

Overview: Discrete Logarithm

Description: Loki, who is a huge fan of file encryption, recently asked Professor Selvig, his reluctant minion, to help him protect the Tesseract. Using his mysteriously magical powers, Loki managed to convert the Tesseract into digital format (represented by a string). Unfortunately for poor Loki, Professor Selvig's encryption scheme is not foolproof!

While Selvig is unable to email you the Tesseract due to filesize limitations, he is confident in your ability to figure out how to decrypt it. His algorithm uses a password consisting of a single positive integer  $k$ . It takes as input a plaintext string consisting of uppercase letters. It then produces as output a ciphertext string consisting of uppercase letters. Selvig tells you that the algorithm works in the following way:

1. Convert each character into a number, such that 'A' = 0, 'B' = 1, 'C' = 2, ... 'Z' = 25.
2. For each number  $x$ , compute a new number  $y$ , such that  $y$  is the remainder of  $x^k$  divided by 26.
3. Convert each encrypted number back into a character, such that 0 = 'A', 1 = 'B', 2 = 'C', ... 25 = 'Z'.

For instance, suppose the input is the string "HELLO", and  $k = 5$ .

1. The characters in the string are converted to numbers to yield [7, 4, 11, 11, 14].
2. Take the 5th power of each number, then take the remainder when divided by 26. This yields the numbers [11, 10, 7, 7, 14].
3. Convert each number to a character to yield "LKHHO".

Selvig has provided you with an example input and the corresponding output. Your job is to determine the password  $k$  - if you do this, other people will be able to decrypt the Tesseract, and you save the world from complete annihilation!

Filename: adv21.{java, cpp, c, cc, py}

Input: The input consists of two lines, which are the input and output of the cryptographic algorithm respectively.

Output: Output the password  $k$ . If there are multiple possible passwords  $k$ , output the smallest possible  $k$ .

Assumptions: The line is at most 1,000 characters long.

Sample Input #1: HELLO  
LKHHO

Sample Output #1: 5

Sample Input #2: MJFWBUGEOBKUKIVPMAPBQGUYVXIIMYKAHMZQMLLZXOKEAKDKXZCORDFOSNZJAMFDVD  
MJVWBGUEOBKGKSFLMALBQUGCFXSSMCKATMZQMPPZXOKEAKDKXZYORDVOINZJAMVDFD

Sample Output #2: 7

