E2 – Complexity

Define the *complexity* of a string to be the number of distinct letters in it. For example, the string `string` has complexity 6 and the string `letter` has complexity 4.

You like strings which have complexity either 1 or 2. Your friend has given you a string and you want to turn it into a string that you like. You have a magic eraser which will delete one letter from any string. Compute the minimum number of times you will need to use the eraser to turn the string into a string with complexity at most 2.

Input

The input consists of a single line that contains a single string of at most 100 lowercase ASCII letters (‘`a`’ – ‘`z`’).

Output

Print, on a single line, the minimum number of times you need to use the eraser.

Input and output samples

Input:	Output:
string	4

Input:	Output:
letter	2

Input:	Output:
aaaaaa	0

Input:	Output:
uncopyrightable	13

Input:	Output:
ambidextrously	12

Input:	Output:
assesses	1

Input:	Output:
assassins	2