5/7 Leetcode

题目：检测大写字母

我们定义，在以下情况时，单词的大写用法是正确的：

全部字母都是大写，比如 "USA" 。

单词中所有字母都不是大写，比如 "leetcode" 。

如果单词不只含有一个字母，只有首字母大写， 比如 "Google" 。

给你一个字符串 word 。如果大写用法正确，返回 true ；否则，返回 false 。

示例 1：

输入：word = "USA"

输出：true

示例 2：

输入：word = "FlaG"

输出：false

来源：力扣（LeetCode）

链接：<https://leetcode-cn.com/problems/detect-capital>

思路：

这道题的要求是判断大写字母用法是否正确，刚读完题以为是只要判断是否是大写字母就好了，所以第一次写了一个错位的程序，简单的判断有没有小写字母，但是提交题解时发现了错误，一开始还不理解哪里错了，然后我又重新仔细审题，才知道原来是要判断大写字母用法是否正确，刚理解题目以后也是有了清晰的思路，我的思路如下：

1. 如果第一个字符是大写则分两种情况：
2. 后面都是大写（如：USA）则正确，返回true；
3. 后面都是小写（如：Leetcode）则正确，返回true；

2．如果第一个字符是小写：

后面的字符也都要是小写（如：apple,banana）则正确，返回true。

思路清晰了以后就可以很快的写出框架，先判断第一个字符，然后根据情况判断第二个字符以及后面的，如果遇到违背上述规则的，直接返回false，如果没有违背则一直遍历下去直到遍历结束，返回true，这样这道题就很快的写完了，在编程过程中还算比较顺利，没有遇到意料之外的错误，也可以说几乎没遇到什么困难，或许这就是提前想好框架的好处吧，编程速度快了许多，遇到的问题也少了许多！

总结：

在写这道题目的时候，第一次读错题目了，让我知道了审题的重要性。第二次读完题目，先自己在注释上写好了思路，然后按着思路一点一点的编下去，就很快的解决问题了，通过这道题，我发现了在编程开始前先想好思路的重要性，可以大幅提高编程的速度，也可以绕开很多奇怪的问题，所以这道题对我来说收获最大的是养成了编程前先搭建框架的习惯！