LAB-10   PRIM'S ALGO   01/07/2021

```c
#include <stdio.h>
#include <conio.h>
void prims();    int c[10][10],n;
void main()
{
    int i,j;
    printf("\n Enter  no  of  vertices: ");
    scanf("%d", &n);
    printf("\n enter the cost matrix \n");
    for (i=1; i<=n; i++)
    {
        for(j=1; j<=n; j++)
        {
            scanf("%d", &c[i][j]);
        }
    }

    prims();
}
void prims()
{
    int i,j,u,v,min;
```

```
int ne = 0, mincost = 0;
int elec[10];
for (i = 1; i <= n; i++)
{ elec[i] = 0; }
elec[1] = 1;
while (ne != n-1)
{ min = 9999.

for (i = 1; i <= n; i++)
{ for (j = 1; j <= n; j++)
{ if (elec[i] == 1)
{ if (c[i][j] < min)
{ min = c[i][j];
u = i; v = j; }
}
} }
```

```c
if (elec[v] != 1)
{ printf("\n o/p d -->:/:d = /:d \n", y, v,
   min);

elec[v] = i;

ne++;

mincost = mincost + min;
}

c[u][v] = c[v][u] = 9999;
}

printf("\n Total mincost = o/:d", mincost);
}
```

LAB-10   -KRUSHKAL - ALGO

```c
# include <stdio.h>
# include <conio.h>
void kruskal();
int c[10][10], a;
void main()
{ int i, j;
```

```c
printf (" Enter no of vertices \t ");
scanf (" %d ", &n);
printf (" Enter cost matrix \n ");
for (i = 1 ; i < n; i++)
{ for (j = 1; j < n; j++) {
scanf (" %d ", &c [i][j]); } }
kruskals ( ); }

void kruskals ()
{ int i, j, u, v, a, b, min;
int ne = 0 , min cost = 0; int parent [10];
for (i = 1 ; i < n; i++)
{ parent [i] = 0;
}
while (ne != n-1)
{ min = 9999;
```

```
for (i=1; i < n; i++)
{ for (j=1; j <= n; j++)
  { if (c[i][j] < min)
    { min = c[i][j];
        u = a:
          = i;
        v = a = j;
    } } }

while (parent[u] != 0)
{ u = parent[u]; }

while (parent[v] != 0)
{ v = parent[v]; }

if (u != v)
{
  printf ("/n %d --> %d = %d \n", a, b, min);

  parent[v] = u;
  ne++;
  mincost = mincost + min; }
```

```
    c[a][b] = c[b][a] = 9999;
    }
    }
    printf(" \n  new cost = %d , new cost);  }
```