**CMSC203 Assignment 3 Implementation (Documentation)**

Class: CMSC203 CRN 24307

 Program: Assignment 3

Instructor: Prof Tsai

 Summary of Description: Encrypt and decrypt Bellasso and Caesar cipher

 Due Date: 10/10/2021

 Integrity Pledge: I pledge that I have completed the programming assignment independently.

 I have not copied the code from a student or any source.

**Part1: Pseudo Code:** Here is a pseudo code for Assignment 2 program:

**stringInBounds(String)**

**for each character of string**

**if character is out of bounds, return false**

**return true**

**encryptCaesar(String, int)**

**for each character of string**

**character = (character+int) % ascii\_limit**

**decryptCaesar(String, int)**

**for each character of string**

**character = (character-int) % ascii\_limit**

**encryptBellaso(String, String)**

**for each character of string**

**character = (character + string2character) % ascii\_limit**

**decryptBellaso(String, String)**

**for each character of string**

**character = (character - string2character) % ascii\_limit**

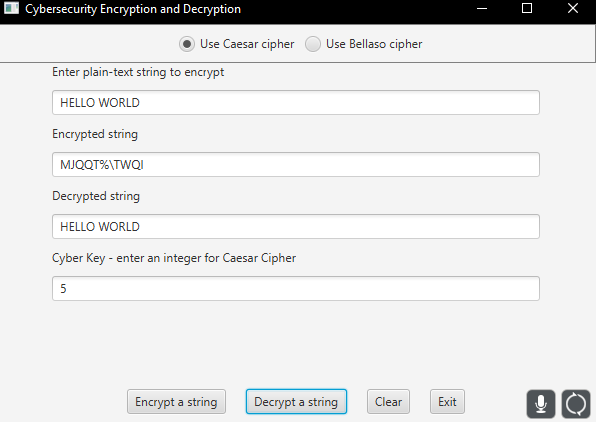
**Part2: Comprehensive Test Plan**

A good test plan should be comprehensive. This means you should have a few test cases that test when the input is in and out of range, division by 0, incorrect Data type, etc. (Provide valid and invalid input)

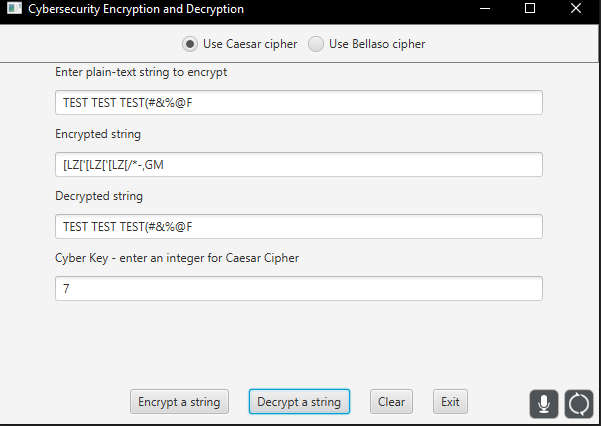
| Cases | Input | Expected Output | Actual Output | Did Test Pass? |
| --- | --- | --- | --- | --- |
| Case 1 | HELLO WORLD  5 | MJQQT%\TWQI | MJQQT%\TWQI | Yes |
| Case 2 | TEST TEST TEST(#&%@F  7 | [LZ['[LZ['[LZ[/\*-,GM | [LZ['[LZ['[LZ[/\*-,GM | Yes |
| Case 3 | THIS IS ANOTHER TEST  TESTSTR | (M\'3]%4F!#'\W&%'Y&( | (M\'3]%4F!#'\W&%'Y&( | Yes |
| Case 4 | TESTING AGAIN (#%@  HELLO | \J\_ XVL,MVINZ,7+\*L | \J\_ XVL,MVINZ,7+\*L | Yes |
| Case 5 | {}{}{}{}{$#TGGV R  342 | OUT OF BOUNDS | OUT OF BOUNDS | Yes |

