824 Goat Latin

2018年5月5日   21:17

A sentence S is given, composed of words separated by spaces. Each word consists of lowercase and uppercase letters only.

We would like to convert the sentence to "*Goat Latin*" (a made-up language similar to Pig Latin.)

The rules of Goat Latin are as follows:

- If a word begins with a vowel (a, e, i, o, or u), append "ma" to the end of the word.
  For example, the word 'apple' becomes 'applema'.

- If a word begins with a consonant (i.e. not a vowel), remove the first letter and append it to the end, then add "ma".
  For example, the word "goat" becomes "oatgma".

- Add one letter 'a' to the end of each word per its word index in the sentence, starting with 1.
  For example, the first word gets "a" added to the end, the second word gets "aa" added to the end and so on.

Return the final sentence representing the conversion from S to Goat Latin.

Example 1:
Input: "I speak Goat Latin"
Output: "Imaa peaksmaaa oatGmaaaa atinLmaaaaa"
Example 2:
Input: "The quick brown fox jumped over the lazy dog"
Output: "heTmaa uickqmaaa rownbmaaaa oxfmaaaaa umpedjmaaaaaa overmaaaaaaa hetmaaaaaaaa azylmaaaaaaaaa ogdmaaaaaaaaaa"

Notes:

- S contains only uppercase, lowercase and spaces. Exactly one space between each word.
- 1 <= S.length <= 150.

来自 <https://leetcode.com/problems/goat-latin/description/>

Solution for Python3:

```
1   class Solution1:
2       def toGoatLatin(self, S):
3           """
4           :type S: str
5           :rtype: str
6           """
7           res = ''
8           for i, word in enumerate(S.split(' ')):
9               if word[0] in 'aeiouAEIOU':
10                  res += ' ' + word + 'ma' + 'a'*(i+1)
11              else:
12                  res += ' ' + word[1:] + word[0] + 'ma' + 'a'*(i+1)
13          return res[1:]
14
15  class Solution2:
16      def toGoatLatin(self, S):
17          """
18          :type S: str
19          :rtype: str
20          """
21          res = ''
22          for i, word in enumerate(S.split(' '), 1):
23              if word[0] not in 'aeiouAEIOU':
24                  word = word[1:] + word[0]
```

```
25          res += ' ' + word + 'ma' + 'a' * i
26       return res[1:]
```

Solution for C++:

```cpp
1  class Solution {
2  public:
3      string toGoatLatin(string S) {
4          string res = "", vowel = "aeiou", word;
5          istringstream ssm(S);
6          int i = 1;
7          while (ssm >> word) {
8              if (vowel.find(tolower(word[0])) != string::npos)
9                  res += " " + word + "ma" + string(i, 'a');
10             else
11                 res += " " + word.substr(1,word.length() - 1) + word[0] + "ma" +
12 string(i, 'a');
13             i++;
14         }
15         return res.substr(1, res.length()-1);
16     }
   };
```