D - Dynamic Tree

Description

You are given a **complete** binary tree consisting of n nodes whose number is from 1 to n.
The root is node 1. For node $i(i > 1)$, its father is node $\lfloor \frac{i}{2} \rfloor$.
Initially, the weight of node i is a_i where $\{a_i\}_{i=1}^{n}$ is a permutation of number 1 to n.

We define an operation as follows:
We set out from the root. When we are at node x,we will move to node y when x is the father of y and **the weight of y is bigger than the weight of another son of** x .(if x has only one son y , then move to y). The process will end at a leaf.

Let the path be $p_1, p_2, \cdots, p_k (p_1 = 1)$, and then we will rotate the weight on the path towards the root, which will make the weight on the path from $a_{p_1}, a_{p_2}, \ldots a_{p_k}$ to $a_{p_2}, a_{p_3}, \ldots a_{p_1}$. In other word, the new value of a_{p_i} equals to the previous value of $a_{p_{i+1}}$ for $i = 1, \cdots, k-1$ and the new value of a_{p_k} equals to the previous value of a_{p_1}.

You should process q queries.
Each query contains two integers m and i. For each query, you should output a_i after m operations.

Input

The first line of Input contains two integers n and q $(1 \leq n, q \leq 10^5)$

The second line of Input contains n integers, representing a_i $(1 \leq i, a_i \leq n)$

Each of the next q lines contains two integers m and i $(0 \leq m \leq 10^{18}, 1 \leq i \leq n)$

It's guaranteed that the permutation is generated randomly.

Output

You should output the required answers for the queries in one line, and they should be seperated by one space.

Sample Input

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

Sample Output

```
1 4 3 1
```

NOTE

After 1 operation, $\{a_i\}_{i=1}^n = [5, 4, 3, 2, 1]$;

After 2 operations, $\{a_i\}_{i=1}^n = [4, 2, 3, 5, 1]$.