Jesse Narkmanee

805190133

CS 31 Project 4 Report

**Encountered Problems**

The main problem I had with this project was mainly my initial design algorithm of the firstNonRepeatedCharacter()and the highestOccurredCharacter(). Their problems are both very similar which is why my idea on how to initially solve both problems were also similarly flawed. At first, my plan was to store the character element (with the most occurrence or no-repeat occurrence) and compare it with the rest of the other characters through a nested for loop, switching the element stored as we find a new character that beat the previous one. However, this design was flawed in that it stopped counting on repeats when characters before were stored but were switched out, losing its count forever. To fix this, I decided to use a second array that I made myself to reflect different character counts. By assigning these counts through for loops after the iteration, I was able to analyze each and every single element and see their occurrence total, then use that to find the correct index and char to return.

**Test Cases**

Test Cases I ran included things that I felt would usually be encountered and then things that were probably rarer. The one usually encountered were the test cases that were listed on the project specs such as if n is 0 does it return to a default value (such as ‘\0’) and when the arrays given would turn out to be true for the people and folk array for the given arrays on the spec. I added some additional test cases for these same arrays as well but this time made a smaller n parameter value to not include the elements that would make the return originally true to see if it indeed would produce a false result now. All my tests were successful.

In trying to find more unusual cases, I did assert tests for if two empty strings are reversible and received an out of bounds error, which was a good catch and so had to fix that, which was not a big problem. I simply created an if statement to automatically return true if met with two empty swings. Also, I found isOnlyDigits() and the two functions which returned the highest occurrence and non-repeating character to not return the intended values for empty strings as well. However, yet again the fix was simple as I implemented an if statement check to solve this issue.

Some additional and final test cases that I would like to mention is that I also made sure that two single same chars would be determined as reversible and that arrays with an additional empty string as one of its additional elements in an array would work. All worked successfully and I am pretty satisfied with all the assert statements I created.