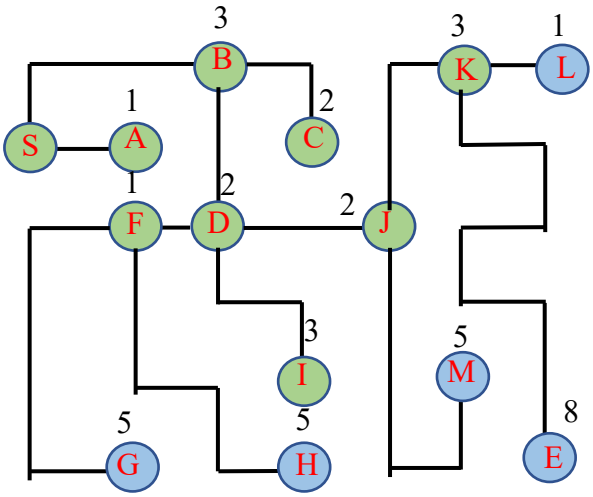
S->A->B->C->D->F->J
1+3+2+2+1+2=11

STEP 8 :



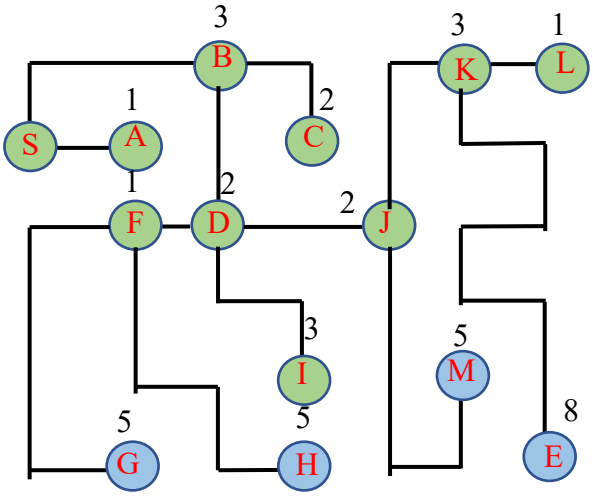
S->A->B->C->D->F->J->I
1+3+2+2+1+2+3=14

STEP 9 :



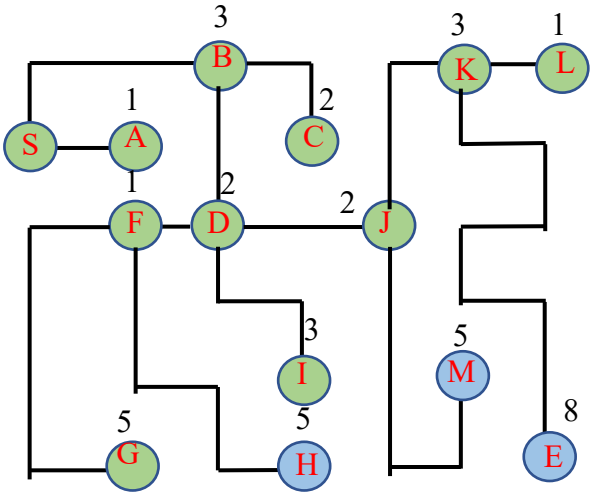
S->A->B->C->D->F->J->I->K
1+3+2+2+1+2+3+3=17

STEP 10 :



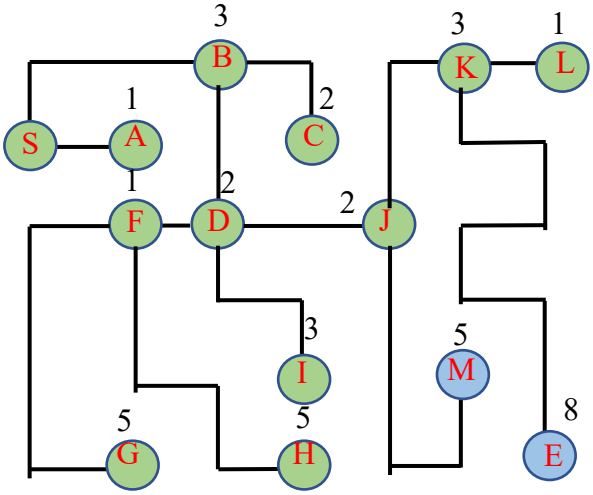
S->A->B->C->D->F->J->I->K->L
1+3+2+2+1+2+3+3+1=18

