

Problem 1. hero

Input file: `hero.in`
Output file: `hero.out`
Time limit: 2s
Memory limit: 256M

SRwudi 最近正在和他的队友 SBwudi 玩守望先锋（在梦里）。

他们一共准备玩 N 局守望先锋，而守望先锋一共有 M 个英雄。

但由于每局地形地图模式不同，在每局里面操纵每个英雄的难度是不一样的，我们约定 $t_{i,j}$ 表示第 i 局操纵第 j 个英雄需要的难度值。SRwudi 和 SBwudi 两人水平相近，所以同一局打同一个英雄的难度值相同。

由于 SRwudi 和 SBwudi 是一对默契的搭档，由于不同英雄可以提升队伍能力，于是他们约定每局两个人都必须操控不同的英雄。

然而随意切换英雄会导致自己的键位操纵混乱，我们规定无论是 SRwudi 还是 SBwudi，除了第一局以外，如果上一局操控了第 a 号英雄，该局操控第 b 号英雄，那么要在总体难度上额外增加 $K \times |a - b|$ 的难度值。

显然他们两个希望整个游戏难度值总和最小。

Input

第一行包含三个正整数， N ， M ， K ($1 \leq K \leq 10^6$)。

后面 n 行描述包括一个矩阵，第 i 行第 j 列表示 $t_{i,j}$ ($1 \leq t_{i,j} \leq 10^6$) 表示第 i 局操纵第 j 个英雄需要的难度值。

Output

一个整数表示总体最小难度值

Example

hero.in	hero.out
3 3 10 2 13 4 4 3 2 16 4 3	28

Scoring

- Subtask1[20pts]: $1 \leq N, M \leq 50$
- Subtask2[80pts]: $1 \leq N, M \leq 300$

Problem 2. area

Input file: **area.in**
Output file: **area.out**
Time limit: 2s
Memory limit: 256M

球球有 N 个点，他希望找到不超过 4 个顶点，使得以这些顶点构成的凸多边形面积最大。

Input

第一行包括一个正整数 N 。

后面 N 行每行包括两个非负整数 $x_i, y_i (0 \leq x_i, y_i \leq 10^9)$, 代表每个点的坐标。

Output

一行表示最大面积，保留一位小数。

Example

area.in	area.out
5 4 0 6 1 4 7 5 7 6 2	10.0
8 0 103700 109900 0 846557200 0 846573800 45900 846573800 273263700 846482500 273275300 115800 273275300 0 273269100	231322890434565000.0

Scoring

- Subtask1[10pts]: $1 \leq N \leq 100$
- Subtask2[20pts]: $1 \leq N \leq 500$
- Subtask3[20pts]: $1 \leq N \leq 4000$
- Subtask4[50pts]: $1 \leq N \leq 10^5$

Problem 3. assign

Input file: **assign.in**
Output file: **assign.out**
Time limit: 1s
Memory limit: 256M

你要安排几年 NOIP 的赛程。

有 N 个选手, 选手标号从 0 到 $N - 1$ 。先要进行 NOIP 初赛。

每个选手都有两个整数属性 L_i, R_i ($L_i \leq R_i$)。

该选手在初赛中取得的成绩将会是区间 $[L_i, R_i]$ 的某一个整数, 且他取得不同的成绩的概率相等。

比完初赛之后, 我们将每个选手的分数从大到小排序之后的前 K 个选手进入复赛 (注意同分的情况下我们会让标号更小的选手进入复赛)。

今年的 NOIP 复赛有些特别, 有复赛 A 组和复赛 B 组。

当 K 个选手进入复赛以后, 我们以标号从小到大的顺序按 A, B, B, A, A, B, B, A, A..... 这样来安排他们的分组 (即确定进入复赛后, 选手的分数与分组无关)。

你现在要求出每个选手进入复赛 A 组以及进入复赛 B 组的概率是多少。

Input

第一行一个整数 N 。

第二行 N 个整数, 表示 L_i 。

第三行 N 个整数, 表示 R_i 。

第四行一个整数。表示 K

Output

输出有 N 行。每行两个实数。第 i 行的两个数分别表示标号为 $i-1$ 的选手进入复赛 A 组的概率和进入复赛 B 组的概率。保留 6 位小数。

Example

assign.in	assign.out
2 1 2 10 9 2	1.000000 0.000000 0.000000 1.000000
6 1 0 1 1 1 1 1 3 1 1 1 1 4	1.000000 0.000000 0.000000 0.750000 0.000000 1.000000 0.750000 0.250000 0.250000 0.000000 0.000000 0.000000
4 1 2 5 4 9 7 8 9 3	0.652778 0.000000 0.347222 0.212963 0.000000 0.930556 0.000000 0.856481

Explanation

样例 1:

无论选手考的怎样他们都能进入复赛。然后按照标号排,0 号一定是 A 组,1 号一定是 B 组。

样例 2:

会有两种可能的结果:

1 号选手以 0.75 的概率取得大于 0 的分数。进入复赛的选手是 0,1,2,3。进入 A 组的是 0 和 3, 进入 B 组的是 1 和 2。

1 号选手以 0.25 的概率取得 0 分。进入复赛的选手是 0,2,3,4。进入 A 组的是 0 和 4, 进入 B 组的是 2 和 3。

Scoring

- Subtask1[5pts]: $1 \leq N \leq 20, R_i - L_i < 2$
- Subtask2[5pts]: $1 \leq N \leq 20, \prod (R_i - L_i + 1) < 2 \times 10^6$
- Subtask3[10pts]: $1 \leq N \leq 5, 0 \leq L_i \leq R_i \leq 100$
- Subtask4[20pts]: $1 \leq N \leq 10, 0 \leq L_i \leq R_i \leq 200$
- Subtask5[30pts]: $1 \leq N \leq 20, 0 \leq L_i \leq R_i \leq 1000$
- Subtask6[30pts]: $1 \leq N \leq 25, 0 \leq L_i \leq R_i \leq 1000$

对于所有数据 $K \leq N$