Problem C - Apple Crusher

Lucca has way too many apples, so he's goint to juice them with a fancy apple crushing machine he bought off Amazon. Unfortuantely, it's a large industrial sized crusher, so its extremely complicated to operate.

The crusher has a central unit that connects to many other units. Other than the central unit, every other unit has exactly one parent unit. Formally, the machine is a rooted tree where the central unit is located at the root. Unit A is said to be *dependent* on unit B if unit B is an ancestor of unit A.

There are a total of 60 possible different operations that any unit can do. Each unit can only do one operation at any given time. These have the uninteresting names of operation 1, operation 2, ... , operation 60.

Lucca needs to modify the state of the juicer to make it work properly. Lucca has figured out how to set a unit and all units dependent on that unit to do one of the 60 operations. In fact this is the only modification he's figured out how to do! Fortunately, this is sufficient for him to operate his juicer.

Lucca ocassionally needs to check that he's followed the instructions correctly. To check if unit A is working correctly, he needs to figure out how many different types of operations are done by all units dependent on unit A.

Input

The first line contains a single integer T denoting the number of test cases.

Each test case begins with a line containing two integers n ($1 \leq n \leq 200,000$) representing the number of units, and m ($1 \leq m \leq 200,000$) denoting the number of modifications and checks Lucca performs.

The next line contains n integers $r_i (1 \leq r_i \leq 60)$ denoting the type of operation that is being done in the ith unit initially.

The next $n-1$ line contain two integers x, y ($1 \leq x, y \leq n$) denoting that describes an edge in the juicer tree. These edges are guarenteed to form a single connected tree. **The edges as given are not necessarily in the direction of parent to child**. The tree is rooted at unit 1.

The last m lines in the test case contains several integers each describing one of the actions Lucca has done:

- 1 A R indicates that Lucca has set unit A and all units dependent on unit A to do operation R.
- 2 A indicates that Lucca is checking on unit A

This list describes the order of actions Lucca has done. Units and operations are numbered from 1.

Output

For each action that Lucca has done of the form $2A$ ouput a single line containing a single integer - the number of different operations currently being done by modules dependent on module A.

Sample Input

```
1
10 17
1 2 3 4 5 6 7 8 9 10
1 2
1 3
1 4
2 9
2 10
3 5
3 6
6 7
6 8
2 1
2 3
2 2
2 4
1 8 3
2 3
1 6 6
2 6
1 3 1
1 6 2
1 7 1
2 1
2 3
2 6
2 7
1 2 60
2 1
```

Sample Output

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