STEP 11 :



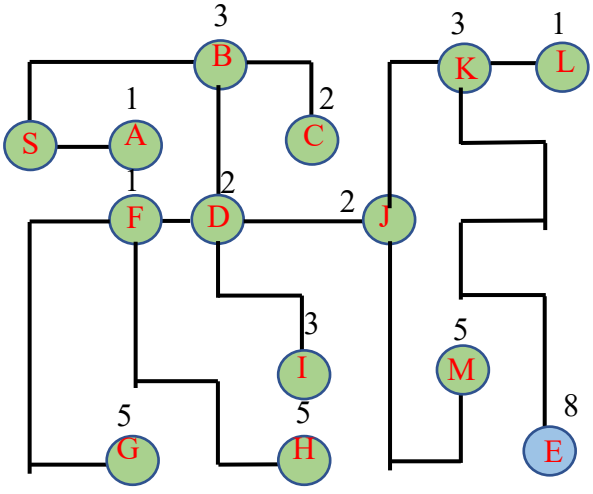
S->A->B->C->D->F->J->I->K->L->G
1+3+2+2+1+2+3+3+1+5=23

STEP 12 :



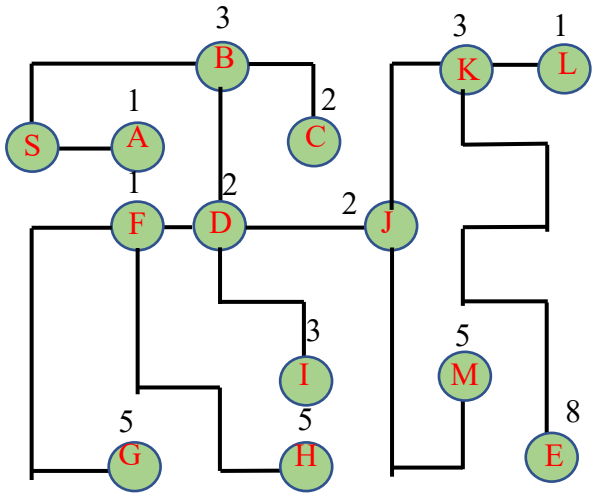
S->A->B->C->D->F->J->I->K->L->G->H
1+3+2+2+1+2+3+3+1+5+5=28

STEP 13 :



S->A->B->C->D->F->J->I->K->L->G->H->M
1+3+2+2+1+2+3+3+1+5+5+5=33

STEP 14 :



The shortest path of the given maze with Prim's minimum spanning tree is as follows:

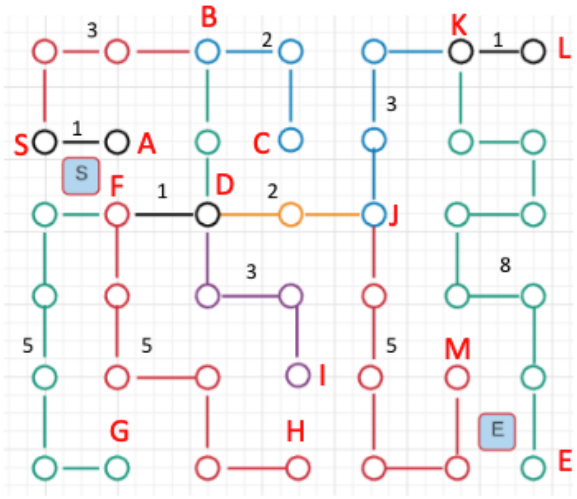
S->A->B->C->D->F->J->I->K->L->G->H->M->E
1+3+2+2+1+2+3+3+1+5+5+5+8=41
Total steps 14

Time Complexity :

O((v + E)logV)

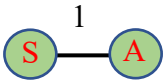
Kruskal's Minimum Spanning Tree

1. Sort all the edges in non-decreasing order of their weight.
2. Pick the smallest edge. Check if it forms a cycle with the spanning tree formed so far. If cycle is not formed, include this edge. Else, discard it.
3. Repeat step#2 until there are (V-1) edges in the spanning tree.



