This problem is an **interactive problem** new to the LeetCode platform.

We are given a word list of unique words, each word is 6 letters long, and one word in this list is chosen as **secret**.

You may call master.guess(word) to guess a word.  The guessed word should have type string and must be from the original list with 6 lowercase letters.

This function returns an integer type, representing the number of exact matches (value and position) of your guess to the **secret word**.  Also, if your guess is not in the given wordlist, it will return -1 instead.

For each test case, you have 10 guesses to guess the word. At the end of any number of calls, if you have made 10 or less calls to master.guess and at least one of these guesses was the **secret**, you pass the testcase.

Besides the example test case below, there will be 5 additional test cases, each with 100 words in the word list.  The letters of each word in those testcases were chosen independently at random from 'a' to 'z', such that every word in the given word lists is unique.

**Example 1:**

**Input:** secret = "acckzz", wordlist = ["acckzz","ccbazz","eiowzz","abcczz"]

**Explanation:**

master.guess("aaaaaa") returns -1, because "aaaaaa" is not in wordlist.

master.guess("acckzz") returns 6, because "acckzz" is secret and has all 6 matches.

master.guess("ccbazz") returns 3, because "ccbazz" has 3 matches.

master.guess("eiowzz") returns 2, because "eiowzz" has 2 matches.

master.guess("abcczz") returns 4, because "abcczz" has 4 matches.

We made 5 calls to master.guess and one of them was the secret, so we pass the test case.

**Note:**  Any solutions that attempt to circumvent the judge will result in disqualification.

这个问题是 LeetCode 平台新增的***交互式问题***。

我们给出了一个由一些独特的单词组成的单词列表，每个单词都是 6 个字母长，并且这个列表中的一个单词将被选作**秘密**。

你可以调用 master.guess(word) 来猜单词。你所猜的单词应当是存在于原列表并且由 6 个小写字母组成的类型字符串。

此函数将会返回一个整型数字，表示你的猜测与**秘密单词**的准确匹配（值和位置同时匹配）的数目。此外，如果你的猜测不在给定的单词列表中，它将返回 -1。

对于每个测试用例，你有 10 次机会来猜出这个单词。当所有调用都结束时，如果您对 master.guess 的调用不超过 10 次，并且至少有一次猜到**秘密**，那么您将通过该测试用例。

除了下面示例给出的测试用例外，还会有 5 个额外的测试用例，每个单词列表中将会有 100 个单词。这些测试用例中的每个单词的字母都是从 'a' 到 'z' 中随机选取的，并且保证给定单词列表中的每个单词都是唯一的。

**示例 1:**

**输入:** secret = "acckzz", wordlist = ["acckzz","ccbazz","eiowzz","abcczz"]

**解释:**

master.guess("aaaaaa") 返回 -1, 因为 "aaaaaa" 不在 wordlist 中.

master.guess("acckzz") 返回 6, 因为 "acckzz" 就是**秘密**，6个字母完全匹配。

master.guess("ccbazz") 返回 3, 因为 "ccbazz" 有 3 个匹配项。

master.guess("eiowzz") 返回 2, 因为 "eiowzz" 有 2 个匹配项。

master.guess("abcczz") 返回 4, 因为 "abcczz" 有 4 个匹配项。

我们调用了 5 次master.guess，其中一次猜到了**秘密**，所以我们通过了这个测试用例。

**提示：**任何试图绕过评判的解决方案都将导致比赛资格被取消。