

7212 Just a bit sorted

Jurgen Guntherswarchzhaffenstrassen is known for his virtuous guitar playing and the cruel teaching methods he employs with his students. What most people ignore about him is that he is also a fan of numbers.

Lately Jurgen has been studying sorted lists, but he is getting bored. He thinks that such lists are too predictable and not very abundant, so he decided to spice things up a bit.

Jurgen says that a list ℓ of N not necessarily different positive integers is just a bit sorted if and only if for each positive integer $x > 1$ that occurs in ℓ , the number $x - 1$ appears at least once before the last occurrence of x in ℓ . For example

- $[2, 3, 1, 2]$ is just a bit sorted because a 1 appears before the last 2, and a 2 appears before the last 3;
- $[2, 3, 4, 3, 2, 1, 3, 4]$ is not just a bit sorted because every 1 appears after the last 2;
- $[1, 1, 3, 1, 3, 3, 1, 3]$ is not just a bit sorted because no 2 appears before the last 3 (since 2 doesn't appear at all in this list).

Jurgen is trying to find out how many different just a bit sorted lists of N positive integers not greater than K exist. Two lists are different if and only if there is at least one position in which the lists have distinct elements. Can you help Jurgen in counting the number of different lists?

Input

The input contains several test cases; each test case is formatted as follows. The first line contains two integers N and Q , representing respectively the number of elements in the just a bit sorted lists and the number of queries to answer ($1 \leq N \leq 5000$ and $1 \leq Q \leq 1000$). The second line contains Q integers K_1, K_2, \dots, K_Q , indicating that the lists you must count in the i -th query cannot contain values greater than K_i ($1 \leq K_i \leq 10^9$ for $i = 1, 2, \dots, Q$).

Output

Output a line with Q integers, such that the i -th integer represents the number of different just a bit sorted lists of N positive integers not greater than K_i (for $i = 1, 2, \dots, Q$). Since this number can be very large, output the remainder of dividing it by $10^9 + 7$.

Sample Input

```
1 1
1
3 4
2 2 1 10
1000 3
100 5 300
```

Sample Output

```
1
5 5 1 6
265428620 285047952 668355714
```