

CMSC203 Assignment 3 Implementation (Documentation)

Class: CMSC203 CRN 40438

Program: Assignment #3

Instructor: Dr. Grinberg

Summary of Description: **The provided code encrypts and decrypts text using the Caesar Cipher and Bellaso Cipher techniques. The Caesar encryption shifts characters by a specified key, while Bellaso encryption uses a repeating key string to offset characters. Decryption methods reverse the encryption process. The class ensures the input strings are within the allowable ASCII bounds before performing the operations.**

Due Date: 07/31/2023

Integrity Pledge: I pledge that I have completed the programming assignment independently. I have not copied the code from a student or any source.

Even Estifanos

Part1: Pseudo Code: Here is a pseudo-code for Assignment 3 program:

Start the CryptoManager Class

isStringInBounds method:

- Given - a plain text to be encrypted
- Check if the input string is within the allowable bounds of ASCII characters.
- Loop through each character in the string and verify if its ASCII code is within the specified range.
- Return true if all characters are within bounds, otherwise return false.

caesarEncryption method:

- Given - a plain text to be encrypted, an integer key to offset the plain text
- Encrypt the input string using the Caesar Cipher technique.
- Take an integer key as an offset for each character in the string.
- Shift each character by the value of the key in the ASCII character set.
- Wrap around the character range if needed to stay within the allowable bounds.
- Return the encrypted string.

bellasoEncryption method:

- Given - a plain text to be encrypted, and another plain text to offset the plain text
- Encrypt the input string using the Bellaso Cipher technique.
- Take a repeating key string, bellasoStr, to offset each character in the plainText.
- The bellasoStr is repeated as needed to cover the entire plainText.
- Offset each character in the plainText by the ASCII value of the corresponding character in bellasoStr.
- Wrap around the character range if needed to stay within the allowable bounds.
- Return the encrypted string.

caesarDecryption method:

- Given - an encrypted text, an integer key to reverse the encrypted text
- Decrypt the input encrypted string that was encrypted using the Caesar Cipher technique.
- Use the same integer key that was used for encryption to reverse the process.

- Shift each character backward by the value of the key in the ASCII character set.
- Wrap around the character range if needed to stay within the allowable bounds.
- Return the decrypted string.

bellasoDecryption method:

- Given - an encrypted text, and another text to reverse the encrypted text
- Decrypt the input encrypted string that was encrypted using the Bellaso Cipher technique.
- Use the same repeating key string, bellasoStr, that was used for encryption to reverse the process.
- Offset each character in the encryptedText backward by the ASCII value of the corresponding character in bellasoStr.
- Wrap around the character range if needed to stay within the allowable bounds.
- Return the decrypted string.

End the CryptoManager class

Part2: UML Class Diagram(see Handouts)

CryptoManager	
- LOWER_RANGE : char - UPPER_RANGE : char - RANGE : int	
+ isStringInBounds (String plainText): Boolean + caesarEncryption(String plainText, int key): String + bellasoEncryption (String plainText, String bellasoStr): String + caesarDecryption (String encryptedText, int key) : String + bellasoDecryption(String encryptedText, String bellasoStr) :String	

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?
1	Type : Caesar Cipher String : JAVAISCOOL key : 50 *to be encrypted*	<3H3;E5AA>	<3H3;E5AA>	YES
2	Type : Caesar Cipher String : APORLA15 key : 15	2A@C=2"&	2A@C=2"&	YES

	to be decrypted			
3	Type : Bellaso Cipher String : POATAN key : ALEXPER *to be encrypted*	Q[F,QS	Q[F,QS	YES
4	Type : Bellaso Cipher String : SHAMAQWR{} Key : UFC291 *to be encrypted*	The selected string is not in bounds, Try again.	The selected string is not in bounds, Try again.	YES

Part4: Screenshots related to the Test Plan:

Case 1

Cybersecurity Encryption and Decryption

☒ Use Caesar cipher ☐ Use Bellaso cipher

Enter plain-text string to encrypt

JAVASCOOL

Encrypted string

<3H3;E5AA>

Decrypted string

Cyber Key - enter an integer for Caesar Cipher

50

Encrypt a string Decrypt a string Clear Exit

Case 2

Cybersecurity Encryption and Decryption

☒ Use Caesar cipher ☐ Use Bellaso cipher

Enter plain-text string to encrypt

Encrypted string

APORLA15

Decrypted string

2A@C=2"&

Cyber Key - enter an integer for Caesar Cipher

15

Encrypt a string Decrypt a string Clear Exit

Case 3

Cybersecurity Encryption and Decryption

☐ Use Caesar cipher ☒ Use Bellaso cipher

Enter plain-text string to encrypt

POATAN

Encrypted string

Q[F,QS

Decrypted string

Cyber Key - enter a string for Bellaso Cipher

ALEXPER

Encrypt a string Decrypt a string Clear Exit

Case 4

Cybersecurity Encryption and Decryption

☐ Use Caesar cipher ☒ Use Bellaso cipher

Enter plain-text string to encrypt

SHAMAQWR}

Encrypted string

The selected string is not in bounds, Try again.

Decrypted string

Cyber Key - enter a string for Bellaso Cipher

UFC291

Encrypt a string Decrypt a string Clear Exit

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?

- ASCII - every character has its own ASCII Code and there is a table of them, which then developers can manipulate to produce like this programs
- ENCRYPTION/DECRYPTION - how to hide text to deceive enemies , two methods : Cipher - offsetting the character by a certain number, and Bellaso, my favorite and more protective - offsetting characters by a corresponding character of another text
- ALSO how to loop with Char values

What did you struggle with?

- Wrapping the character - In bellaso, if the length of the second text is smaller than the first one, you have to loop again through ex- HELLOHELLOHELLO, I was struggling to implementing this process

What would you do differently on your next project?

- Better time management - I was rushing and got it done with in two days nonstop working

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

- Coding was okay, but design was exhausting as usual. I feel like the design part should be voluntary, cause I find pseudocode and UML, and flowcharts to be a time waster instead of a helping aid

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