Unique Encryption Keys

`unique.c`, `unique.C`, `unique.java`

The security of many ciphers strongly depends on the fact that the keys are unique and never re-used. This may be vitally important, since a relatively strong cipher may be broken if the same key is used to encrypt several different messages.

In this problem, we will try to detect repeating (duplicate) usage of keys. Given a sequence of keys used to encrypt messages, your task is to determine what keys have been used repeatedly in some specified period.

Input Specification

The input contains several cipher descriptions. Each description starts with one line containing two integer numbers M and Q separated by a space. M ($1 \leq M \leq 1\,000\,000$) is the number of encrypted messages, Q is the number of queries ($0 \leq Q \leq 1\,000\,000$).

Each of the following M lines contains one number K_i ($0 \leq K_i \leq 2^{30}$) specifying the identifier of a key used to encrypt the i-th message. The next Q lines then contain one query each. Each query is specified by two integer numbers B_j and E_j, $1 \leq B_j \leq E_j \leq M$, giving the interval of messages we want to check.

There is one empty line after each description. The input is terminated by a line containing two zeros in place of the numbers M and Q.

Output Specification

For each query, print one line of output. The line should contain the string "`OK`" if all keys used to encrypt messages between B_j and E_j (inclusive) are mutually different (that means, they have different identifiers). If some of the keys have been used repeatedly, print one identifier of *any* such key.

Print one empty line after each cipher description.

Sample Input

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

5 2
1
2
3
1
2
2 4
1 5

0 0
```

Output for Sample Input

```
3
OK
4
3
OK

OK
1
```