**Increasing Speed Limits**

**Problem**

You were driving along a highway when you got caught by the road police for speeding.

It turns out that they've been following you, and they were amazed by the fact that you were accelerating the whole time without using the brakes! And now you desperately need an excuse to explain that.

You've decided that it would be reasonable to say "all the speed limit signs I saw were in increasing order, that's why I've been accelerating".

The police officer laughs in reply, and tells you all the signs that are placed along the segment of highway you drove, and says that's unlikely that you were so lucky just to see some part of these signs that were in increasing order.

Now you need to estimate that likelihood, or, in other words, find out how many different subsequences of the given sequence are strictly increasing.

The empty subsequence does not count since that would imply you didn't look at any speed limits signs at all!

For example, (1, 2, 5) is an increasing subsequence of (1, 4, 2, 3, 5, 5), and we count it twice because there are two ways to select (1, 2, 5) from the list.

**Input**

The first line of input gives the number of cases, N. N test cases follow.

The first line of each case contains n, m, X, Y and Z each separated by a space.

‘n’ will be the length of the sequence of speed limits. m will be the length of the generating array A.

The next m lines will contain the m elements of A, one integer per line (from A[0] to A[m-1]).

Using A, X, Y and Z, the following pseudo code will print the speed limit sequence in order.

'mod' indicates the remainder operation.

for i = 0 to n-1

print A[i mod m]

A[i mod m] = (X \* A[i mod m] + Y \* (i + 1)) mod Z

Note: The way that the input is generated has nothing to do with the intended solution and exists solely to keep the size of the input files low.

**Output**

For each test case you should output one line containing "T: S" (quotes for clarity)

where T is the number of the test case and S is the number of non-empty increasing subsequences mod 1000000007. M = 1000000007.

**Limits**

1 ≤ N ≤ 20

1 ≤ m ≤ 100

0 ≤ X ≤ 109

0 ≤ Y ≤ 109

1 ≤ Z ≤ 109

0 ≤ A[i] < Z

Small dataset

1 ≤ m ≤ n ≤ 1000

Large dataset

1 ≤ m ≤ n ≤ 500000

**Sample**

Input

2

5 5 0 0 5

1

2

1

2

3

6 2 2 1000000000 6

1

2

Output

1: 15

2: 13

The sequence of speed limit signs for case 2 should be 1, 2, 0, 0, 0, 4.