

2716. Minimize String Length

Description

Given a **0-indexed** string `s`, repeatedly perform the following operation **any** number of times:

- Choose an index `i` in the string, and let `c` be the character in position `i`. **Delete** the **closest occurrence** of `c` to the **left** of `i` (if any) and the **closest occurrence** of `c` to the **right** of `i` (if any).

Your task is to **minimize** the length of `s` by performing the above operation any number of times.

Return *an integer denoting the length of the **minimized** string*.

Example 1:

Input: `s = "aaabc"`
Output: `3`
Explanation: In this example, `s` is "aaabc". We can start by selecting the character 'a' at index 1. We then remove the closest 'a' to the left of index 1, which is at index 0, and the closest 'a' to the right of index 1, which is at index 2. After this operation, the string becomes "abc". Any further operation we perform on the string will leave it unchanged. Therefore, the length of the minimized string is 3.

Example 2:

Input: `s = "cbdd"`
Output: `3`
Explanation: For this we can start with character 'b' at index 1. There is no occurrence of 'b' to the left of index 1, but there is one to the right at index 2, so we delete the 'b' at index 2. The string becomes "cbd" and further operations will leave it unchanged. Hence, the minimized length is 3.

Example 3:

Input: `s = "dddaaa"`
Output: `2`
Explanation: For this, we can start with the character 'd' at index 1. The closest occurrence of a 'd' to its left is at index 0, and the closest occurrence of a 'd' to its right is at index 2. We delete both index 0 and 2, so the string becomes "daaa". In the new string, we can select the character 'a' at index 2. The closest occurrence of an 'a' to its left is at index 1, and the closest occurrence of an 'a' to its right is at index 3. We delete both of them, and the string becomes "da". We cannot minimize this further, so the minimized length is 2.

Constraints:

- `1 <= s.length <= 100`
- `s` contains only lowercase English letters

