Find the Length of String

The given code aims to calculate the length of a C-style string (a character array terminated by a null character '\0'). It provides two approaches to calculate the length: one using a custom function and the other using the **strlen** function from the **<cstring>** library.

Example: Suppose the user enters the name "John Doe" as input. The code will calculate the length of the name using both approaches and display the results as output.

Explanation: The code has two functions: **getLength** and **getLengthStrLen**, which both take a C-style string as input and return its length.

Approach 1: Function to calculate the length of a C-style string

- This approach iterates through each character of the string until it encounters the null character ('\0').
- It uses a **for** loop to increment a count variable for each character encountered.
- The loop terminates when the null character is found, indicating the end of the string.
- The count is returned as the length of the string.

Time Complexity: O(n), where n is the length of the string. In the worst case, it iterates through the entire string once.

Space Complexity: O(1). It uses a single variable (count) to store the length.

Approach 2: Function to calculate the length of a C-style string using strlen function

- This approach utilizes the **strlen** function from the **<cstring>** library.
- **strlen** is a standard C library function that returns the length of a null-terminated string.
- It calculates the length by scanning the string until it reaches the null character.
- The length returned by **strlen** is directly used as the result.

Time Complexity: O(n), where n is the length of the string. The strlen function scans the entire string once.

Space Complexity: O(1). It does not require any additional space.