

# 2024 省夏令营树结构模拟

时间：2024 年 7 月 3 日 8:00~11:30

题目名称	最浅二叉树	管道运输	松鼠家的灯	大雪封路
题目类型	传统型	传统型	传统型	传统型
可执行文件名	T1	T2	T3	T4
输入文件名	T1.in	T2.in	T3.in	T4.in
输出文件名	T1.out	T2.out	T3.out	T4.out
每个测试点时限	1 秒	1 秒	2 秒	1 秒
内存限制	512MB	512MB	512MB	512MB
子任务数目	10	10	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++语言	T1.cpp	T2.cpp	T3.cpp	T4.cpp
----------	--------	--------	--------	--------

编译选项

对于 C++语言	-O2 -std=c++14
----------	----------------

## 注意事项

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C++中函数 `main()` 的返回值类型必须是 `int`。程序正常结束时的返回值必须是 `0`。
3. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
4. 程序可使用的栈空间内存限制与题目的内存限制一致。
5. 选手提交的程序源文件必须不大于 100KB。

# 最浅二叉树 (T1)

【题目描述】

二叉树的特点是树中的每个结点最多只有两个孩子结点。现在有一个二叉树，提供给你三个整数  $a,b,c$ ，分别代表着有关于这一棵二叉树的三个信息：

- 1) 该二叉树中有且仅有  $a$  个结点有两个孩子结点
- 2) 该二叉树中有且仅有  $b$  个结点有一个孩子结点
- 3) 该二叉树中有且仅有  $c$  个结点没有孩子结点

请找出满足这三个条件的所有二叉树中，**深度最小**的二叉树的深度为多少。本题中深度为：我们定义树中两个顶点之间的距离是它们之间**最短路径的边数**，树的高度就是在所有顶点到树根顶点的距离中取**最大距离**。

特别的，如果没有任何一个二叉树可以同时满足这三个条件，那么输出“-1”。

【输入格式】

从文件 T1.in 中读入数据。  
第一行输入一个正整数  $T$ ，表示数据组数。  
接下来  $T$  行，每行 3 个整数  $a,b,c$ ，代表一棵二叉树的三个信息。

【输出格式】

输出到文件 T1.out 中。  
共  $T$  行，每行一个整数，表示满足三个信息的二叉树的最小深度，若没有任何二叉树能够满足三个信息，则输出-1。

【样例 1 输入】

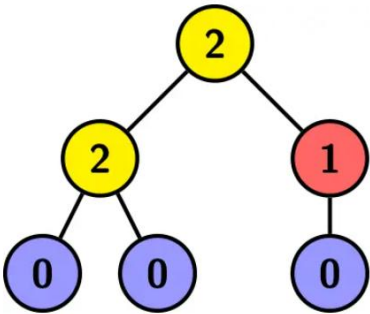
```
10
2 1 3
0 0 1
0 1 1
1 0 2
1 1 3
3 1 4
8 17 9
24 36 48
1 0 0
0 3 1
```

【样例 1 输出】

```
2
0
1
1
-1
3
```

6
-1
-1
3

**样例解释：**对于第一组数据  $a=2,b=1,c=3$ ，我们可以建立如下图所示的一棵二叉树，这棵树的深度为 2（任选一个没有孩子的结点到根结点的距离最远，因为经过了两条边，所以深度为 2）。易证明，这是一种满足三个信息的同时深度最浅的二叉树，没有比深度为 2 更小的结果。



**【数据范围】**

测试点	T	$a+b+c$	特殊性质
1-3	$\leq 100$	$\leq 20$	
4-5	$\leq 10^4$	$\leq 3 \cdot 10^5$	$a=0$
6-7	$\leq 10^4$	$\leq 3 \cdot 10^5$	$b=0$
8-10	$\leq 10^4$	$\leq 3 \cdot 10^5$	

## 管道运输（T2）

### 【题目描述】

油国有  $N$  个城市，通过  $N-1$  条管道来在各个城市之间运输油，任意两座城市之间都通过管道连通。

油国一共有  $Q$  条运输油的路线，第  $i$  条路线从城市  $u_i$  运输到  $v_i$ 。一条运输路线会给它的起止城市以及中间途径的所有城市都带来  $k_i$  的运输压力。油国需要知道每个城市的运输压力。

### 【输入格式】

从文件 `T2.in` 中读入数据。

第一行一个正整数  $n$ ，代表油国城市个数。

第  $2\sim n$  行，每行两个整数  $u_i, v_i$ ，表示一根连接  $u_i, v_i$  两个城市的管道。

第  $n+1$  行一个正整数  $Q$ ，代表运输路线的数量。

接下来  $q$  行，每行三个整数  $s_i, t_i, k_i$ ，表示有一条从城市  $s_i$  到城市  $t_i$  运输压力为  $k_i$  的运输线路。

### 【输出格式】

输出到文件 `T2.out` 中。

输出共  $n$  行，每行一个整数，按照编号从  $1\sim n$  的顺序依次输出每个城市的运输压力。

### 【样例 1 输入】

```
5
3 4
1 5
4 2
5 4
10
5 4 1
5 4 1
3 5 1
4 3 1
4 3 1
1 3 1
3 5 1
5 4 1
1 5 1
3 4 1
```

### 【样例 1 输出】

```
2
0
6
```

9
7

【样例 2】  
见文件

【数据范围】

测试点	$n \leq$	$q \leq$	$k \leq$	特殊性质
1	20	100	$10^5$	
2-3	3000	3000	$10^9$	
4-5	$2 \cdot 10^4$	$10^5$	$10^9$	第 $i$ 条管道连接 $i$ 和 $i+1$ 两个城市 ( $u_i=i, v_i=i+1$ 或 $u_i=i+1, v_i=i$ )
6-7	$5 \cdot 10^4$	$10^5$	$10^9$	保证有一个城市是所有运输线路的起点或者终点
8-10	$5 \cdot 10^4$	$10^5$	$10^9$	

