Tarun SriRam.D
1BM19CS171
5/07/2021

ADA - LAB TEST-2

WRITEUP

```
# include<stdio.h>
int minKey( int key[], int mstSet[], int n)
 {
   int min = 100, min_index;
   int v;
   for(v=0 ; v<n ; v++)
       if(mstSet[v] == 0 && key[v]<min)
        min = = key[v], min_index = v;
     return min_index;
   }
 int printMST(int parent[10], int graph[10][10], int n)
 {
   int i;
  printf("Edge \t Weight \n");
   for (i=1 ; i<n ; i++)
       printf("%d - %d \t %d \n", parent[i], i, graph[i][parent[i]]);
  }
  void primMST (int graph[10][10], int n)
   {
    int parent [n];
    int key [n];
    int mstSet [n];
    int i, count, v, u;

    for(i=0; i<n ; i++)
      key[i]= 100, mstSet[i]= 0;

    key[0] = 0;
    parent[0] = -1;
```

Tarun Sri Ram. D
1BM19CS171

```
for ( count =0; count < n-1; count ++) {
    u= minKey (key , mstSet , n);
    mstSet[u] = 1;
    for ( v=0 ; v<n ; v++)
    if ( graph[u][v] && mstSet [v] == 0 && graph[u][v] < key [v])
     parent [v] = u, key [v] = graph[u][v];
   }
   printMST ( parent, graph , n);
}
int main()
{
int graph [10][10];
int i,j,n;
printf (" enter number of nodes \n");
scanf ("%d", &n);
printf (" Enter adjacency matrix \n");
for ( i=0 ;i<n ;i++)
{
   for ( j=0; j<n; j++)
   scanf (" %d', &graph (i)[j]);
}
prim MST (graph,n);
return 0;
}
```

Tarun SriRam.D
1BM19CS173

Modification :-

* Take the input from user which node has to be excluded. Ex:- ~~Exclude~~ Excluded node = "c"
* In primMST function set it has visited, than make its corresponding nodes as infinity.

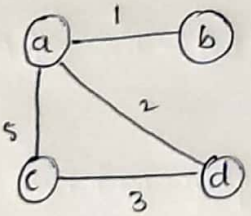
```
mstSet[specified_node] = 2

for(i=0;i<n;i++)
  {
   for(j=0;j<n;j++){
  if(i==specified_node || j==specified_node)
  graph[i][j] = 999;
```

~~From this above algo. do~~

From above pseudo code we are excluding 'c' by making its distance infinity w.r.t ~~the~~ other node.

Ex:-



=>

By excluding 'c' node