

投影 (projection)

【问题描述】

换一个角度看，世界可能就不同。—— 小强

k 维空间中有 n 个黑点与 n 个白点。我们为每一个黑点确定一个互不相同的对应的白点，这样一共有 $n!$ 种对应方法。我们定义这 n 个黑点与 n 个白点之间的“移动距离”为，在所有的对应方法中，对应的黑点与白点之间的 n 个欧几里德距离的和的最小值。

例如：考虑一维中的三个黑点 $\{1,5,6\}$ 与三个白点 $\{2,3,4\}$ ，那么它们之间的移动距离为： $|1-2|+|5-3|+|6-4|=4$ 。你可以验证一下这确实是距离和最小的一种对应方法。

你得到了三维空间中的 n 个黑点与 n 个白点。你想把它们投影到一个 $k(1 \leq k \leq 2)$ 维子空间上。一维子空间就是三维空间中的一条直线，二维子空间则是三维空间中的一个平面。一个点在一个子空间中的投影点就是这个子空间中距离它最近的点。例如， $(0,0,0)$, $(1,1,0)$, $(1,0,0)$, $(0,1,0)$ 这四个点投影到 $x-y=0, z=0$ 这条直线上之后，得到的投影点是 $(0,0,0)$, $(1,1,0)$, $(0.5,0.5,0)$, $(0.5,0.5,0)$ 。

你希望这 n 个黑点和 n 个白点投影到这个 k 维子空间之后的移动距离最大。请你计算这个最大值除以 n 。

【输入格式】

从文件 *projection.in* 中读入数据。

每个测试文件中可能有多个测试用例，每个测试用例的格式如下：

第一行两个数 n 和 k ，表示点数以及降到的维度。

后面 $2n$ 行，每行三个数，表示点的坐标。

文件最后有一行包含两个数 $-1 -1$ 。

【输出格式】

输出到文件 *projection.out* 中。

对文件中的每个测试用例，输出一行一个数 (长度不要超过 30)，表示结果。

你的答案与参考答案的相对误差不超过 10^{-7} 时被认为是正确的。

只有一个文件中所有的测试用例的结果都是正确的才能获得这个测试文件所对应的分数。

【样例 1 输入】

```
2 1
0 0 0
1 1 0
```

```
0 1 0
1 0 0
-1 -1
```

【样例 1 输出】

0.70710678118655

【样例 2 输入】

```
4 2
0 0 0
0 1 1
1 0 1
1 1 0
0 0 1
0 1 0
1 0 0
1 1 1
-1 -1
```

【样例 2 输出】

0.73883404559321

【样例 3】

见选手目录下的 *projection/projection3.in* 与 *projection/projection3.ans*。

【数据规模】

对于 30% 的数据, $k = 1$, $n \leq 1000$, 所有的点的 z 都是 0。一个文件中最多包含一个测试用例。

对于另外 40% 的数据, $k = 1$, $n \leq 1000$, 所有的点的坐标都是 $[-1, 1]$ 之间均匀独立随机生成的, 一个文件中最多包含十个测试用例。

对于另外 30% 的数据, $k = 2$, $n \leq 20$, 所有的点的坐标都是 $[-1, 1]$ 之间均匀独立随机生成的, 一个文件中最多包含十个测试用例。

对于 100% 的数据, 点的坐标的范围都是 -1 到 $+1$ 之间的, 答案不小于 0.01。