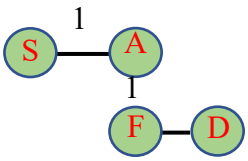
After Sorting		
Weight	**Source**	**Destination**
1	S	A
1	D	F
1	K	L
2	B	C
2	B	D
2	D	J
3	S	B
3	D	I
3	J	K
5	F	G
5	F	H
5	J	M
8	K	E

STEP 1 :

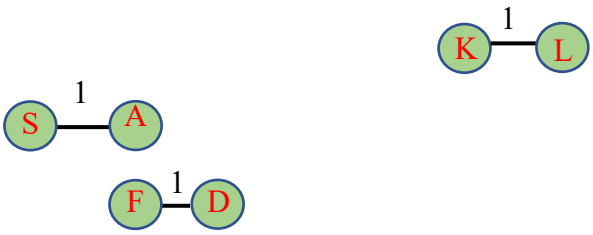


The smallest weight is 1 which edge is S – A.

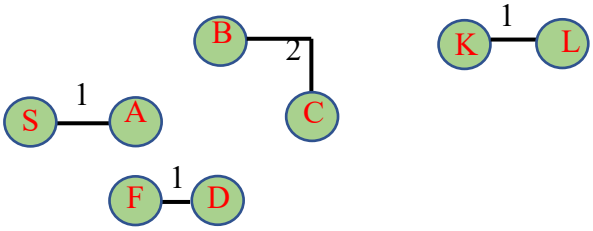
STEP 2 :



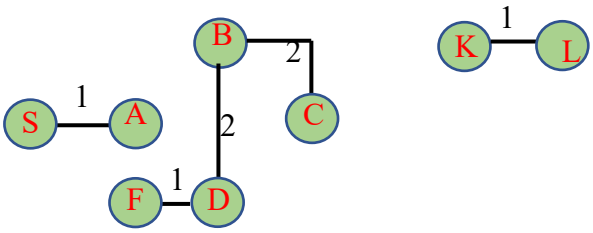
STEP 3 :



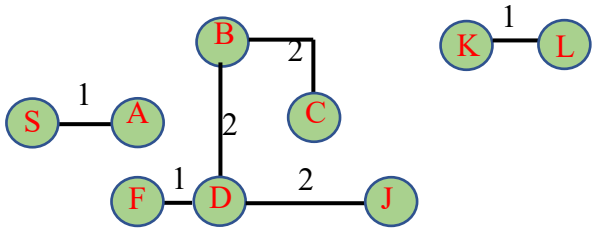
STEP 4 :



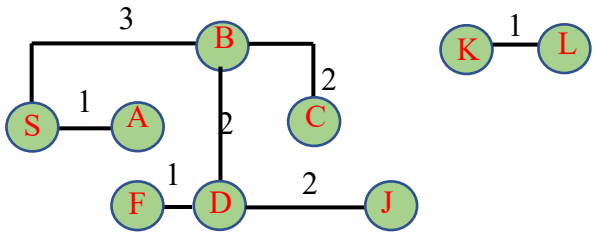
STEP 5 :



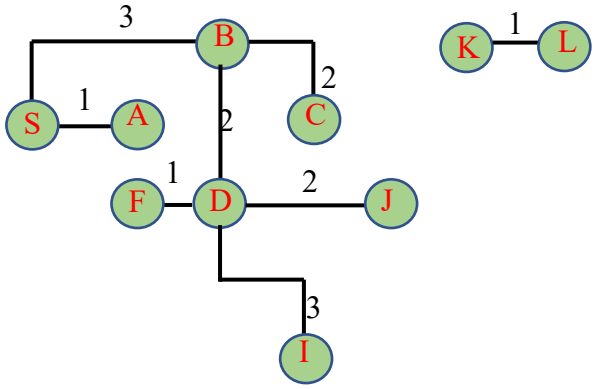
STEP 6 :



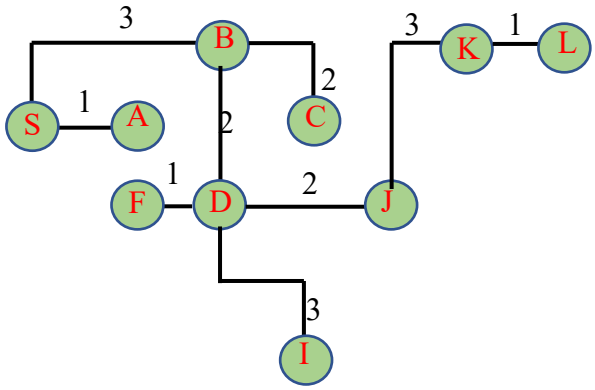
STEP 7 :



