

# NOIP 模拟 Day1

Colin

## 1 题目概况

中文题目名称	计算几何	花花的聚会	文本编辑器
英文题目名称	geometry	party	editor
输入文件名	geometry.in	party.in	editor.in
输出文件名	geometry.out	party.out	editor.out
每个测试点时限	1 秒	1 秒	2 秒
测试点数目	20	10	10
每个测试点分值	5	10	10
附加样例文件	有	有	有
题目类型	传统	传统	传统
运行内存上限	128M	128M	128M

## 2 注意事项

- 考试时间：3.5 小时。
- 评测在 windows 下进行。
- 评测时将开启 -O2 优化。
- 部分题目涉及大规模文件操作，请尽可能提升程序输入输出操作的效率。

### 3 计算几何

#### 3.1 题意描述

花花对计算几何有着浓厚的兴趣。他经常对着平面直角坐标系发呆，思考一些有趣的问题。今天，他想到了一个十分有意思的题目：

首先，花花会在  $x$  轴正半轴和  $y$  轴正半轴分别挑选  $n$  个点。随后，他将  $x$  轴的点与  $y$  轴的点一一连接，形成  $n$  条线段，并保证任意两条线段不相交。花花确定这种连接方式有且仅有一种。最后，花花会给出  $m$  个询问。对于每个询问，将会给定一个点  $P(x_p, y_p)$ ，问线段  $OP$  ( $O$  为坐标原点) 与  $n$  条线段会产生多少个交点？

#### 3.2 输入格式

第 1 行包含一个正整数  $n$ ，表示线段的数量；

第 2 行包含  $n$  个正整数，表示花花在  $x$  轴选取的点的横坐标；

第 3 行包含  $n$  个正整数，表示花花在  $y$  轴选取的点的纵坐标；

第 4 行包含一个正整数  $m$ ，表示询问数量；

随后  $m$  行，每行包含两个正整数  $x_p$  和  $y_p$ ，表示询问中给定的点的横、纵坐标。

#### 3.3 输出格式

共  $m$  行，每行包含一个非负整数，表示你对这条询问给出的答案。

#### 3.4 样例输入

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

#### 3.5 样例输出

```
0
3
```

#### 3.6 样例解释

3 条线段分别为：(3,0) – (0,3)、(4,0) – (0,4)、(5,0) – (0,5)  
(0,0) – (1,1) 不与他们有交点，答案为 0。

$(0,0) - (3,3)$  与三条线段均有交点，答案为 3。

### 3.7 数据规模与约定

- 对于 40% 的数据：  $n, m \leq 10$ ;
- 另有 20% 的数据：  $n, m \leq 100$ ;
- 另有 20% 的数据：  $n, m \leq 1000$ ;
- 对于 100% 的数据：  $n, m \leq 10^5, 1 \leq x, y < 2^{31}$ 。

## 4 花花的聚会

### 4.1 题意描述

花花住在  $H$  国。 $H$  国有  $n$  个城市，其中 1 号城市为其首都。城市间有  $n - 1$  条单向道路。从任意一个城市出发，都可以沿着这些单向道路一路走到首都。事实上，从任何一个城市走到首都的路径是**唯一**的。

过路并不是免费的。想要通过某一条道路，你必须使用一次过路券。 $H$  国一共有  $m$  种过路券，每张过路券以三个整数表示： $v \ k \ w$ ：你可以在城市  $v$  以价格  $w$  买到一张过路券。这张券可以使用  $k$  次。这意味着，拿着这张券通过了  $k$  条道路之后，这张券就不能再使用了。

请注意你同一时间最多只能拥有**最多一张**过路券。但你可以随时撕掉手中已有的过路券，并且在所在的城市再买一张。

花花家在首都。他有  $q$  位朋友，他希望把这些朋友们都邀请到他家做客。所以他想要知道每位朋友要花多少路费。他的朋友们都很聪明，永远都会选择一条花费最少的方式到达首都。

花花需要准备晚餐去了，所以他没有时间亲自计算出朋友们将要花费的路费。你可以帮帮他么？

### 4.2 输入格式

输入的第一行包含两个空格隔开的整数  $n$  和  $m$ ，表示  $H$  国的城市数量和过路券的种数。

之后的  $n - 1$  行各自包含两个数  $a_i$  和  $b_i$ ，代表城市  $a_i$  到城市  $b_i$  间有一条单向道路。

之后的  $m$  行每行包括三个整数  $v_i, k_i$  和  $w_i$ ，表示一种过路券。

下一行包含一个整数  $q$ ，表示花花朋友的数量。

之后的  $q$  行各自包含一个整数，表示花花朋友的所在城市。

### 4.3 输出格式

输出共  $q$  行，每一行代表一位朋友的路费。

### 4.4 样例输入

```
7 7
3 1
2 1
7 6
6 3
5 3
4 3
7 2 3
```

7 1 1  
2 3 5  
3 6 2  
4 2 4  
5 3 10  
6 1 20  
3  
5  
6  
7

#### 4.5 样例输出

10  
22  
5

#### 4.6 样例解释

对于第一位朋友，他在 5 号城市只能购买一种过路券，花费 10 元并且可以使用 3 次。这足够他走到首都，因此总花费是 10 元。

对于第二位朋友，他在 6 号城市只能购买 20 元的过路券，并且只能使用一次。之后，他可以在 3 号城市购买 2 元，可以使用 3 次的过路券走到首都。总花费是 22 元。

