CSE 436 Computer and Networks Security

Assignment I – Classical Encryption Techniques

This assignment is a practice on the five famous and traditional encryption techniques: Caesar, Play Fair, Hill, Vigenere, and Vernam. Read the following problems carefully and don't forget to check the end of this file where you can find instructions on how to submit the assignment.

1. Caesar Cipher

In this problem, you are required to implement the Caesar cipher algorithm. Build a function that takes string input of the plaintext and integer for the key, and outputs a string containing the ciphertext. *Use the following keys to test your function: 3, 6, 12*

2. Play Fair Cipher

Implement the Play Fair cipher algorithm including the creation of the 5x5 Play Fair matrix that you will use to encrypt the input. Use the following guidelines:

- If a pair is a repeated letter, insert 'X' as filler between the two characters.
- If there is a single trialing letter, attach 'X' to the end to complete the two-letter block.
- If both letters fall in the same row, replace each with letter to right (wrapping back to start from end.)
- If both letters fall in the same column, replace each with the letter below it (wrapping to top from bottom.)
- Otherwise, each letter is replaced by the letter in the same row and in the column of the other letter of the pair.

You are required to build a function that takes string input of the plaintext and string input of the key, and outputs string output for the ciphertext. Build any other necessary functions to help you get the final output (i.e. function to create the Play Fair matrix). *Use the following keys: rats, archangel*

3. Hill Cipher

Implement the Hill cipher algorithm. Create a function that takes the plaintext as string and the key matrix as an array of integers, and outputs the ciphertext as string. You will have to convert the any text to its ASCII representation to multiply with the key matrix. Use the following formula:

$$\begin{pmatrix} C1 \\ C2 \end{pmatrix} = \begin{pmatrix} K1 & K2 \\ K3 & K4 \end{pmatrix} \begin{pmatrix} P1 \\ P2 \end{pmatrix}$$

Use the following keys to test your algorithm:

$$\begin{pmatrix} 5 & 17 \\ 8 & 3 \end{pmatrix} \quad , \quad \begin{pmatrix} 2 & 4 & 12 \\ 9 & 1 & 6 \\ 7 & 5 & 3 \end{pmatrix}$$

4. Vigenere Cipher

Create a function that implements the Vigenere cipher algorithm. The function takes three inputs: plaintext as string, key as string, and mode as bool, where true is auto mode and false is repeating mode. The output is a string representing the ciphertext. *Use the following keys: pie (repeating mode), aether (auto mode)*

5. Vernam Cipher

Create a function that implements Vernam (One Time Pad) cipher algorithm. The function takes the plaintext and the key as string inputs. The output ciphertext should also be a string. *Use the following key: SPARTANS*

• Implementation Notes

- You are free to use any programming language you prefer, but I recommend sticking to the same language in every assignment so in the end you will have a complete library of your own making that includes the encryption techniques you study.
- The input must be read from the *.txt files attached with this file. Each problem has its own plaintext file to read the plaintext line by line. You are then required to save your output to another *.txt file (one for each plaintext file) which you will submit along with your code.
- Write the output of every algorithm in a separate text file using the plaintexts in each input file.
- Create an executable ".exe" file for your program in which you can input a plaintext, specify the key for every one of the five techniques, and run the program to print out the ciphertext output of all five techniques for the same input plaintext.
- Make sure that your code files and the text files they read from or write to are in the same directory and use a relative path when accessing these files.

- **Submission Details**
  - You are required to submit your code files (not entire projects) along with the pairs of plaintext and ciphertext files.
  - Have each problem solved in a self-contained function and then group all these functions in one code file with a main function where you can call your algorithms.
  - Write a one-page report about the design and documentation of your library.
  - Group all files including the code and executable into one folder and compress it.
  - Rename the compressed file to **"CSE436_Assignment1_StudentID.rar"** format, and submit it to google classroom.
  - Please adhere strictly to the above instructions as they will significantly facilitate marking your answers.