XOR Minimum

Time limit: 1.0s **Memory limit:** 128M
Java: 1.4s

Given a 0-indexed list of N non-negative integers, and Q queries each consisting of one non-negative integer x, output the smallest index of the minimum value in the list after each number in the list is XOR'd with x for each query. Queries are persistent. That is, any previous queries change the list directly and affect the current query's answer.

Input Specification

The first line will contain two space-separated integers, N $(1 \leq N \leq 10^5)$, and Q $(1 \leq Q \leq 10^5)$.

The second line will contain N space-separated integers, $a_0, a_1, \ldots, a_{N-1}$ $(0 \leq a_i \leq 10^9)$.

The next Q lines will each contain one integer x $(0 \leq x \leq 10^9)$.

Output Specification

For each query, output the smallest index of the minimum value in the list after every number in the list has been XOR'd with x.

Constraints

For all subtasks:

$1 \leq N, Q \leq 10^5$

Subtask 1 [5/17]

$1 \leq N, Q \leq 10^4$

Subtask 2 [12/17]

No additional constraints.

Sample Input 1

```
8 4
2 15 8 1 6 2 16 8
0
2
10
7
```

Sample Output 1

```
3
0
2
1
```

Explanation for Sample Output 1

```
2 15 8 1 6 2 16 8
```

The array is unchanged after the first XOR operation. The minimum is simply 1 at index 3.

```
0 13 10 3 4 0 18 10
```

In this case, the first of the two zeroes is chosen (index 0).

```
10 7 0 9 14 10 24 0
```

The first occurrence of the minimum value (0) is at index 2.

```
13 0 7 14 9 13 31 7
```

The minimum value (0) is at index 1.

Sample Input 2

```
7 5
7 7 8 0 0 10 16
12
14
16
6
12
```

Sample Output 2

```
2
3
6
6
6
```