Cryptography: Coding Lab 1

Due: Tuesday, Sep. 24, 2024

Goal: Implement a computer program utilizing modular arithmetic that can encrypt and decrypt messages using a Caesar shift cipher and a transposition cipher.

Part 1: Caesar Shift

1) Below, write pseudocode for a program that will ask for a plaintext message and an integer key between 1 and 26 and output the appropriate cipher text.
Python: define a function `Cencrypt` that performs the above algorithm.

```
def Cencrypt():
    plaintext = input("Enter plaintext:").upper()
    key = int(input("Enter key (1-26):"))

    # Ensure key is between 1 and 26
    if not 1 <= key <= 26:
        print("Key must be between 1 and 26.")
        return

    # Filter out non-alphabetic characters
    plaintext = ''.join(filter(str.isalpha, plaintext))

    # Shift each letter by the key
    ciphertext = ''.join(chr((ord(c) - ord('A') + key) % 26
                             + ord('A')) for c in plaintext)

    print("Ciphertext:", ciphertext)
```

2) How would this program need to be changed to decrypt a cipher text given a key?
Python: define a function `Cdecrypt` and have the user choose encrypt/decrypt upon opening the program.

3) How would the decrypt function be changed to allow a brute force attack if we did not know the key?
Python: Add an option to the decrypt function so if the person does not know the key, the program will perform a brute force attack.

4) Check that your program works with the following message/ciphertext pair `e(3, 'hello world') = KHOORZRUOG`

5) Choose a message of your own to encrypt and then send the ciphertext to another group for them to decrypt (without the key).
Write down your message/ciphertext/key here.

Write the other group's ciphertext and your decryption of it here, along with their key.

Part 2: Transposition Cipher

1) Below, write pseudocode for a program that will ask for a plaintext message and an integer key and output the appropriate cipher text.
   Python: define a function `Tencrypt` that performs the above algorithm.

2) How would this program need to be changed to decrypt a cipher text given a key?
   Python: define a function `Tdecrypt` and have the user choose encrypt/decrypt upon opening the program.

3) How would the decrypt function be changed to allow a brute force attack if we did not know the key?
   Python: Add an option to the decrypt function so if the person does not know the key, the program will perform a brute force attack.

4) Check that your program works with the following message/ciphertext pair `e(3, 'hello world') = HOLEWDLOLR`

5) Choose a message of your own to encrypt and then send the ciphertext to another group for them to decrypt (without the key). Write down your message/ciphertext/key here.

   Write the other group's ciphertext and your decryption of it here, along with their key.