

# NOIP 2019 模拟赛 Contest 14

diamond\_duke

题目名称	游戏	面积	删除
可执行文件名	game	area	delete
输入文件名	标准输入	标准输入	标准输入
输出文件名	标准输出	标准输出	标准输出
时间限制	1s	1s	1s
内存限制	512MB	512MB	512MB
子任务个数	4	4	5
题目类型	传统型	传统型	传统型

**请注意：** 评测时开启 O2 优化和 C++11 编译选项，栈空间限制同空间限制。

# 1 游戏

## 1.1 Problem Statement

小 D 和小 Y 正在玩游戏。

小 D 和小 Y 画出了一棵  $n$  个节点的树，这棵树的节点依次编号为  $1, 2, \dots, n$ 。

我们定义节点 1 为这棵树的根节点，定义**叶子节点**是指**除了根节点以外**，只与恰好一个节点相邻的节点。

初始时，在根节点处有一个棋子。小 Y 想要把棋子移动到叶子节点，而小 D 想要阻止他。

具体而言，小 D 和小 Y 会轮流进行操作，小 D 先操作。对于每次操作，小 D 会选择一个叶子把它涂黑，而小 Y 会将棋子向一个相邻的点移动。

如果小 Y 成功地将棋子移动到了**没有涂黑**的叶子，或小 D 成功地将**所有叶子都涂黑**了，那么游戏结束，达成目标的人获胜。

小 D 和小 Y 想要知道，如果他们都以最优策略进行游戏，那么谁会获胜。

但是他们并不会，请你帮帮他们。

## 1.2 Input Format

从标准输入读入数据。

第一行一个空格隔开的整数  $n$ ，表示树的节点数。

第二行  $n - 1$  个空格隔开的数  $p_2, p_3, \dots, p_n$ ，其中  $p_i$  表示节点  $i$  的父亲节点。

## 1.3 Output Format

向标准输出输出答案。

输出一行一个字符串，若小 D 获胜，则输出 D，否则输出 Y。

## 1.4 Sample 1

### 1.4.1 Input

```
3
1 1
```

### 1.4.2 Output

```
Y
```

## 1.5 Sample 2

见下发文件 `game/game2.in` 与 `game/game2.ans`。

## 1.6 Sample 3

见下发文件 `game/game3.in` 与 `game/game3.ans`。

## 1.7 Constraints

对于所有测试数据,  $2 \leq n \leq 10^5$ ,  $1 \leq p_i < i$ 。

- 子任务 1 (20 分):  $n \leq 10$ ;
- 子任务 2 (20 分):  $n \leq 500$ ;
- 子任务 3 (25 分):  $n \leq 5000$ ;
- 子任务 4 (35 分): 无特殊限制。

## 2 面积

### 2.1 Problem Statement

小 D 正在研究面积。

小 D 画了一个凸多边形，这个凸多边形共有  $n$  个顶点。在把顶点按照**顺时针**顺序排列后，第  $i$  个顶点的坐标是  $(x_i, y_i)$ 。

小 D 想要知道，如果他随机剪掉这个凸多边形的一些角，那么剩下的凸多边形面积会是多大。

具体而言，小 D 想要知道，从顶点中等概率随机选择  $k$  个顶点，并将这些点顺次连接后得到的凸多边形面积的期望。如果你不知道期望是什么，你可以把它理解为所有可能的情况中，得到的凸多边形面积的平均数。

但是他并不会，请你帮帮他。

### 2.2 Input Format

从标准输入读入数据。

第一行两个整数  $n, k$ ，表示凸多边形的顶点个数，以及选择的顶点个数。

接下来  $n$  行，每行两个**小数点后不超过 6 位的浮点数**  $x_i, y_i$ ，表示第  $i$  个顶点的坐标是  $(x_i, y_i)$ 。

