#include<stdio.h>

int i,j,k,a,b,u,v,n,ne=1;

int min,mincost=0,a1[9][9],parent[9];

int find(int);

int uni(int,int);

int main()

{

printf("\nEnter the no. of vertices:");

n=6;

printf("\nEnter the cost adjacency matrix\n");

a1[1][1]=999;

a1[1][2]= 3;

a1[1][3]= 1;

a1[1][4]= 6;

a1[1][5]= 999;

a1[1][6]= 999;

a1[2][1]= 3;

a1[2][2]= 999;

a1[2][3]= 5;

a1[2][4]= 999;

a1[2][5]= 3;

a1[2][6]= 999;

a1[3][1]= 1;

a1[3][2]=5;

a1[3][3]= 999;

a1[3][4]= 5;

a1[3][5]= 6;

a1[3][6]= 4;

a1[4][1]= 6;

a1[4][2]= 999;

a1[4][3]= 5;

a1[4][4]= 999;

a1[4][5]= 999;

a1[4][6]= 2;

a1[5][1]= 999;

a1[5][2]= 3;

a1[5][3]= 6;

a1[5][4]= 999;

a1[5][5]= 999;

a1[5][6]= 6;

a1[6][1]= 999;

a1[6][2]= 999;

a1[6][3]= 4;

a1[6][4]= 2;

a1[6][5]= 6;

a1[6][6]= 999;

printf("\nThe edges of Minimum Cost Spanning Tree are\n\n");

while(ne<n)

{

for(i=1,min=999;i<=n;i++)

{

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

{

if(a1[i][j]<min)

{

min=a1[i][j];

a=u=i;

b=v=j;

}

}

}

u=find(u);

v=find(v);

if(uni(u,v))

{

printf("\n%d edge (%d,%d) =%d\n",ne++,a,b,min);

mincost +=min;

}

a1[a][b]=a1[b][a]=999;

}

printf("\n\tMinimum cost = %d\n",mincost);

return 0;

}

int find(int i)

{

while(parent[i])

i=parent[i];

return i;

}

int uni(int i,int j)

{

if(i!=j)

{

parent[j]=i;

return 1;

}

return 0;

}

***Reading from user:***

printf("\nEnter the no. of vertices:");

scanf("%d",&n);

printf("\nEnter the cost adjacency matrix\n");

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

{ printf("\n");

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

{

printf("a[%d][%d]: ",i,j);

scanf("%d",&cost[i][j]);

if(cost[i][j]==0)

cost[i][j]=999;

}

}