[LeetCode](https://leetcode.com/problems/minimum-length-of-string-after-deleting-similar-ends/submissions/1194091564/?envType=daily-question&envId=2024-03-05)

<https://github.com/AdityaKonda6/-50DaysOfCoding>

<https://leetcode.com/problems/minimum-length-of-string-after-deleting-similar-ends/submissions/1194091564/?envType=daily-question&envId=2024-03-05>

<https://www.linkedin.com/in/aditya-adi-konda/>

 Day 15 of [#50dayscodingchallenge](https://www.linkedin.com/feed/hashtag/?keywords=50dayscodingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633):  
[#leetcode](https://www.linkedin.com/feed/hashtag/?keywords=leetcode&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodechallenge](https://www.linkedin.com/feed/hashtag/?keywords=leetcodechallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodestreak](https://www.linkedin.com/feed/hashtag/?keywords=leetcodestreak&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode2024](https://www.linkedin.com/feed/hashtag/?keywords=leetcode2024&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode50day](https://www.linkedin.com/feed/hashtag/?keywords=leetcode50day&highlightedUpdateUrns=urn:li:activity:7166316239483461633)  
   
Just kicked off my coding journey with a fascinating problem - "Successfully solved LeetCode Problem “1750. Minimum Length Of String After Deleting Similar Ends” !”  
   
✨ Task: Given a string s consisting only of characters 'a', 'b', and 'c'. You are asked to apply the following algorithm on the string any number of times:

* Pick a non-empty prefix from the string s where all the characters in the prefix are equal.
* Pick a non-empty suffix from the string s where all the characters in this suffix are equal.
* The prefix and the suffix should not intersect at any index.
* The characters from the prefix and suffix must be the same.
* Delete both the prefix and the suffix.
* Return the minimum length of s after performing the above operation any number of times (possibly zero times).

Examples:

Example 1:

Input: s = "ca"

Output: 2

Explanation: You can't remove any characters, so the string stays as is.

Example 2:

Input: s = "cabaabac"

Output: 0

Explanation: An optimal sequence of operations is:

- Take prefix = "c" and suffix = "c" and remove them, s = "abaaba".

- Take prefix = "a" and suffix = "a" and remove them, s = "baab".

- Take prefix = "b" and suffix = "b" and remove them, s = "aa".

- Take prefix = "a" and suffix = "a" and remove them, s = "".

Example 3:

Input: s = "aabccabba"

Output: 3

Explanation: An optimal sequence of operations is:

- Take prefix = "aa" and suffix = "a" and remove them, s = "bccabb".

- Take prefix = "b" and suffix = "bb" and remove them, s = "cca".

Let's Connect:

If you find this problem intriguing or have insights to share, let's connect! I'm passionate about problem-solving, algorithmic thinking, and collaborative learning. Feel free to comment or reach out for engaging discussions and knowledge exchange.Unravel the mystery using your coding skills!

[#CodingChallenge](https://www.linkedin.com/feed/hashtag/?keywords=codingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#Algorithm](https://www.linkedin.com/feed/hashtag/?keywords=algorithm&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#LinkedInPost](https://www.linkedin.com/feed/hashtag/?keywords=linkedinpost&highlightedUpdateUrns=urn:li:activity:7166316239483461633) #Algorithm #Optimization #DataStructures #CodingChallenge  
  
Excited about the progress and challenges ahead!  
   
Make Sure You Follow My GitHub For Solutions: <https://github.com/AdityaKonda6/-50DaysOfCoding>  
  
  
Happy coding!

**Solution:-**

class Solution {

  public int minimumLength(String s) {

    int i = 0;

    int j = s.length() - 1;

    while (i < j && s.charAt(i) == s.charAt(j)) {

      final char c = s.charAt(i);

      while (i <= j && s.charAt(i) == c)

        ++i;

      while (i <= j && s.charAt(j) == c)

        --j;

    }

    return j - i + 1;

  }

}

