# 第一个只出现一次的字符

## 题目

[牛客网](https://www.nowcoder.com/practice/1c82e8cf713b4bbeb2a5b31cf5b0417c?tpId=13&tqId=11187&tPage=2&rp=2&ru=%2Fta%2Fcoding-interviews&qru=%2Fta%2Fcoding-interviews%2Fquestion-ranking)

在一个字符串(0<=字符串长度<=10000，全部由字母组成)中找到第一个只出现一次的字符,并返回它的位置, 如果没有则返回 -1（需要区分大小写）.

## 解题思路

1. 通过 LinkedHashMap 记录数组顺序，然后计算字符出现的次数
2. 遍历找到第一个只出现 1次 的字符

public int FirstNotRepeatingChar(String str) {
 LinkedHashMap<Character, Integer> data = new LinkedHashMap<>();
 char[] chars = str.toCharArray();
 for (char c : chars) {
 Integer count = data.getOrDefault(c, 0);
 data.put(c, count + 1);
 }

 Character res = null;
 for (Character c : data.keySet()) {
 if (data.get(c) == 1) {
 res = c;
 break;
 }
 }

 if (res == null) {
 return -1;
 }

 for (int i = 0; i < chars.length; i++) {
 if (chars[i] == res) {
 return i;
 }
 }

 return -1;
}