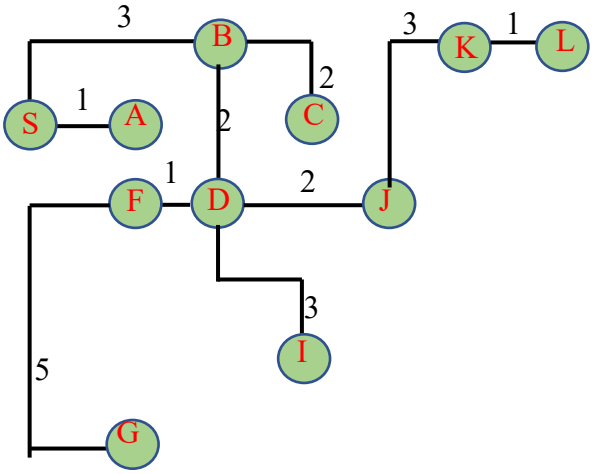
STEP 8 :



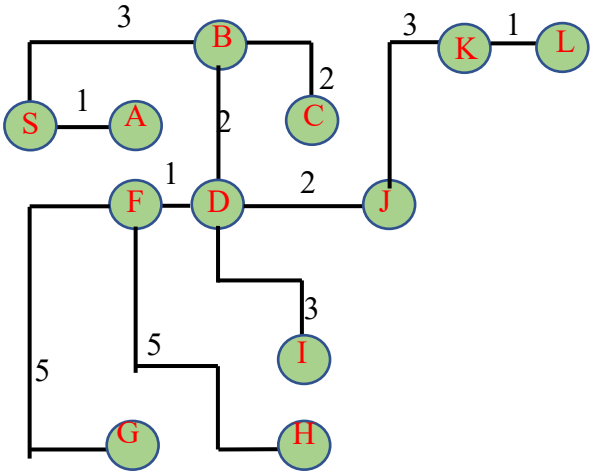
STEP 9 :



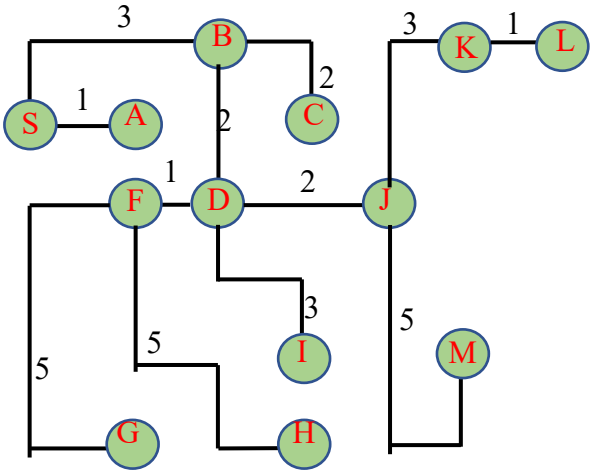
STEP 10 :



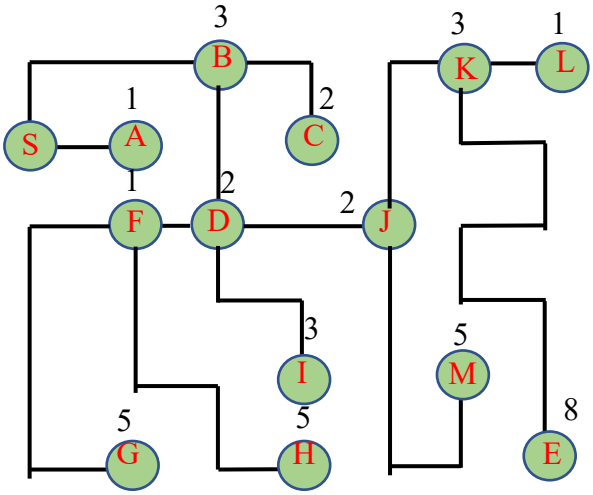
STEP 11 :



STEP 12 :



STEP 13 :



The graph contains 14 vertices . So, the minimum spanning tree formed will be having (14 – 1) = 13 edges.

Since the number of edges equals to (V – 1), the algorithm stops here.

Total weight = 41
Total steps 13.
Time complexity :

O(E * logV)