

Factored Aggregates (2/4/8 Points)

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

points: 100 · problem code: factoredaggregates · author: glipR

For this problem, we need to define three functions:

- $F(n)$: The total number of factors of n . ($F(10) = 4$ [1, 2, 5, 10])
- $F(n, k)$: The total number of factors of n , which are also divisible by k . ($F(10, 2) = 2$ [2, 10])
- $F(n, k, j)$: The total number of factors of n , which are also divisible by k , but *not* divisible by j . ($F(10, 2, 5) = 1$ [2])

In this problem, depending on the type of query, we need you to compute one of the following values:

In type 1:

$$F(n)$$

In type 2:

$$\sum_{k=1}^n F(n, k)$$

In type 3:

$$\sum_{k=1}^n \sum_{j=1}^n F(n, k, j)$$

Since the answer may be large, output your result modulo $10^9 + 7$.

Input

Input will contain two space-separated integers, t and n .

t Will be the type of query (1, 2, or 3).

Output

Output the value of one of the three values above depending on the value of t . Output your result modulo $10^9 + 7$.

Constraints

- $1 \leq n \leq 10^9$
- For 25% of test cases (Batch #1), **Only Type 1 Queries will feature.**
- For 25% of test cases (Batch #2), **Only Type 2 Queries will feature.**
- For 50% of test cases (Batch #3), **Only Type 3 Queries will feature.**

Example 1

For input

```
1 1000
```

Your program should output `16`, since there are 16 unique factors of 1000 (1, 2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 125, 200, 250, 500, 1000)

Example 2 (Not tested on judge to avoid short-circuiting)

For input

2 1000

Your program should output 100 .

Example 3 (Not tested on judge to avoid short-circuiting)

For input

3 1000

Your program should output 99100 .