#include<stdio.h>

#include<math.h>

#define size 10

int n,npow,g[size][100],p[size][100],adj[size][size];

int nrow;

int tsp(int source,int set)

{

int masked,mask,result = 999,temp,i;

if(g[source][set] != -1)

{

return g[source][set];

}

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

{

mask = (npow -1) - (1 << i);

masked=set&mask;

if(masked != set)

{

temp = adj[source][i] + tsp(i,masked);

if(temp < result)

{

result =temp;

p[source][set]=i;

}

}

}

return g[source][set] = result;

}

void getpath(int n,int npow,int result)

{

if(nrow == 1)

{

return ;

}

int i,j;

int val;

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

{

if(g[n][i] == result)

{

npow = i;

break;

}

}

result = g[n][npow];

val = p[n][npow];

printf("%d ->",val);

result = result - adj[n][val];

n = val;

nrow--;

getpath(n,npow,result);

}

void TSP()

{

int i,j,result;

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

{

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

{

g[i][j] = p[i][j] = -1;

}

}

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

{

g[i][0] = adj[i][0];

}

result = tsp(0,npow-2);

printf("Tour cost : %d \n",result);

printf("Tour path : 0 ->");

getpath(0,npow-2,result);

printf("0\n");

}

int main()

{

int i,j;

printf("Enter no of cities\n");

scanf("%d",&n);

nrow = n;

npow = (int)pow(2,n);

printf("npow = %d",npow);

printf("Enter the adjacency matrix\n");

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

{

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

{

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

}

}

TSP();

return 0;

}

OUTPUT:

