#include<iostream>

#define N 100

#define maxL 1000000

using namespace std;

void prim(int a[][N],int u[],int w[],int b[],int n)

{

int i=0,j=0,m=0,minL=maxL;

int allL=0;//光纤总长度

for(i=0;i<n;i++)

{

u[i]=0;//设置每个点初始没有访问过

w[i]=a[0][i];//初始每个顶点最短路径为到0顶点的距离

}

u[0]=1;//从0顶点开始设置为访问

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

{

minL=maxL;

j=0;

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

{

if(!u[m] && w[m]<minL)//很关键，得到到访问顶点到未访问顶点的最短路径以及对应顶点j

{

minL=w[m];

j=m;

b[j]=0;

}

}

u[j]=1;//把下一个顶点标为已访问

allL+=minL;

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

{

if(!u[m] && a[j][m]<w[m])//重新更新最短路径以及对应的顶点

{

w[m]=a[j][m];

b[m]=j;

}

}

}

cout<<allL;

}

int main()

{

int u[N],w[N],b[N],a[N][N];

int i=0,j=0;

int n;

cin>>n;

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

cin>>a[i][j];

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

a[i][j]=maxL;

}

}

prim(a,u,w,b,n);

return 0;

}