# 松鼠家的灯（T3）

【题目描述】

松鼠的家非常大，一共有  $n$  个房间。这  $n$  个房间的构成可以看作是一棵树，其中 1 号房间是树根。每个房间内都有一盏灯，每盏灯最初的时候可能亮着，也可能灭着。每个房间内也有一个按钮，按下按钮后，以该房间为子树根的子树内的所有房间的灯的状态都会发生变化，即原本亮着的会灭掉，原本灭着的会亮起来。松鼠会依次进行  $q$  次操作，操作分为两种：

- 1) `get x`，询问以  $x$  房间作为子树根的子树内一共有多少盏灯是亮着的
- 2) `pow x`，按下  $x$  房间的按钮

你需要输出所有 `get` 操作的答案。

【输入格式】

从文件 `T3.in` 中读入数据。  
第一行一个整数  $n$ ，代表房间总数量。  
第二行共  $n-1$  个整数，第  $i$  个数  $x_i$  表示在树中  $x_i$  房间是  $i+1$  房间的父亲结点  
第三行给出每个房间灯的初始状态，1 代表开着，0 代表灭着。  
第四行一个整数  $q$ ，接下来  $q$  行，每行一次操作。

【输出格式】

按照顺序输出所有 `get` 询问操作的答案，一行一个。

【样例 1 输入】

```
4
1 1 1
1 0 0 1
9
get 1
get 2
get 3
get 4
pow 1
get 1
get 2
get 3
get 4
```

【样例 1 输出】

```
2
0
0
1
2
```

1
1
0

【样例 2】

见文件

【数据范围】

每个测试点的具体限制见下表：

测试点编号	$n \leq$	$q \leq$	特殊性质
1	50	50	
2	500	500	
3-4	3000	3000	
5-8	$2 \cdot 10^5$	$2 \cdot 10^5$	所有的 <code>get</code> 在所有的 <code>pow</code> 之后
9-12	$2 \cdot 10^4$	$2 \cdot 10^5$	$X_i = i$
13-20	$2 \cdot 10^5$	$2 \cdot 10^5$	

# 大雪封路（T4）

【题目描述】

雪国有  $N$  个城市，通过  $N-1$  条公路相互连通，即可以将雪国看作是一棵  $N$  个结点  $N-1$  条边的树。每条公路都有自己的长度。两个城市之间的距离  $\text{dist}(u,v)$  定义为：从  $u$  到  $v$  的简单路径边权和。特别的，我们规定  $\text{dist}(u,u)=0$ 。

雪国目前正在遭受暴雪。现在有  $q$  个时刻，每个时刻过后，都会有一条公路因为暴雪的原因无法通行。显然每次有公路无法通行，都会使得城市之间的连通状况发生改变，仍能够相互连通的城市看作是一个连通块。你要求出每个时刻后每个连通块内距离最远两个点的距离的最大值。形式化的，每次操作后，你要求出

$$\max_{c \in C} \{ \max_{u,v \in c} \text{dist}(u,v) \}$$

其中， $C$  为当前所有的连通块构成的集合。

【输入格式】

从文件 `T4.in` 中读取数据。

第一行一个整数  $T$ ，表示数据组数，对每组数据：

第一行两个整数  $n,q$ ，依次表示雪国的城市数和出现大雪封路的时刻数量。

接下来  $n-1$  行，每行三个整数  $u,v,w$ ，表示上有一条连接城市  $u$  和城市  $v$  的长度为  $w$  的公路。

接下来  $q$  行，每行一个整数  $e_i$ ，表示该时刻后，第  $e_i$  条公路被大雪封住无法通行，保证每组数据内， $e_i$  不会重复出现。

【输出格式】

输出到文件 `T4.out` 中。

对每组数据输出  $q$  行，每行一个整数，依次表示每个时刻后所求的答案。

【样例 1 输入】

```
2
4 2
1 2 1
2 3 2
3 4 3
2
3
12 2
1 2 1
2 3 1
1 4 3
2 5 4
5 6 3
5 7 2
7 8 1
```



8 9 1
9 10 1
7 11 5
8 12 3
4
6

【样例 1 输出】

3
1
10
9

**样例解释：**对于第一组数据，4 个城市构成 1-2-3-4 一条链状。  
第一个时刻被封的是 2 号公路，此时连通块有 2 个，分别为{1,2},{3,4}，其中{1,2}内城市间最长距离为 1，{3,4}内城市间最长距离为 3， $\max(1,3)=3$ ，输出 3。  
第二个时刻被封的是 3 号公路，此时连通块有 3 个，分别为{1,2},{3},{4}，其中{1,2}内城市间最长距离为 1，剩余连通块内城市间最长距离为 0， $\max(1,0,0)=1$ ，输出 1。

【样例 2】

见文件

【数据范围】

每个测试点的具体限制见下表：

测试点编号	$T \leq$	每组 $q < n \leq$	特殊性质
1-2	100	100	
3-4	10	2000	
5-8	10	$10^4$	所有城市构成一条链
9-12	5	$3 \cdot 10^4$	
13-20	5		

所有的  $w \leq 10^5$ 。对测试点 13-20，保证  $\sum n \leq 3 \cdot 10^5$ 。