

## MaxPal

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 2048 MB

You are given a string  $s$  consisting only of lowercase letters.

You can do the following operation an **unlimited number of times (possibly zero)**: choose a substring of  $s$  and reverse it.

Your task is to maximise the number of substrings of  $s$  that are palindromes after doing the operations.

### Input

The first and only line of the input contains the string  $s$  ( $1 \leq |s| \leq 10^6$ ) consisting only of lowercase letters.

### Output

The output should contain the maximum number of substrings of  $s$  that are palindromes after doing an unlimited number of operations (possibly zero).

### Example

| standard input | standard output |
|----------------|-----------------|
| abcca          | 7               |

### Explanation

Starting from the string  $s = abcca$  (using 1-indexed positions, so  $s_1 = a$ ,  $s_2 = b$ ,  $s_3 = c$ ,  $s_4 = c$ ,  $s_5 = a$ ), by reversing the substring  $[3, 5]$  we can create the string **abacc** which has the following 7 palindromic substrings:  $[1, 1]$ ,  $[1, 3]$ ,  $[2, 2]$ ,  $[3, 3]$ ,  $[4, 4]$ ,  $[4, 5]$ ,  $[5, 5]$ .