PROGRAM 9

**9)**

Find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.

package labprograms;

import java.util.Scanner;

public class p9

{

public static void main(String [] args)

{

int n,i,j;

int[][] cost=new int[10][10];

Scanner sc=new Scanner(System.in);

System.out.print("Enter the number of vertices : ");

n=sc.nextInt();

System.out.println("Enter the cost adjacency matrix,'1000' for no direct path ");

for(i=1;i<=n;i++)

for(j=1;j<=n;j++)

cost[i][j]=sc.nextInt();

prims(cost,n);

sc.close();

}

static void prims(int cost[][],int n)

{

int[] v=new int[10];

int min,p,q,i,j,flag=0,mincost=0,s=1;

for(i=1;i<=n;i++)

v[i]=0;

v[1]=1;

System.out.println("The spanning tree has the following edges : ");

while(s!=n)

{

min=9999;

i=j=-1;

flag=0;

for(p=1;p<=n && s!=n;p++)

{

for(q=1;q<=n;q++)

{

if(p==q)

continue;

if((min>cost[p][q]) && (v[p]==1) && (v[q]!=1))

{

min=cost[p][q];

i=p;

j=q;

flag=1;

}

}

}

if(i!=-1)

{

v[j]=1;

System.out.println("("+i+","+j+") ="+ cost[i][j]);

mincost+=cost[i][j];

s++;

}

if(flag==0)

{

System.out.println("Graph is disconnected\n");

return;

}

}

System.out.println("Cost of spanning tree:"+ mincost);

}

}