

War Game

Submission Date: Thursday 23rd of March, 2017

A war is being fought between two countries, A and B . As a loyal citizen of C , you decide to help your country by secretly attending the peace talks between A and B . There are n other people at the talks, but you do not know which person belongs to which country. You can see people talking to each other, and by observing their behavior during occasional one-to-one conversations you can guess if they are friends or enemies.

Your country needs to know whether certain pairs of people are from the same country, or whether they are enemies. You can expect to receive such questions from your government during the peace talks, and will have to give replies on the basis of your observations so far.

Now, more formally, consider a black box with the following operations:

<code>setFriends(x,y)</code>	shows that x and y are from the same country
<code>setEnemies(x,y)</code>	shows that x and y are from different countries
<code>areFriends(x,y)</code>	returns true if you are sure that x and y are friends
<code>areEnemies(x,y)</code>	returns true if you are sure that x and y are enemies

The first two operations should signal an error if they contradict your former knowledge. The two relations “friends” (denoted by \sim) and “enemies” (denoted by $*$) have the following properties:

\sim is an equivalence relation: i.e.,

1. If $x \sim y$ and $y \sim z$, then $x \sim z$ (The friends of my friends are my friends as well.)
2. If $x \sim y$, then $y \sim x$ (Friendship is mutual.)
3. $x \sim x$ (Everyone is a friend of himself.)

$*$ is symmetric and irreflexive:

1. If $x * y$ then $y * x$ (Hatred is mutual.)
2. Not $x * x$ (Nobody is an enemy of himself.)
3. If $x * y$ and $y * z$ then $x \sim z$ (A common enemy makes two people friends.)
4. If $x \sim y$ and $y * z$ then $x * z$ (An enemy of a friend is an enemy.)

Operations `setFriends(x,y)` and `setEnemies(x,y)` must preserve these properties.

Input

The first line contains a single integer, n , the number of people. Each subsequent line contains a triple of integers, $c\ x\ y$, where c is the code of the operation,

$c=1$, setFriends
 $c=2$, setEnemies
 $c=3$, areFriends
 $c=4$, areEnemies

and x and y are its parameters, integers in the range $[0, n)$ identifying two different people. The last line contains 0 0 0.

All integers in the input file are separated by at least one space or line break. There are at most 10,000 people, but the number of operations is unconstrained.

Output

For every areFriends and areEnemies operation write “0” (meaning no) or “1” (meaning yes) to the output. For every setFriends or setEnemies operation which conflicts with previous knowledge, output a “-1” to the output; such an operation should produce no other effect and execution should continue. A successful setFriends or setEnemies gives no output.

All integers in the output file must be separated by one line break.

<i>Sample Input</i>	<i>Sample Output</i>
10	1
1 0 1	0
1 1 2	1
2 0 5	0
3 0 2	0
3 8 9	-1
4 1 5	0
4 1 2	
4 8 9	
1 8 9	
1 5 2	
3 5 2	
0 0 0	