

Totient in permutation (hard)

In number theory, Euler's totient (or PHI function), is an arithmetic function that counts the number of positive integers less than or equal to a positive integer N that are relatively prime to this number N .

That is, if N is a positive integer, then $\text{PHI}(N)$ is the number of integers K for which $\text{GCD}(N, K) = 1$ and $1 \leq K \leq N$. We denote GCD the Greatest Common Divisor. For example, we have $\text{PHI}(9)=6$.

Interestingly, $\text{PHI}(87109)=79180$, and it can be seen that 87109 is a permutation of 79180.

Input

There is only one number M .

Output

For the given M , you have to print on a single line the value of N , for which $1 < N < M$, $\text{PHI}(N)$ is a permutation of N and the ratio $N/\text{PHI}(N)$ produces a minimum. If there's several answers output the greatest, or if need, "No solution." without quotes.

Leading zeros are not allowed for integers greater than 0.

Example

Input:

22

Output:

21

Input:

222

Output:

63

Input:

2222

Output:

291

Explanations : For the first case, in the range $]1..22[$, the lonely number n for which $\text{phi}(n)$ is in permutations(n) is 21, (we have $\text{phi}(21)=12$). So the answer is obviously 21.

For the second case, in the range $]1..222[$, there's two numbers n for which $\text{phi}(n)$ is in permutations(n), we have $\text{phi}(21)=12$ and $\text{phi}(63)=36$. But as $63/36$ is equal to $21/12$, we're taking the greater : 63.

For the third case, in the range $]1..2222[$, there's four numbers n for which $\text{phi}(n)$ is in permutations(n), $\text{phi}(21)=12$, $\text{phi}(63)=36$, $\text{phi}(291)=192$ and $\text{phi}(502)=250$. Within those solutions $291/192$ is the minimum, we output 291.

Constraints

$$1 < M < 10^{27}$$

If you got TLE, you should consider before [TIP2](#). Enjoy.

Warning : don't try to investigate the input number, [judge](#) is strict and interactive ; test case is randomly changing, staying equivalent in difficulty. Please don't use spurious spaces and end your answer with '\n' ; e.g. "21\n" awaited for the sample.

There is many ways to optimize the solution for this problem, to get AC here, you'll need to find many of them.

Time limit allows python3 solutions.

There is different judges, time is the sum of them.