

2049: [Sdoi2008]Cave 洞穴勘测

Time Limit: 10 Sec Memory Limit: 259 MB
Submit: 7820 Solved: 3652
[\[Submit\]](#)[\[Status\]](#)[\[Discuss\]](#)

Description

辉辉热衷于洞穴勘测。某天，他按照地图来到了一片被标记为JSZX的洞穴群地区。经过初步勘测，辉辉发现这片区域由 n 个洞穴（分别编号为1到 n ）以及若干通道组成，并且每条通道连接了恰好两个洞穴。假如两个洞穴可以通过一条或者多条通道按一定顺序连接起来，那么这两个洞穴就是连通的，按顺序连接在一起的这些通道则被称之为这两个洞穴之间的一条路径。洞穴都十分坚固无法破坏，然而通道不太稳定，时常因为外界影响而发生改变，比如，根据有关仪器的监测结果，123号洞穴和127号洞穴之间有时会出现一条通道，有时这条通道又会因为某种稀奇古怪的原因被毁。辉辉有一台监测仪器可以实时将通道的每一次改变状况在辉辉手边的终端机上显示：如果监测到洞穴 u 和洞穴 v 之间出现了一条通道，终端机上会显示一条指令 **Connect u v** 如果监测到洞穴 u 和洞穴 v 之间的通道被毁，终端机上会显示一条指令 **Destroy u v** 经过长期的艰苦卓绝的手工推算，辉辉发现一个奇怪的现象：无论通道怎么改变，任意时刻任意两个洞穴之间至多只有一条路径。因而，辉辉坚信这是由于某种本质规律的支配导致的。因而，辉辉更加夜以继日地坚守在终端机之前，试图通过通道的改变情况来研究这条本质规律。然而，终于有一天，辉辉在堆积成山的演算纸中崩溃了……他把终端机往地面一砸（终端机也足够坚固无法破坏），转而求助于你，说道：“你老兄把这程序写写吧”。辉辉希望能随时通过终端机发出指令 **Query u v** ，向监测仪询问此时洞穴 u 和洞穴 v 是否连通。现在你要为他编写程序回答每一次询问。已知在第一条指令显示之前，JSZX洞穴群中没有任何通道存在。

Input

第一行为两个正整数 n 和 m ，分别表示洞穴的个数和终端机上出现过的指令的个数。以下 m 行，依次表示终端机上出现的各条指令。每行开头是一个表示指令种类的字符串 s （"Connect"、"Destroy"或者"Query"，区分大小写），之后有两个整数 u 和 v （ $1 \leq u, v \leq n$ 且 $u \neq v$ ）分别表示两个洞穴的编号。

Output

对每个Query指令，输出洞穴 u 和洞穴 v 是否互相连通：是输出"Yes"，否则输出"No"。（不含双引号）

Sample Input

样例输入1 cave.in

200 5

Query 123 127

Connect 123 127

Query 123 127

Destroy 127 123

Query 123 127

样例输入2 cave.in

3 5

Connect 1 2

Connect 3 1

Query 2 3

Destroy 1 3

Query 2 3

Sample Output

样例输出1 cave.out

No

Yes

No

样例输出2 cave.out

Yes

No

HINT

数据说明 10%的数据满足 $n \leq 1000$, $m \leq 20000$ 20%的数据满足 $n \leq 2000$, $m \leq 40000$
30%的数据满足 $n \leq 3000$, $m \leq 60000$ 40%的数据满足 $n \leq 4000$, $m \leq 80000$ 50%的数据满足 $n \leq 5000$, $m \leq 100000$ 60%的数据满足 $n \leq 6000$, $m \leq 120000$ 70%的数据满足 $n \leq 7000$, $m \leq 140000$ 80%的数据满足 $n \leq 8000$, $m \leq 160000$ 90%的数据满足 $n \leq 9000$, $m \leq 180000$ 100%的数据满足 $n \leq 10000$, $m \leq 200000$ 保证所有Destroy指令将摧毁的是一条存在的通道
本题输入、输出规模比较大，建议c/c++选手使用scanf和printf进行IO操作以免超时

Source

[\[Submit\]](#)[\[Status\]](#)[\[Discuss\]](#)