Task: ROZ
Crossroads of parity

Olimpiada Informatyczna

XXIV OI, Stage III, Day two. Source file `roz.*` Available memory: 256 MB. *12.04.2017*

Byteasar is administering a newly constituted county in the ever expanding kingdom of Byteotia. The county consists of n towns and no roads — the county is so recent that it has no infrastructure yet! One of Byteasar's projects was building m bidirectional roads, each directly linking a pair of towns. Were the plan fully implemented, every city could be reached by road from every other. Byteasar's lack of experience however resulted in huge inefficiencies: expert engineers estimated that each successive road's construction would cost more than all the previous ones together. Specifically, it was reported to Byteasar that the i-th road's construction cost is exactly 2^i bythalers.

Due to limited budget, progress is slow. So slow in fact, that a new fashion took hold in Byteotia – the citizens are suddenly obsessed with parity. Citizens of some towns believe that even numbers symbolize harmony, conformity, and peace, and consequently require that the number of roads leaving their towns be even. At the same time, citizens of the remaining towns believe that odd numbers symbolize independence, individualism, and resourcefulness, hence requiring that the number of roads leaving their towns be odd.

Now Byteasar has to come up with a new road network project, choosing only a subset of previously designed roads, so that he satisfies the wishes of all citizens. Naturally, he intends to minimize its cost. The Byteotian law on public tenders stipulates that $k-1$ cheapest offers are discarded, so Byteasar is actually looking for the k-th cheapest road network that satisfies the citizens' demands. Note that unlike Byteasar, they only care about parity, i.e., such network need not connect all the towns.

Moreover, the parliament of Byteotia alters the law on public tenders quite often, changing the value of parameter k, whereas the citizens frequently change their opinion on parity (independently in each town), suddenly favoring even numbers instead of odd or vice versa. Poor Byteasar has to adjust his project to each and every change of circumstances. Help him in proving his generosity, management skills, and ease of following fickle trends by telling him which road networks he should construct!

Input

In the first line of the standard input, there are two integers n and m ($1 \leq n, m \leq 500\,000$) that specify, respectively, the number of towns and of potential roads that may be build in the county.

Next, m lines follow, which describe the originally designed roads, i.e., those that may be built. The i-th such line contains two integers a_i, b_i ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$), which indicate that it is possible to construct a bidirectional road directly linking towns a_i and b_i at a cost of 2^i bythalers. Every pair of towns appears on input at most once.

In the following line, there are n integers $p_1, \ldots, p_n$, separated by single spaces. If $p_i = 0$, then the citizens of the i-th town love even numbers, whereas $p_i = 1$ means they rather favor odd numbers.

The next line contains a single integer k ($1 \leq k \leq 10^{18}$) that specifies that Byteasar should initially determine the k-th cheapest road network that satisfies the citizens' wishes (i.e., when $k = 1$, he should find the cheapest one). Notice that every pair of different networks differ in cost.

Next, there is a line with a single integer q ($0 \leq q \leq 500\,000$) that specifies the number of queries, after which q lines describing those queries follow. A query's description consists of a character c specifying a type of the query, followed by a number v. If $c =$ `M`, then the citizens of town no. v ($1 \leq v \leq n$) have just changed their preferred parity. If $c =$ `K`, then the parliament has altered the bill, and from now on Byteasar should determine the v-th cheapest road network ($1 \leq v \leq 10^{18}$). Finally, $c =$ `D` means that Byteasar is asking you to tell if the road no. v ($1 \leq v \leq m$), i.e., the one of cost 2^v, is part of the currently sought road network.

Output

In the first line of the standard output, a description of the initially sought road network, i.e., before any change of circumstance, should be printed – you may assume that such network always exists. The description should consist of m integers separated by single spaces, where the i-th number is to indicate if the i-th road should be built, i.e., equal 1 if it should be built and to 0 otherwise. Successive queries of type `D` should be answered in the following lines in the same format, with the following exception: If the desired road network does not exist due to insufficient number of different networks conforming with the citizen's requirement, the number -1 should be printed. Note that the number of conforming networks may even be 0. Otherwise, if the sought network exists, the numbers 1 and 0 should be used as before, i.e., respectively when the road belongs to said network and when it does not.

Example

For the input data:

```
3 3
1 2
2 3
3 1
1 1 0
1
7
D 1
M 2
D 1
M 3
D 2
K 2
D 2
```

the correct result is:

```
1 0 0
1
-1
1
0
```

Explanation for the example: There are three towns and three roads, which form a cycle. The cheapest network such that the numbers of roads leaving towns 1 and 2 are each odd, and those leaving town 3 even, contains only the 1–2 road (of cost 2). There is no network with exactly one city with an odd number of roads leaving it. There are exactly two different networks such that the numbers of roads leaving towns 1 and 3 are each odd (and those leaving town 2 even): the cheaper of two networks consists of the roads 1–2 and 2–3 (of total cost $2 + 4 = 6$), whereas the more expensive one consists of the single road 1–3 (of cost 8).

Sample grading tests:

- `1ocen`: $n = 6$, $m = 15$, $q = 50$, projected roads between every pair of cities, no queries of type `K`;
- `2ocen`: $n = 10$, $m = 10$, $q = 84$, projected roads are arranged on a cycle; upon every query of type `D` there are exactly two conforming networks;
- `3ocen`: $n = 101$, $m = 150$, 50 triangles, each sharing a single vertex with the previous one; all requirements are even, $k = 2^{50}$, no queries.

Grading

The set of tests consists of the following subsets. Within each subset, there may be several test groups.

Subset	Property	Score
1	$k = 1$, $q = 0$ (no queries once the cheapest network is found)	32
2	$q = 0$ (no queries)	25
3	all queries are of type `M` or `D`	30
4	no additional property	13