

Waiter

Problem Statements :

You are a waiter at a party. There are N stacked plates on pile A_0 . Each plate has a number written on it. Then there will be Q iterations. In i -th iteration, you start picking the plates in A_{i-1} from the top one by one and check whether the number written on the plate is divisible by the i -th prime. If the number is divisible, you stack that plate on pile B_i . Otherwise you stack that plate on pile A_i . After Q iterations, plates can only be on pile $B_1, B_2, B_3, \dots, B_Q, A_Q$. Output numbers on these plates from top to bottom of each piles in order of $B_1, B_2, B_3, \dots, B_Q, A_Q$.

Problem Source : <https://www.hackerrank.com/challenges/waiter/problem>

Input Format :

First line contains two space separated integers, N and Q .

The next line contains N space separated integers representing the initial pile of plates, i.e. A_0 .

The leftmost value represents the bottom plate of the pile.

Constraints :

$$1 < N < 5 \cdot 10^4$$

$$2 < A_i, B_i < 10^4$$

Output Format :

Output N lines. Each line contains a number written on the plate. Printing should be done in the order defined above.

Sample Input :

```
5 1
3 4 7 6 5
```

Sample Output :

```
4
6
3
```

7
5

Explanation :

Initially:

$$A_0 = [3, 4, 7, 6, 5] \leftarrow \text{TOP}$$

After 1 iteration:

$$A_0 = [] \leftarrow \text{TOP}$$

$$B_1 = [6, 4] \leftarrow \text{TOP}$$

$$A_1 = [5, 7, 3] \leftarrow \text{TOP}$$

We should output numbers in B_1 first from top to bottom, and then output numbers in A_1 from top to bottom.

Time Limit :