IT 166 Fall 2019

**The Cryptography Project**

**Project Description**

In this project, you will be asked to implement two ciphers, the Caesar’s cipher and the Vigenere cipher. You will implement both the encryption and decryption methods for the two ciphers.

There is some code that has been written to help you get started. This project can be a group project if you decide to work in groups. However, the maximum number of members is 3 per group.

The grade of this project will be used as the grade of the second midterm exam.

**Steps to complete the project**

**Step 1** Download the file:

Download the “cryptography.py” and you will find there is some existing code in the file. DO NOT MODIFY them.

**Step 2** Understand the existing code:

* plain\_text is a piece of string that is meant to be encrypted.
* look\_up\_list\_gen is a function that generates a look-up list. For example, if the given text is “Alice talks to Bob.”, it will generate the following list: ['l', 'i', 't', 'e', 'k', 'c', 'a', 'o', ' ', '.', 'b', 'A', 's', 'B']
* assert(key > 0 and type(key) == int) makes sure the key of the Caesar’s cipher must be a positive integer.
* assert(type(key) == str) makes sure the key of the Vigenere cipher must be a string.
* assert(len(self.key) <= len(plain\_text)) makes sure the key of the Vigenere cipher must be shorter or at least be the same length of the plain text.
* The statements under if \_\_name\_\_ == "\_\_main\_\_": are the function calls that test both ciphers.

**Step 3** About the Caesar’s cipher:

The key used by the Caesar’s cipher controls the number of down shifts inside the look-up list. For example, using the look-up list above with the key’s value being 3, the first letter, “A”, in the plain text, “Alice talks to Bob.”, will shift down 3 positions inside the look-up list, such that “A” will be replaced by “l”, “l” will be replaced by “e”, “i” will be replaced by “k” and so on. The encrypted information known as the cipher text looks like: “lek aAc.eoiAcbAtbBs”

**Step 4** Implement the encrypt and decrypt methods for the Caesar cipher. The encrypt method takes the plain text as input and generates its encrypted counterpart named cipher\_text. The decrypt method decrypts the cipher\_text and returns the plain text.

**Step 5** About the Vigenere cipher:

The key used by the Vigenere cipher is a string. And more importantly, it must be a substring extracted from the plain text. Suppose the key is “Bob”, since the length of “Bob” is less than the plain text, you will need to pad the key to the same length of the plain text before the encryption:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | l | i | c | e |  | t | a | l | k | s |  | t | o |  | B | o | b | . |
| B | o | b | B | o | b | B | o | b | B | o | b | B | o | b | B | o | b | B |

The first letter, “A”, is encrypted by “B”. This means “B” has the index of 13 in the look-up list, such that “A” will then shift down 13 positions and becomes “b”. Another example is the second letter, “l”, is encrypted by “o”, which has an index of 7 based on the look-up list. Thus, “l” will be shifted down 7 positions and becomes “o” after the encryption.

**Step 6** Implement the encrypt and decrypt methods for the Vigenere cipher. The encrypt method takes the plain text as input and generates its encrypted counterpart named cipher\_text. The decrypt method decrypts the cipher\_text and returns the plain text.

**Step 7** Submit your solution on ReggieNet no later than November 6 11:55 PM. If you worked in a group, only one submission from the group is required with a proper documentation of all group members’ names and ULID.