4 Baffling Boolean Bulbs

The memorial of the Big Earth's Disaster of 2077 aims to be the most energy efficient building of the whole EIRE (Environmentally-Insensitive Reckless Empire). To save money on light switches they decided to only having switches in a single central control. Unfortunately, they also chose to save electricity by not installing windows or cameras. Therefore, the lighting manager has no way of seeing which lights are on or off.

However, he has been given the schematics for the lighting circuits. Between the light switches and the light bulbs is a complicated network of boolean logic gates. This network consists of both switches as well as light bulbs, and some number of Boolean logic gates which are connected in between. The signal from switches and gates are fed through the network until they reach a bulb.

The set of Boolean gates[1] consists of AND, OR, XOR, NOR, and NAND. Each of these gates takes two input signals and outputs a single signal. These signals are either an *ON* signal, represented by 1, or an *OFF* signal, represented by 0. Gates will take inputs from switches or other gates, and the output will connect to a gate or a light bulb. Output from a gate or switch can be used as input multiple times. Finally the bulbs take a single input. If the bulb receives an *ON* signal, the bulb will light up. If it receives an *OFF* the bulb will remain unlit. Bulbs do not output signals themselves.

With the knowledge of which switches are turned-on and the particular lighting schematics, the unfortunate lighting manager sets out to determine which lights are on and which lights are off. Can you write a software to ease his work?

Input The first line contains three integers $2 \leq L \leq 5000$; $0 \leq B \leq 50000$ and $0 \leq G \leq 50000$. L represents the number of light switches, B is the number of light bulbs, and G represents the number of gates in the network.

Each of the next L lines contain two integers i and s. The first integer i, $1 \leq i \leq (L+B+G)$, is the identifier for a light switch. All identifiers are unique across the switches, bulbs, and gates. The second integer $s \in \{0,1\}$, is the state of the gate, with 1 representing *ON* and 0 representing *OFF*. Each of the next B lines contain two integers i and j. The first integer i, $1 \leq i \leq (L+B+G)$, is the identifier for a light bulb. The second integer j, $1 \leq j \leq (L+B+G)$, is the id of the input signal source for the light bulb.

The next G lines contain four integers representing a gate in the circuit: i, j, k, and t, with $1 \leq i,j,k \leq (L+B+G)$ and $t \in \{1,2,3,4,5\}$. i is the identifier of the gate, j and k are the identifiers of the input signal source of the gate. Finally, t represents the type of the gate. Table 1 lists the types of gates and their respective Boolean truth-tables.

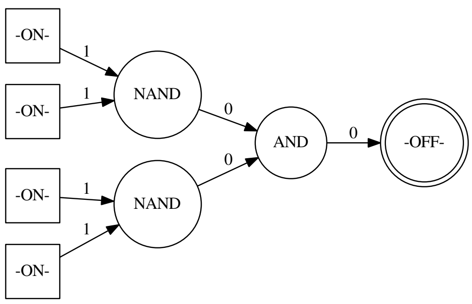

Figure 2: Sample Input 1

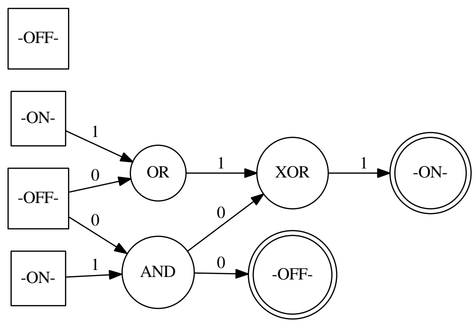

Figure 3: Sample Input 2

Output The output consists of a single line of space separated integers, representing the state of the lights in the circuit. 0 represents a light in the *OFF* state, and 1 a light in the *ON* state. The lights should be sorted in ascending order according to their identifier. As before, the output should immediately be followed by a single newline character.

[1]Boolean algebra was introduced by George Boole, the first Math Professor of UCC

Gate Type		1	2	3	4	5
Input		AND	OR	XOR	NOR	NAND
0	0	0	0	0	1	1
0	1	0	1	1	0	1
1	0	0	1	1	0	1
1	1	1	1	0	0	0

Table 1: Output values for logic gates

Sample Input 1

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

Sample Output 1

```
0
```

Sample Input 2

```
4 2 3
1 1
2 0
3 1
4 0
8 7
9 6
5 1 2 2
6 2 3 1
7 5 6 3
```

Sample Output 2

```
1 0
```