**难度: 简单**

给你两个字符串，请你从这两个字符串中找出最长的特殊序列。

「最长特殊序列」定义如下：该序列为某字符串独有的最长子序列（即不能是其他字符串的子序列）。

子序列 可以通过删去字符串中的某些字符实现，但不能改变剩余字符的相对顺序。空序列为所有字符串的子序列，任何字符串为其自身的子序列。

输入为两个字符串，输出最长特殊序列的长度。如果不存在，则返回 -1。

示例 1：

输入: "aba", "cdc"

输出: 3

解释: 最长特殊序列可为 "aba" (或 "cdc")，两者均为自身的子序列且不是对方的子序列。

示例 2：

输入：a = "aaa", b = "bbb"

输出：3

示例 3：

输入：a = "aaa", b = "aaa"

输出：-1

提示：

两个字符串长度均处于区间 [1 - 100] 。

字符串中的字符仅含有 'a'~'z' 。

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/longest-uncommon-subsequence-i

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

思路:

1). 先判断a是否与b相同,相同返回-1

2). 如果ab不相同,判断ab哪个长度更长,直接返回

|  |
| --- |
| public int findLUSlength(String a, String b) {  if (a.equals(b)){  return -1;  }else {  return Math.max(a.length(), b.length());  }  } |