### 2.3 Output Format

向标准输出输出答案。

输出一行一个浮点数，表示得到的凸多边形面积的期望。你的答案被认为是正确的，当且仅当相对误差或绝对误差不超过  $10^{-6}$ 。形式化地，若你的答案为  $a$  而参考答案为  $b$ ，则你的答案被认为是正确的当且仅当  $\min\{|a - b|, |a - b|/b\} \leq 10^{-6}$ 。

### 2.4 Sample 1

#### 2.4.1 Input

```
4 3
0 0
1 1
2 1
1 0
```

### 2.4.2 Output

0.5

## 2.5 Sample 2

见下发文件 `area/area2.in` 与 `area/area2.ans`。

## 2.6 Sample 3

见下发文件 `area/area3.in` 与 `area/area3.ans`。

## 2.7 Constraints

对于所有测试数据,  $3 \leq k \leq n \leq 2500$ ,  $-10 \leq x_i, y_i \leq 10$ , 保证顶点按照顺时针顺序给出, 保证任意三个顶点不共线, 保证输入的浮点数小数点后不超过 6 位。

- 子任务 1 (20 分):  $n \leq 20$ ;
- 子任务 2 (20 分):  $n \leq 50$ ;
- 子任务 3 (30 分):  $n \leq 500$ ;
- 子任务 4 (30 分): 无特殊限制。

## 3 删除

### 3.1 Problem Statement

小 D 正在玩删除游戏。

这个删除游戏中共有  $2n$  个点，他们被等距地排列成了两行，每行都有  $n$  个点。

在游戏的初始状态中，每个点都有一根绳子与另一侧的某个点相连。形式化地，存在一个  $1, 2, \dots, n$  的排列  $p_1, p_2, \dots, p_n$ ，使得第一行的第  $i$  个点与第二行的第  $p_i$  个点之间有绳子相连，而除此之外没有任何绳子存在。

小 D 每次可以付出一些代价来删除一根绳子。具体地，小 D 可以花费  $w_i$  的代价删除第一行第  $i$  个点以及第二行第  $p_i$  个点之间的绳子。

我们用  $(a, b)$  表示连接第一行第  $a$  个点以及第二行第  $b$  个点的绳子。我们称两根绳子  $(a, b)$  以及  $(c, d)$  **相交**，当且仅当  $a < c, b > d$ ，或  $a > c, b < d$  成立。

在删除的过程中，如果小 D 删除了一根绳子，那么所有与这根绳子相交的绳子也会同时会被删去（这不用花费任何额外的代价）。

小 D 想要知道，删除完所有绳子需要的最小代价是多少。

但是他并不会，请你帮帮他。

### 3.2 Input Format

从标准输入读入数据。

第一行一个整数  $n$ ，表示每一行的节点个数。

第二行  $n$  个空格隔开的整数  $p_1, p_2, \dots, p_n$ ，表示绳子的连接方式。

第三行  $n$  个空格隔开的整数  $w_1, w_2, \dots, w_n$ ，表示删除绳子的代价。

### 3.3 Output Format

向标准输出输出答案。

输出一行一个整数，表示删除完所有绳子的最小代价。

### 3.4 Sample 1

#### 3.4.1 Input

```
5
3 1 4 5 2
3 4 3 4 1
```

### 3.4.2 Output

5

### 3.5 Sample 2

见下发文件 `delete/delete2.in` 与 `delete/delete2.ans`。

### 3.6 Sample 3

见下发文件 `delete/delete3.in` 与 `delete/delete3.ans`。

### 3.7 Constraints

对于所有测试数据,  $1 \leq n \leq 2 \times 10^5$ ,  $p_1, p_2, \dots, p_n$  是一个  $1, 2, \dots, n$  的排列,  $1 \leq w_i \leq 10^4$ 。

- 子任务 1 (15 分):  $n \leq 10$ ;
- 子任务 2 (20 分):  $n \leq 1000$ ;
- 子任务 3 (20 分):  $|i - p_i| \leq 5$ ;
- 子任务 4 (25 分):  $n \leq 10^5$ ;
- 子任务 5 (20 分): 无特殊限制。