**Part3: Screenshots related to the Test Plan:**

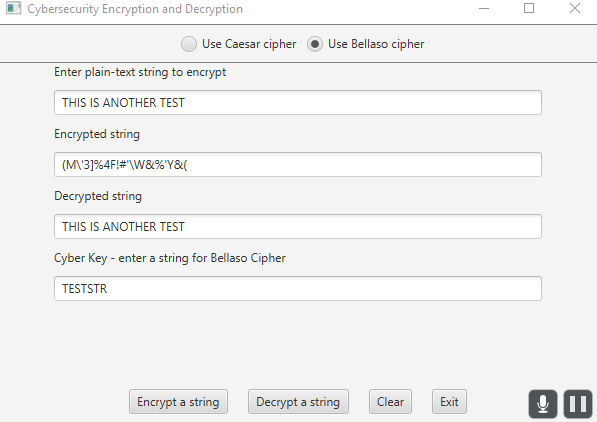
**Case 1**

****

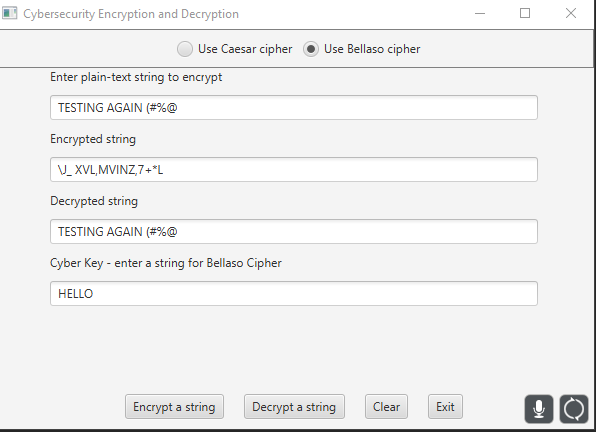
**Case 2**

****

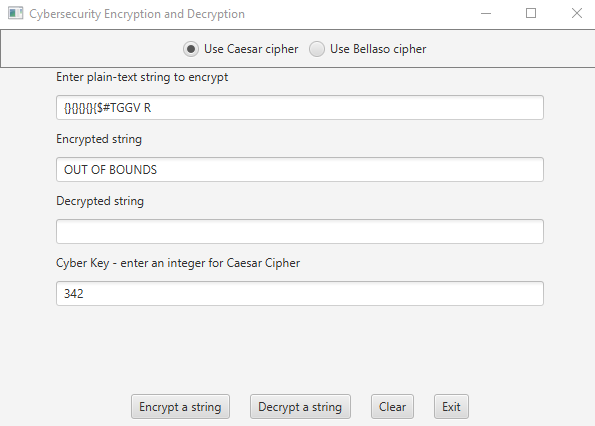
**Case 3**

****

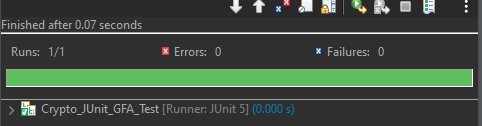
**Case 4**

****

**Case 5**

****

**GFA Test**

****

**Lessons Learned** <Provide answers to the questions listed above>**:**

Write about your Learning Experience, highlighting your lessons learned and learning experience from working on this project.

What have you learned?

**I learned how to effectively use the modulus operator to wrap around overflowed characters and prevent errors.**

What did you struggle with?

**I struggled with creating effective algorithms that would work with any key that would be used without error.**

What would you do differently on your next project?

**One thing I would do differently is I would do more research on the task at hand, such as ciphers for this assignment.**

What parts of this assignment were you successful with, and what parts (if any) were you not successful with?

**I was successful with all parts of the assignment.**

Provide any additional resources/links/videos you used to while working on this assignment/project.

**Check List:** <Provide answers to the column Y/N or N/A >**:**

| **#** |  | **Y/N** | **Comments** |
| --- | --- | --- | --- |
|  | **Assignment files:** |  |  |
|  | * FirstInitialLastName\_ Assignment#\_Moss.zip | **Yes** |  |
|  | * FirstInitialLastName\_Assignment#.docx/.pdf | **Yes** |  |
|  | * Source java files | **Yes** |  |
|  | **Program compiles** | **Yes** |  |
|  | **Program runs with desired outputs related to a Test Plan** | **Yes** |  |
|  | **Documentation file:** |  |  |
|  | * Comprehensive Test Plan | **Yes** |  |
|  | * Screenshots related to the Test Plan | **Yes** |  |
|  | * Screenshots of your GitHub account with submitted Assignment# (if required) | **Yes** |  |
|  | * UML Diagram (if required) | **N/A** |  |
|  | * Algorithms/Pseudocode (if required) | **Yes** |  |
|  | * Flowchart (if required) | **N/A** |  |
|  | * Lessons Learned | **Yes** |  |
|  | * Checklist is completed and included in the Documentation | **Yes** |  |