CPSC 335 – Project 2

Algorithm 2: String Run Encoding

Members:

- Michael Bui / buimichael@csu.fullerton.edu
- Natalia Garcia / natgarcia@csu.fullerton.edu

Project Report:

For the string run encoding problem, we decided that for a string input, we iterate through every character in the string with a for-loop and check for conditions with if-else statements. There is a variable that keeps count of the same characters and a current variable that is initialized to the first character of the string, which is then compared to the next character. If there is a repeating character, the count variable increases by one, after if count is greater than one, the count and the character are added into a new string. If the current character and the next character are not the same, they are added to the new string. Once the function `StrRunEncode()` is finished iterating through the input string, the new string is returned.

Mathematical Analysis:

The program starts by making sure the input is not an empty string and then checks if the string is all in lowercase, which if it's the case, it continues to the for-loop. The best case scenario for the function StrRunEncode() would be **O(n)**, if the string is iterated through once. For instance, given the sample input, the string "ddd" is a good example for when the best case scenario is used, as the string iterates and is then checked for the first condition in the for-loop. Constant operations are performed, including the if-statement outside the for-loop, producing a linear time complexity. On the other hand, given a string with no consecutive characters that are similar, the for-loop is executed ***n*** times, additionally, 'new_string += curr' is being executed ***n*** times as well. This makes the time complexity of the worst case scenario to be of quadratic complexity or **O(n^2)**.