**import** java.util.\*;

**public** **class** TSP

{

**public** **static** **void** main(String args[])

{

**int** c[][] = **new** **int**[10][10],tour[]=**new** **int**[10];

Scanner in = **new** Scanner(System.***in***);

**int** i,j,cost;

System.***out***.println("Enter the number of cities");

**int** n=in.nextInt();

**if**(n==1)

{

System.***out***.println("Path is not possible");

System.*exit*(0);

}

System.***out***.println("Enter the cost matrix");

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

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

c[i][j]=in.nextInt();

System.***out***.println("The entered cost matrix is ");

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

{

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

{

System.***out***.print(c[i][j]+"\t");

}

System.***out***.println();

}

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

tour[i]=i;

cost=*tspdp*(c,tour,1,n);

System.***out***.println("The accurate path is ");

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

System.***out***.print(tour[i]+"---->");

System.***out***.print("1");

}

**static** **int** tspdp(**int** c[][],**int** tour[],**int** start,**int** n)

{

**int** mintour[]=**new** **int**[10],temp[]=**new** **int**[10],mincost=999,cost,i,j,k;

**if**(start==n-1)

{

**return**(c[tour[n-1]][tour[n]]+c[tour[n]][1]);

}

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

{

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

temp[j]=tour[j];

temp[start+1]=tour[i];

temp[i]=tour[start+1];

**if**((c[tour[start]][tour[i]]+((cost=*tspdp*(c,temp,start+1,n)))<mincost))

{

mincost=c[tour[start]][tour[i]]+cost;

**for**(k=1;k<=n;k++)

mintour[k]=temp[k];

}

}

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

tour[i]=mintour[i];

**return** mincost;

}

}