5. Prims Algorithm ADA LAB Test - 2

1BM19CS085
Md Ibaduddin Saffan

```
#include <stdio.h>
int cost[10][10], vis[10], vt[10], et[10][2], u, v, ecost, sum=0;

void prims()
{
    int x=1;
    vt[x]=1;
    int min, k, m, j;
    for(int i=1; i<n; i++)
    {
        min = 999;
        j=x;
        while (j>0)
        {
            k = vt[j];
            for (m=2; m<=n; m++)
            {
                if (cost[k][m] <min && vis[m]==0 && cost[k][m]!=0)
                {
                    min = cost[k][m];
                    u=k;
                    v=m;
                }
            }
            j--;
        }
        vt[++x] = v;
        et[i][0]=u;
        et[i][1]=v;
        ecost;
        vis[v]=1;
        sum = sum + cost[u][v];
    }
}
```

ADA Lab test - 2.

1BM19CS085
Md Ibaanuddin Saffan

```
void main()
{
    printf("Enter the number of vertices : \n");
    scanf("%d", &n);
    printf("Enter the cost adjacency matrix : \n");
    for(int i=1; i<=n ; i++)
    {
        for(int j=1 ; j<=n ; j++)
        {
            scanf("%d", &cost[i][j]);
        }
    }
    prims();
    printf("Cost : %d \n", sum);
    printf("Edges selected : \n");
    for(int i=0; i<e ; i++)
    {
        printf("%d to %d \n", et[i][0], et[i][1]);
    }
}
```

Modification;

Print nodes that can be visited excluding specified node.

```
#include <stdio.h>
int cost[10][10], vt[10], vis[10], et[10][2], u, v, e=0, n, sum=0;

void prims()
{   int x=1;
    vt[x]=1;
    int min, k, m, j;
    for(int i=1; i<n; i++)
    {   min=999;
        j=x;
        while(j>0)
        {   k=vt[j]
            for(m=2; m<=n; m++)
            {   if(cost[k][m]<min && vis[m]==0 && cost[k][m]!=0)
                {   min=cost[k][m];
                    u=k;
                    v=m;
                    vis[m]=1;
                }
            }
            j--;
        }
        vt[++x]=v; if(et[i-1][0]!=u && et[i-1][1]!=v && u!=v
        et[i][0]=u;
        et[i][1]
        {   vt[++x]=v;
            et[i][0]=u;
            et[i][1]=v;
            vis[v]=1;
            sum=sum+cost[u][v];
        }   e++;
}
```

1BM19CS078
Md Ibaadudallin Saffan

```
void main()
{
    printf("Enter the number of vertices : \n");
    scanf("%d", &n);
    printf("Enter the cost adjacency matrix : \n");
    for(int i=1; i<=n; i++)
    {
        for(int j=1; j<=n; j++)
        {
            scanf("%d", &cost[i][j]);
        }
    }
    printf("Enter the vertex which should not be visited : \n");
    scanf("%d", &forbidden);
    vis[forbidden] = 1;
    printf("Edges visited : \n");
    for(int i=0; i<ne; i++)
    {
        printf("%d to %d \n", et[i][0], et[i][1]);
    }
    printf("Total cost : %d\n", sum);
}
```