Simplicity

Problem ID: simplicity
CPU Time limit: 1 second
Memory limit: 1024 MB
Difficulty: 2.6

Source: 2015 Southeast USA Regionals Division 1

For a string of letters, define the *Simplicity* of the string to be the number of distinct letters in the string. For example, the string `string` has simplicity 6, and the string `letter` has simplicity 4.

You like strings which have simplicity 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 letters you must erase in order to turn the string into a string with simplicity at most 2.

Input

Each input will consist of a single test case. Note that your program may be run multiple times on different inputs. The input will consist of a line with a single string consisting of at least 1 and at most 100 lowercase letters ‘`a`’-‘`z`’.

Output

Output a single integer, indicating the minimum number letters you need to erase in order to give the string a simplicity of 1 or 2.

Sample Input 1

```
string
```

Sample Output 1

```
4
```

Sample Input 2

```
letter
```

Sample Output 2

```
2
```

Sample Input 3

```
aaaaaa
```

Sample Output 3

```
0
```

Sample Input 4

```
uncopyrightable
```

Sample Output 4

```
13
```

Sample Input 5

```
ambidextrously
```

Sample Output 5

```
12
```

Sample Input 6

```
assesses
```

Sample Output 6

```
1
```

Sample Input 7

```
assassins
```

Sample Output 7

```
2
```