Ira A. Fulton Schools of Engineering
Arizona State University

CSE 543: Information Assurance and Security

Caesar Cipher

Purpose

The purpose of this assignment is to familiarize you with basic encryption schemes and how to derive a decryption function from a known encryption function. Additionally, this assignment will familiarize you with Python (or the programming language you choose) and writing a simple decryption function in Python (or your programming language). You will learn how to read encryption code, deriving the corresponding decryption function (which is the inverse function of the encryption function), implementing the decryption function, and applying the decryption function on a given input string.

Objectives

Students will be able to:

- Read encryption implementation of the Caesar cipher.
- Implement their own decryption function for a Caesar cipher.
- Deal with files programmatically.
- Decrypt a given ciphertext that is encrypted using a Caesar cipher and its corresponding key.

Technology Requirements

The reference encryption code is provided as a Python script; however, you may implement your solutions in any programming language that you choose.

Note: The course team will not be able to help you if you choose any language that is not Python, Java, or C#; therefore, to create the best learning experience, Python is strongly recommended.

Project Description/Directions

For this project, you will use the three files:

1. `ciphertext.txt` - A ciphertext encrypted with a Caesar cipher.

2. `caesarkey.txt` - The key used for the Caesar cipher. It is the stdout output when running `caesar.py` to encrypt the plaintext.
3. `caesar.py` - The reference implementation of the Caesar cipher with only the encryption feature.

You will need to write some code so you can plug in the key and decrypt. Here are recommended steps to solve the problem:

1. The Caesar cipher that is used in this assignment is a standard (and ancient) encryption algorithm. You may find its description on Wikipedia. Please read the description first.
2. (Optional) A reference implementation of the Caesar cipher is provided in `caesar.py`. If you are new to Python, you are strongly recommended to read and understand the implementation.
3. Open `caesarkey.txt` and find the key: the offset that was used during encryption.
4. Think about how the encryption can be negated.
5. Write a Python program that will open `ciphertext.txt`, read out the content, shift the characters by the given offset (which is the decryption process), and write the decrypted content into a new file.

The alphabet for the plaintext and ciphertext is the capital letters [A-Z], and the key is a number between 0 and 25. This means your decrypted text (plaintext) should only have capital letters.

Debugging tips: In case you find your decryption code not working as expected, you may first manually decrypt the first few characters in `ciphertext.txt`. Then you may print out results after each statement in your code and verify the results against your manual decryption.

Evaluation

You will earn 100 points if your decryption result is fully correct. Otherwise, you will earn 0 points. Partial credit will not be granted for this project.

If your decryption works correctly, you will see multiple meaningful sentences in English as the plaintext (uppercase letters only, with no spaces between words). Otherwise please go back and fix your implementation.

Submission Directions for Project Deliverables

This assignment is autograded. In the submission space in this course, you will only submit the plain text `plaintext.txt`. Do not submit the original ciphertext or include the original ciphertext in your submission. You have unlimited submissions.