E. Army Creation

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

As you might remember from our previous rounds, Vova really likes computer games. Now he is playing a strategy game known as Rage of Empires.

In the game Vova can hire n different warriors; ith warrior has the type a_i. Vova wants to create a *balanced* army hiring some subset of warriors. An army is called *balanced* if for each type of warrior present in the game there are not more than k warriors of this type in the army. Of course, Vova wants his army to be as large as possible.

To make things more complicated, Vova has to consider q different plans of creating his army. ith plan allows him to hire only warriors whose numbers are not less than l_i and not greater than r_i.

Help Vova to determine the largest size of a *balanced* army for each plan.

Be aware that the plans are given in a modified way. See input section for details.

Input

The first line contains two integers n and k ($1 \le n, k \le 100000$).

The second line contains n integers $a_1, a_2, \ldots a_n$ ($1 \le a_i \le 100000$).

The third line contains one integer q ($1 \le q \le 100000$).

Then q lines follow. ith line contains two numbers x_i and y_i which represent ith plan ($1 \le x_i, y_i \le n$).

You have to keep track of the answer to the last plan (let's call it $last$). In the beginning $last = 0$. Then to restore values of l_i and r_i for the ith plan, you have to do the following:

1. $l_i = ((x_i + last) \bmod n) + 1$;
2. $r_i = ((y_i + last) \bmod n) + 1$;
3. If $l_i > r_i$, swap l_i and r_i.

Output

Print q numbers. ith number must be equal to the maximum size of a *balanced* army when considering ith plan.

Example

input
6 2 1 1 1 2 2 2 5 1 6 4 3 1 1 2 6 2 6

output
2 4 1 3 2

Note

In the first example the real plans are:

1. 1 2
2. 1 6
3. 6 6
4. 2 4
5. 4 6