520 Detect Capital

2018年4月13日 17:37

Given a word, you need to judge whether the usage of capitals in it is right or not.

We define the usage of capitals in a word to be right when one of the following cases holds:

1. All letters in this word are capitals, like "USA".
2. All letters in this word are not capitals, like "leetcode".
3. Only the first letter in this word is capital if it has more than one letter, like "Google".

Otherwise, we define that this word doesn't use capitals in a right way.

Example 1:

Input: "USA"

Output: True

Example 2:

Input: "FlaG"

Output: False

Note: The input will be a non-empty word consisting of uppercase and lowercase latin letters.

来自 <https://leetcode.com/problems/detect-capital/description/>

给定一个单词，你需要判断单词的大写使用是否正确。

我们定义，在以下情况时，单词的大写用法是正确的：

1. 全部字母都是大写，比如"USA"。
2. 单词中所有字母都不是大写，比如"leetcode"。
3. 如果单词不只含有一个字母，只有首字母大写，比如 "Google"。

否则，我们定义这个单词没有正确使用大写字母。

注意: 输入是由大写和小写拉丁字母组成的非空单词。

Solution for Python3：

```python
class Solution1:
    def detectCapitalUse(self, word):
        """
        :type word: str
        :rtype: bool
        """
        if word.upper() == word or word.lower() == word:
            return True
        if ord(word[0]) >= ord('a'):
            return False
        print(word)
        return not any([ord(w) < ord('a') for w in word[1:]])
        # return all([ord(w) >= ord('a') for w in word[1:]])

class Solution2:
    def detectCapitalUse(self, word):
        """
        :type word: str
        :rtype: bool
        """
        return word[1:] == word[1:].lower() or word == word.upper()

class Solution3:
    def detectCapitalUse(self, word):
        """
        :type word: str
        :rtype: bool
        """
        return word in [word.upper(), word.lower(), word.capitalize()]

class Solution4:
    def detectCapitalUse(self, word):
```

```python
33            """
34            :type word: str
35            :rtype: bool
36            """
37            return word.isupper() or word.islower() or word.istitle()
```

Solution for C++:

```cpp
1    class Solution1 {
2    public:
3        bool detectCapitalUse(string word) {
4            int cnt = 0;  //大写字母个数
5            for (char c : word) {
6                if ('Z' - c >= 0)
7                    cnt++;
8            }
9            return cnt == 0 || cnt == word.length() || (cnt == 1 && 'Z' - word[0] >= 0);
10       }
11   };
12
13   class Solution2 {
14   public:
15       bool detectCapitalUse(string word) {
16           return regex_match(word, regex("[A-Z]+|[a-z]+|[A-Z][a-z]+"));
17       }
18   };
19
20   class Solution3 {
21   public:
22       bool detectCapitalUse(string word) {
23           return regex_match(word, regex("[A-Z]*|[A-Z]?[a-z]*"));
24       }
25   };
```