Yesterday, my teacher taught us about math: +, -, *, /, GCD, LCM... As you know, LCM (Least common multiple) of two positive numbers can be solved easily because of

\[ a * b = GCD(a, b) * LCM(a, b) \]

In class, I raised a new idea: ”how to calculate the LCM of \( K \) numbers”. It’s also an easy problem indeed, which only cost me 1 minute to solve it. I raised my hand and told teacher about my outstanding algorithm. Teacher just smiled and smiled ...

After class, my teacher gave me a new problem and he wanted me solve it in 1 minute, too. If we know three parameters \( N, M, K \), and two equations:

1. \( SUM(A_1, A_2, \ldots, A_i, A_{i+1}, \ldots, A_K) = N \)
2. \( LCM(A_1, A_2, \ldots, A_i, A_{i+1}, \ldots, A_K) = M \)

Can you calculate how many kinds of solutions are there for \( A_i \) (\( A_i \) are all positive numbers). I began to roll cold sweat but teacher just smiled and smiled.

Can you solve this problem in 1 minute?

**Input**

There are multiple test cases.

Each test case contains three integers \( N, M, K \). (\( 1 \leq N, M \leq 1,000, 1 \leq K \leq 100 \))

**Output**

For each test case, output an integer indicating the number of solution modulo 1,000,000,007(1e9 + 7).

You can get more details in the sample and hint below.

**Hint:**

The first test case: the only solution is (2, 2).

The second test case: the solution are (1, 2) and (2, 1).

**Sample Input**

4 2 2
3 2 2

**Sample Output**

1
2