

---

# 全国信息学奥林匹克联赛（NOIP2019）复赛

## 模拟试题

题目名称	一样远	拆网线	喝红茶
英文代号	equal	tree	tea
输入文件名	equal.in	tree.in	tea.in
输出文件名	equal.out	tree.out	tea.out
时限	1s	1s	1.5s
空间限制	256M	256M	256M
测试点数目	20	20	10
测试点分值	5	5	10
题目类型	传统	传统	传统

测评机环境：Intel®Core i5-5200 CPU @2.20GHz 2.20GHz, 4G 内存

注意：避免测试时出现编译错误，请加上如下常见的头文件：

```
#include <cstdio>

#include <cstring>

#include <iostream>

#include <algorithm>

#include <bits/stdc++.h>
```

---

## 1. 一样远

(equal.pas/c/cpp)

### 【问题描述】

企鹅国的城市结构是一棵树，有  $N$  座城市和  $N-1$  条无向道路，每条道路都一样长。Smart 和 Sarah 准备去参加 NOIP (National Olympiad in Informatics for Penguin)，但是他们住在不同的地方，Smart 住在城市 A，Sarah 住在城市 B。他们想找一个距离 A 和 B 一样远的集合地点，所以他们想知道有多少个城市满足这个要求？

由于他们会参加很多次 NOIP，所以有很多个询问。

### 【输入格式】

第一行一个整数  $N$  代表城市个数。

接下来  $N-1$  行，每行两个数字  $F$  和  $T$ ，表示城市  $F$  和城市  $T$  之间有一条道路。

接下来一行一个整数  $M$  代表询问次数。

接下来  $M$  行，每行两个数字  $A$  和  $B$ ，表示这次询问的城市  $A$  和城市  $B$  ( $A$  可能与  $B$  相同)。

### 【输出格式】

输出  $M$  行，每行一个整数表示到  $A$  和  $B$  一样远的城市个数。

### 【样例】

equal.in	equal.out
4	0
1 2	2

---

2 3	
2 4	
2	
1 2	
1 3	

**【数据范围】**

30%的数据：  $N, M \leq 1000$ ;

另外 10%的数据：  $A=B$ ;

另外 30%的数据： 保证树的形态随机;

100%的数据：  $1 \leq N, M \leq 100000$ 。

## 2. 拆网线

(tree.pas/c/cpp)

**【问题描述】**

企鹅国的网吧们之间由网线互相连接，形成一棵树的结构。现在由于冬天到了，供暖部门缺少燃料，于是他们决定去拆一些网线来做燃料。但是现在有  $K$  只企鹅要上网和别人联机游戏，所以他们需要把这  $K$  只企鹅安排到不同的机房（两只企鹅在同一个机房会吵架），然后拆掉一些网线，但是需要保证每只企鹅至少还能通过留下来的网线和至少另一只企鹅联机游戏。

---

所以他们想知道，最少需要保留多少根网线？

**【输入格式】**

第一行一个整数  $T$ ，表示数据组数；

每组数据第一行两个整数  $N, K$ ，表示总共的机房数目和企鹅数目。

第二行  $N-1$  个整数，第  $i$  个整数  $A_i$  表示机房  $i+1$  和机房  $A_i$  有一根网线连接( $1 \leq A_i \leq i$ )。

**【输出格式】**

每组数据输出一行包含一个整数表示最少保留的网线数目。

**【样例】**

tree.in	tree.out
2	2
4 4	2
1 2 3	
4 3	
1 1 1	

**【数据范围】**

30%的数据：  $N \leq 15$ ；

50%的数据：  $N \leq 300$ ；

70%的数据：  $N \leq 2000$ ；

100%的数据：  $2 \leq K \leq N \leq 100000, T \leq 10$ 。

---

### 3. 喝红茶

(tea.pas/c/cpp)

#### 【问题描述】

Sarah 喜欢喝红茶，一天她要求 Smart 开卡车帮她运红茶过来。

红茶其实是编好号了的，从 0 开始一直到无穷……

一开始卡车上是没有红茶的，然后接下来到企鹅国的路上有  $m$  个时刻，每个时刻都会发生一种事情。

第一种事件，Smart 到了一个红茶店，买了一个编号为  $x$  的红茶（之前不会买过相同编号的红茶）。

第二种事件，一个目前在卡车上的编号为  $x$  的红茶飞出了卡车。

第三种事件，Smart 把目前不在卡车上的最早飞出去的红茶捡回了卡车上。

第四种事情，Smart 清点红茶，希望找到一个最小的编号，这种编号的红茶并不在卡车上。

#### 【输入格式】

第一行两个整数分别表示  $m$  和  $czy$ 。 $czy=1$  表示进行强制在线，等于 0 则允许离线。

接下来  $m$  行，每行表示一个事件，首先会读入事件编号。

如果编号为 1，还要再读入一个数表示加入的红茶的编号。 $czy=1$  时请把这个编号异或上一次的答案。

如果编号为 2，还要再读入一个数表示飞出去的红茶的编号。 $czy=1$  时请把这个编号异或上一次的答案。

---

编号为 3 表示捡红茶。

编号为 4 表示询问。

【输出格式】

对于编号为 4 的事件输出一行表示答案。

【样例】

tea.in	tea.out
10 0	1
1 0	4
1 1	5
1 2	
1 3	
2 1	
4	
3	
4	
1 4	
4	

【数据范围】

本题采取捆绑数据测试。

Task1(20 分):

$M \leq 1000$ ，红茶的编号在 1000 以内，czy=0;

Task2(20 分)

---

$m \leq 100000$ ，红茶的编号在 100000 以内， $czy=0$ ；

Task3(20 分)

$m \leq 10000000$ ，红茶的编号在 10000000 以内， $czy=0$ ；

Task4(40 分)：

$m \leq 10000000$ ，红茶的编号在 10000000 以内， $czy=1$ 。

提示：由于数据范围很大，请采用快速读入方式读取输入数据。