Problem L – Problem L

Hugo is learning to factorize numbers into primes, and he's going to make a program to test his new algorithm. First, he will factorize every integer from 1 to N and save every factorization. Then, he will rebuild every integer in that range from its factorization, and add up all the results (the number 1 is rebuilt as 1). Of course, he will get as result $N(N + 1)/2$ if his algorithm is correct.

But he made a silly mistake in the second part: suppose that he's rebuilding the integer m with prime factorization $p_1{}^{a_1} p_2{}^{a_2} \cdots$, then he really will add $a_1{}^{p_1} a_2{}^{p_2} \cdots$ to the answer, because he accidentally flipped the base and its exponent in every prime.

He's obtaining very strange results, and you are curious to see how far is his wrong answer from the correct one.

Input

The first line contains the positive integer N ($1 \leq N \leq 10^{14}$).

Output

Output a line containing the wrong answer obtained by flipping the bases and the exponents of every prime in every factorization. Since this number can be very large, output it modulo $10^9 + 7$.

Sample input 1	Sample output 1
5	8

Sample input 2	Sample output 2
10	28

Sample input 3	Sample output 3
20	66

Notes:

The first 10 positive integers are rebuilt as:

- $1 \rightarrow 1$

- $2 = 2^1 \rightarrow 1^2 = 1$

- $3 = 3^1 \rightarrow 1^3 = 1$

- $4 = 2^2 \rightarrow 2^2 = 4$

- $5 = 5^1 \rightarrow 1^5 = 1$

- $6 = 2^1 \times 3^1 \rightarrow 1^2 \times 1^3 = 1$

- $7 = 7^1 \rightarrow 1^7 = 1$

- $8 = 2^3 \rightarrow 3^2 = 9$

- $9 = 3^2 \rightarrow 2^3 = 8$

- $10 = 2^1 \times 5^1 \rightarrow 1^2 \times 1^5 = 1$

Hence the answer for the second example is $1 + 1 + 1 + 4 + 1 + 1 + 1 + 9 + 8 + 1 = 28$.