1. 某次科研调查时得到了n个自然数，每个数均不超过1500000000（1.5\*10^9）。已知不相同的数不超过10000个，现在需要统计这些自然数各自出现的次数，并按照自然数从小到大的顺序输出统计结果。

【**要求**】

【**数据输入**】包含多个测试数据，每个包含n+1行：

第1行是整数n，表示自然数的个数。

第2~n+1行每行一个自然数。

1<=n<=200000，每个数均不超过1 500 000 000（1.5\*109）

【**数据输出**】对每个测试数据输出m行（m为n个自然数中不相同数的个数），按照自然数从小到大的顺序输出。每行输出两个整数，分别是自然数和该数出现的次数，其间用一个空格隔开。

相邻两个测试数据间用一个空行隔开。

【**样例输入**】

8

2

4

2

4

5

100

2

100

【**样例输出**】

2 3

4 2

5 1

100 2

1. 对于一个给定的n\*m的矩阵，矩阵中的每个元素aij均为非负整数。游戏规则如下：

1）每次取数时须从每行各取走一个元素，共n个。m次后取完矩阵所有元素；

2）每次取走的各个元素只能是该元素所在行的行首或行尾；

3）每次取数都有一个得分值，为每行取数的得分之和，每行取数的得分 = 被取走的元素值\*2i，其中i表示第i次取数（从1开始编号）；

4）游戏结束总得分为m次取数得分之和。

对于任意矩阵，可以求出取数后的最大得分。

【要求】

【数据输入】输入有多个测试数据，每个包括n+1行：

第1行为两个用空格隔开的整数n和m。

第2~n+1行为n\*m矩阵，其中每行有m个用单个空格隔开的非负整数。

1<=n, m<=80, 0<=aij<=1000

【数据输出】对每个数据，输出一行，为一个整数，即输入矩阵取数后的最大得分。相邻两个输出间用一个空行隔开。

【样例输入】

1 4

4 5 0 5

2 10

96 56 54 46 86 12 23 88 80 43

16 95 18 29 30 53 88 83 64 67

【样例输出】

122

316994

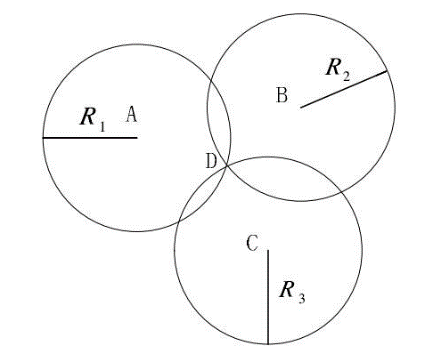
1. Trilateration（三边测量）是一种常用的定位算法：

已知三个信标点位置 (x1, y1), (x2, y2), (x3,y3)。已知未知点 (x, y) 到三点距离 d1, d2, d3。以 d1, d2, d3为半径作三个圆，根据毕达哥拉斯定理，得出交点即未知点的位置计算公式：

(x1 - x)2 + (y1 - y)2 = d12

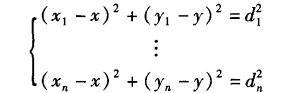
(x2 - x)2 + (y2 - y)2 = d22

(x3 - x)2 + (y3 - y)2 = d32



但在实际定位中，给定的距离由于测量的误差，并不能真正让三个圆交于一点，因此一种最基本的办法是利用最小二乘法求解方程组。由以下步骤求解：

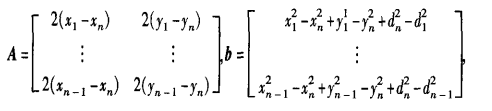
①：建立信标节点与未知节点距离方程组

[](https://github.com/megagao/IndoorPos/blob/master/image/%E4%BF%A1%E6%A0%87%E8%8A%82%E7%82%B9%E4%B8%8E%E6%9C%AA%E7%9F%A5%E8%8A%82%E7%82%B9%E8%B7%9D%E7%A6%BB%E6%96%B9%E7%A8%8B%E7%BB%84.png)

②：上边方程组为非线性方程组，用方程组中前n-1个方程减去第n个方程后，得到线性化的方程：

[得到的线性化方程](https://github.com/megagao/IndoorPos/blob/master/image/%E5%BE%97%E5%88%B0%E7%9A%84%E7%BA%BF%E6%80%A7%E5%8C%96%E6%96%B9%E7%A8%8B.png)

其中：

[](https://github.com/megagao/IndoorPos/blob/master/image/A%E3%80%81b%E7%9F%A9%E9%98%B5.png)

③：用最小二乘法求解上边方程得：[结果](https://github.com/megagao/IndoorPos/blob/master/image/%E7%BB%93%E6%9E%9C.png)

X 便是未知终端的坐标计算值。

【要求】

【数据输入】一组测试数据输入为两行，第一行输入是三个信标点位置 x1,y1,x2,y2,x3,y3，第二行输入是未知点到三点距离 d1, d2, d3。

【数据输出】输出一行，为两个数，即未知点的坐标(x, y)。