[2399. 检查相同字母间的距离 - 力扣（Leetcode）](https://leetcode.cn/problems/check-distances-between-same-letters/description/)

给你一个下标从 **0** 开始的字符串 s ，该字符串仅由小写英文字母组成，s 中的每个字母都 **恰好** 出现 **两次** 。另给你一个下标从 **0** 开始、长度为 26 的的整数数组 distance 。

字母表中的每个字母按从 0 到 25 依次编号（即，'a' -> 0, 'b' -> 1, 'c' -> 2, ... , 'z' -> 25）。

在一个 **匀整** 字符串中，第 i 个字母的两次出现之间的字母数量是 distance[i] 。如果第 i 个字母没有在 s 中出现，那么 distance[i] 可以 **忽略** 。

如果 s 是一个 **匀整** 字符串，返回 true ；否则，返回 false 。

**示例 1：**

**输入：**s = "abaccb", distance = [1,3,0,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

**输出：**true

**解释：**

- 'a' 在下标 0 和下标 2 处出现，所以满足 distance[0] = 1 。

- 'b' 在下标 1 和下标 5 处出现，所以满足 distance[1] = 3 。

- 'c' 在下标 3 和下标 4 处出现，所以满足 distance[2] = 0 。

注意 distance[3] = 5 ，但是由于 'd' 没有在 s 中出现，可以忽略。

因为 s 是一个匀整字符串，返回 true 。

**示例 2：**

**输入：**s = "aa", distance = [1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

**输出：**false

**解释：**

- 'a' 在下标 0 和 1 处出现，所以两次出现之间的字母数量为 0 。

但是 distance[0] = 1 ，s 不是一个匀整字符串。

**提示：**

* 2 <= s.length <= 52
* s 仅由小写英文字母组成
* s 中的每个字母恰好出现两次
* distance.length == 26
* 0 <= distance[i] <= 50