G: Complexity

Time Limit: 1 second(s)

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.

Sample Input and Output

Sample Input 1	Output for Sample Input
`string`	`4`

Sample Input 2	Output for Sample Input
`letter`	`2`

Sample Input 3	Output for Sample Input
`aaaaaa`	`0`

Sample Input 4	Output for Sample Input
`uncopyrightable`	`13`

Sample Input 5	Output for Sample Input
`ambidextrously`	`12`

Sample Input 6	Output for Sample Input
`assesses`	`1`

Sample Input 7	Output for Sample Input
`assassins`	`2`