对于第三位朋友，他在 7 号城市可以购买两种过路券。他可以花 3 元买一张可以使用 2 次的券，然后在 3 号城市再买一张 2 元，可以使用 3 次的券，走到首都。总花费是 5 元，而且其他的购买方式不会比这种更省钱。

#### 4.7 数据规模与约定

- 对于 40% 的数据：  $n, m, q \leq 10, w_i \leq 10$ ;
- 另有 20% 的数据：  $n, m, q \leq 500, w_i \leq 100$ ;
- 另有 20% 的数据：  $n, m, q \leq 5000, w_i \leq 1000$ ;
- 对于 100% 的数据：  $n, m, q \leq 10^5, w_i \leq 10000, 1 \leq v_i, k_i \leq n$ 。

## 5 文本编辑器

### 5.1 题意描述

九发明了一个完美的文本编辑器。这个编辑器拥有两个光标（cursor），所以九能够同时在两处地方插入和删除文本。这个编辑器除了正常的编辑功能以外，还有一些只有九才知道用的功能，例如翻转两个光标之间的文本。某一天，九把自己的完美文本编辑器给弄丢了，但是她还有好多好多文本需要处理。于是她想请聪明又智慧的你帮她实现完美文本编辑器的一些功能。

功能列表如下：

功能名称	命令格式	说明
< (move left)	< w	w 为一个字符，“L”或“R”，表示左光标还是右光标（下同）。 该命令将选定光标向左移动，如果已经是最左端则不移动。 命令执行成功时输出“T”，若光标已经在最左端，则输出“F”。
> (move right)	> w	w 同上。 与< 命令不同的是，该命令将光标向右移动。 命令执行成功时输出“T”，若光标已经在最右端，则输出“F”。
I (insert)	I w c	w 同上。 c 是一个可见字符( $33 \leq \text{ascii 码} \leq 126$ )，代表在该光标左侧插入该字符。 该命令始终输出“T”。
D (delete)	D w	w 同上。 代表删除该光标右侧的一个字符。 命令执行成功时输出“T”，若光标右侧没有字符输出“F”。
R (reverse)	R	代表翻转左光标和右光标之间的字符。 该命令只有左光标在右光标左侧时才能执行。 (两光标重合时也不能执行) 命令执行成功时输出“T”，否则输“F”。
S (show)	S	代表显示当前处理的文本。 该命令只输出文本，不输出“T”和“F”。

开始时文本编辑器中有一定内容，左光标在第一个字符左，右光标在最后一个字符右。

注意：在插入和删除操作中，没有被操作的光标与文本的相对左右位置保持不变。特别地，若两个光标重叠，操作后也仍然重叠。

### 5.2 输入格式

第一行是初始时文本编辑器内容。

第二行是一个正整数  $N$ ， $N$  表示操作次数。

接下来有  $N$  行，每行有一个命令，命令格式如上方表格。

### 5.3 输出格式

对于每个命令，按上方表格要求执行并输出。

### 5.4 样例输入

```
goodykc
11
I R u
I R l
> L
> L
> L
> L
R
D R
< R
D R
S
```

### 5.5 样例输出

```
T
T
T
T
T
T
T
T
F
T
T
goodluck
```

## 5.6 样例解释

```
[goodykc]
[goodykcu]
[goodykcul]
g[oodykcul]
go[odykcul]
goo[dykcul]
good[ykcul]
good[lucky]
good[lucky]（光标右边没有字符，失败删除）
good[luck]y
good[luck]
goodluck
```

## 5.7 数据规模与约定

- 对于 40% 的数据： $1 \leq N$ ，初始文本长度  $\leq 100$ ，数据不包含翻转（Reverse）操作；
- 另有 30% 的数据： $1 \leq N$ ，初始文本长度  $\leq 10^5$ ，数据不包含翻转（Reverse）操作；
- 另有 20% 的数据： $1 \leq N$ ，初始文本长度  $\leq 10^5$ ，数据包含翻转（Reverse）操作；
- 对于 100% 的数据： $1 \leq N$ ，初始文本长度  $\leq 4 \times 10^6$ ，输出文件大小  $\leq 20MB$ ；