

# H: Anagrams

Memory limit: **128 MB**

An *anagram* of a word  $w$  is any word obtained by rearranging (permuting) letters of  $w$ , including word  $w$  itself. Anagrams are a passion of Johnny ever since he started playing Scrabble: whenever he has at his disposal some collection of letters arranged as a word  $w$  he immediately starts rearranging them in all possible ways, counting how many anagrams the word  $w$  has. Apparently Johnny is more interested in anagrams than Scrabble, as it doesn't matter for him at all, whether any of  $w$ 's anagrams has a meaning and occurs in dictionary. Johnny quickly noticed that from different words of the same length he can obtain different numbers of anagrams. When he decided that he knows how to efficiently compute the number of anagrams of a given word he started to ponder the opposite problem, that is, how long is the shortest word consisting only of small Latin letters (a-z) that has exactly  $n$  anagrams. This question turned out too hard for Johnny, so is counting on your help with it.

## Input

The first and only input line contains a single integer  $n$  ( $1 \leq n \leq 10^{12}$ ), the number of anagrams Johnny wants to obtain.

## Output

In the first and only output line you should print one integer: minimum possible length of a word consisting of small Latin letters (a-z) that has exactly  $n$  anagrams.

## Example

Input	Output
12	4

Word *baca* has 12 anagrams. Any shorter word has at most 6 anagrams, just like, e.g., *bac*.