

UCF Local Contest — August 31, 2013

LIS Number

filename: lis

Let A be a sequence of integers. The *LIS Number* of A is the smallest positive integer L such that A can be obtained by concatenating L strictly increasing sequences. For example, the *LIS Number* of $A = \{1, 4, 4, 2, 6, 3\}$ is 4, since we can obtain A as $\{1, 4\} + \{4\} + \{2, 6\} + \{3\}$, and there is no way to create A by concatenating 3 (or fewer) strictly increasing sequences. The *LIS Number* of a strictly increasing sequence is 1.

The Problem:

You are given a sequence of length N and an integer K . You want to transform the given sequence into a sequence with *LIS Number* K . The only operation you are allowed to do is to delete 0 or more numbers from the original sequence. Count how many ways you can do that. Two ways are different if the set of removed numbers (their indices/positions) are different.

The Input:

The first input line contains a positive integer, t , indicating the number of test cases. First line of each test case consists of two integers N ($1 \leq N \leq 50,000$) and K ($1 \leq K \leq 10$). The second line contains N integers of the sequence (separated by a single space). These integers will be between 0 and 100000, inclusive.

The Output:

For each test case, output the number of ways you can transform the given sequence of length N into a sequence with *LIS Number* K . Since the number of ways can be too large, output the result modulo 1,000,000,007.

(Sample Input/Output on the next page)

Sample Input:

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

Sample Output:

```
31
5
10
0
```