集合划分（Partition）

［问题描述］

给定一个集合 X = {x1, x2, x3...xn}。

定义函数 $D[x_u, x_v]$：$D[x_u, x_v] = D[x_v, x_u]$且 $D[x_u, x_u] = 0$。

一个 partition 是指一种将 X 划分为 K 个不相交的子集 $T = (C_1, C_2...C_K)$。C_P 是 X 的一个非空子集。

定义一个 partition 的费用 Cost(T) = min{D[u, v]}，其中 u 属于 C_p、v 属于 C_q 且有 p <>q。

[编程任务]

给定 N、K 和 D，求一个划分使其费用最大。

[输入文件]input.txt。

N K

然后一个 N * N 的矩阵，第 i 行 j 列描述 D[i, j]。

[输出文件]output.txt。

你所找到的最大费用。

[样例输入输出]

Input.txt

4 3

0 1 2 3

1 0 2 3

2 2 0 3

3 3 3 0

Output.txt

2

[数据约定]

1 < k <= n <= 200

0 <= D[u, v] <= 32000

对于 50%的数据满足 k <= 10

新的开始

【题目描述】

发展采矿业当然首先得有矿井， 小 FF 花了上次探险获得的千分之一的财富请人在岛上挖了 n 口矿井， 但他似乎忘记考虑的矿井供电问题……

为了保证电力的供应， 小 FF 想到了两种办法：

1、 在这一口矿井上建立一个发电站， 费用为 v（发电站的输出功率可以供给任意多个矿井）。

2、 将这口矿井与另外的已经有电力供应的矿井之间建立电网， 费用为 p。

小 FF 希望身为"NewBe_One"计划首席工程师的你帮他想出一个保证所有矿井电力供应的最小花费。

【输入格式】 第一行一个整数 n， 表示矿井总数。

第 2 ~ n+1 行 ,每行一个整数，第 i 个数 v[i]表示在第 i 口矿井上建立发电站的费用。

接下来为一个 n*n 的矩阵 P， 其中 p[i , j]表示在第 i 口矿井和第 j 口矿井之间建立电网的费用（数据保证有 p[i, j] = p[j, i], 且 p[i, i]=0 ）。

【输出格式】仅一个整数， 表示让所有矿井获得充足电能的最小花费。

【输入样例】

```
4
5
4
4
3
0 2 2 2
2 0 3 3
2 3 0 4
2 3 4 0
```

【输出样例】

```
9
```

输出样例说明：小 FF 可以选择在4号矿井建立发电站然后把所有矿井都与其建立电网，总花费是 3+2+2+2 = 9。

【数据范围】 对于30%的数据： 1<=n<=50;

对于100%的数据： 1<=n<=300; 0<=v[i], p[i,j] <=10^5.

公路建设（Road.exe， 1s， 64M）

【问题描述】

A 国是一个新兴的国家，有 N 个城市，分别编号为 1,2.3…N。政府想大搞公路建设，提供了优惠政策：对于每一个投资方案的预计总费用，政府负担 50%，并且允许投资的公司对过往的汽车收取连续 5 年的养路费。世界各地的大公司纷纷投资，并提出了自己的建设方案，他们的投资方案包括这些内容：公路连接的两座城市的编号，预计的总费用（假设他们的预计总师准确的）。你作为 A 国公路规划局的总工程师，有权利决定每一个方案是否接受。但是政府给你的要求是：

1）要保证各个城市之间都有公路直接或间接相连。

2）因为是新兴国家，政府的经济实力还不强。政府希望负担最少的费用。

因为大公司并不是同时提出方案，政府希望每接到一个方案，就可以知道当前需要负担的最小费用和接受的投资方案，以便随时开工。关于你给投资公司的回复可以等到开工以后再给。

注意：A 国一开始是没有公路的。

【数据说明】A 国的城市数目 N<=500，投资的方案总数 M<=2000。

【输入】输入文件名：Road.in

第 1 行有两个数字：N、M

第 2 行到第 M+1 行给出了各个投资方案，第 i 行的方案编号为 i-1

编号小的方案先接到，一个方案占一行，每行有 3 个数字，分别是连接的两个城市编号 a、b，和投资的预计总费用 cost。

【输出】输出文件名：Road.out

输出文件共有 M 行。

每一行的第一个数字是当前政府需要负担的最少费用（保留 1 位小数），后面是 X 个数字，表示当前政府接受的方案的编号，不要求从小到大排列。但如果此时接受的所有投资方案不能保证政府的第一条要求，那么这一行只有一个数字 0

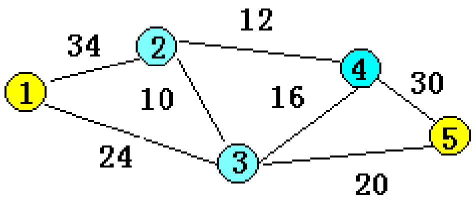
【样例】

Road.in	Road.out
3 5 1 2 4 1 3 4 2 3 4 1 3 2 1 2 2	0 4.00 1 2 4.00 1 2 3.00 1 4 2.00 4 5

城市交通

题目

某城市 有 N（1<=N<=50）个街区，某些街区由公共汽车线路相连，如在图 1 中，街区 1，2 有一条公共汽车线路相连，且由街区 1 至街区 2 的时间为 34 分钟。由于街区与街区之间的距离较近，与等车时间相比可忽略不记，所以这个时间为两趟公共汽车的间隔时间，即平均的等车时间。



（图1）

由街区 1 至街区 5 的最快走法为 1-3-5，总时间为 44 分钟。

现在市政府为了提高城市交通质量，决定加开 M(1<=M<=10)条公共汽车线路。若在某两个街区 A,B 之间加开线路（前提是 A、B 之间必须已有线路），则从 A 到 B 的旅行时间缩小为原来的一半（距离未变，只是等车的时间缩短了一半）。例如，若在 1，2 之间加开一条线路，则时间变为 17 分钟，加开两条线路，时间变为 8.5 分钟，以此类推。所有的线路都是环路，即如果由 1 至 2 的时间变为 17 分钟，则由 2 至 1 的时间也变为 17 分钟。

求加开某些线路，能使由城市 1 至城市 N 的时间最少。例如，在图 1 中，如果 M=2，则改变 1-3,3-5 的线路，总的时间可以减少为 22 分钟。

输入

输入文件名为 City.Inp。

第一行为城市数 N 与加开线路数 M。

第二行至第 N+1 行，每行为 N 个实数，第 I+1 行第 J 列表示由城市 I 到城市 J 的时间。

如果时间为 0，则城市 I 不可能到城市 J。

注意：输入数据中，从城市 1 到城市 N 至少有一条路线。

输出

输出文件名为 City.Out.

第一行为由城市 1 到城市 N 的最小时间 X（保留小数点后两位）。

第二行至第 M+1 行为更改的线路。每行由两个整数（x,y）构成。表示将城市

x 与城市 y 之间增加一条线路。

样例输入

```
City.Inp
5 2
0 34 24 0 0
34 0 10 12 0
24 10 0 16 20
0 12 16 0 30
0 0 20 30 0
```

样例输出

```
City.Out
22.00
1 3
3 5
```