

Historic Ciphers

February 1, 2024

Welcome to the world of historic ciphers programming exercises! In these tasks, you'll delve into the fascinating realm of classical ciphers: Shift Cipher, Substitution Cipher, and Vigenère Cipher. Using the Python programming language, you have the freedom to explore and implement these ancient encryption techniques.

For seamless command-line interaction, you are encouraged to utilize the argparse library to manage command-line options efficiently. However, please note that while incorporating external libraries is encouraged for command-line handling, it is strictly forbidden to import cryptographic libraries for the actual encryption or decryption processes. This restriction aims to enhance your understanding of the historical cipher algorithms and encourages you to implement them from scratch using Python.

Get ready to embark on a journey into the cryptographic past as you tackle these engaging exercises and bring these classic ciphers to life with your Python coding skills!

To effectively manage command-line arguments, we recommend using the argparse library in Python. Below is a template code snippet that you can integrate into your programs for handling command-line options:

```
import argparse

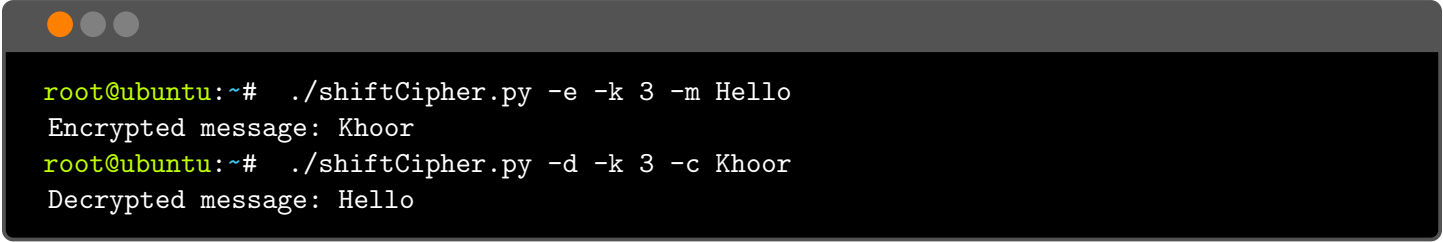
parser = argparse.ArgumentParser(description='Shift Cipher Encryptor/Decryptor')
parser.add_argument('-e', '--encrypt', action='store_true', help='Encrypt mode')
parser.add_argument('-d', '--decrypt', action='store_true', help='Decrypt mode')
parser.add_argument('-k', '--key', required=True)

group = parser.add_mutually_exclusive_group(required=True)
group.add_argument('-m', '--message', help='Message to encrypt')
group.add_argument('-c', '--cypher', help='Cypher text to decrypt')

args = parser.parse_args()
```

1 Shift cipher

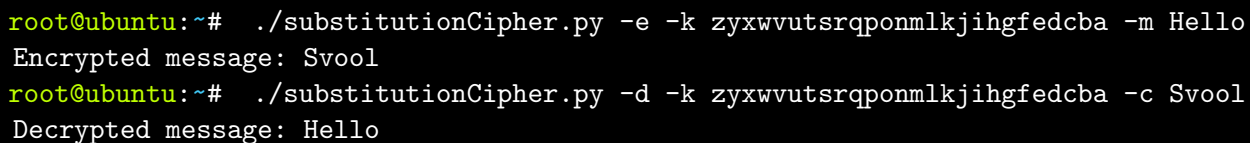
Create a program named "shiftCipher" that requires the argument "-e" for encryption or "-d" for decryption, along with "-k" followed by the key. If the encryption flag is set, include the "-m" flag followed by the message to encrypt. If the decryption flag is set, include the "-c" flag with the ciphertext to decrypt. Ensure the program implements the shift cipher algorithm and performs encryption or decryption accordingly based on the specified flags and print the encrypted or decrypted text to the standard output. An example usage for the program is provided below. The key needs to be an integer between 0 and 25.

A terminal window with a dark background and three colored window control buttons (orange, grey, grey) in the top-left corner. The terminal shows the execution of the shiftCipher.py program for both encryption and decryption.

```
root@ubuntu:~# ./shiftCipher.py -e -k 3 -m Hello
Encrypted message: Khoor
root@ubuntu:~# ./shiftCipher.py -d -k 3 -c Khoor
Decrypted message: Hello
```

2 Substitution cipher

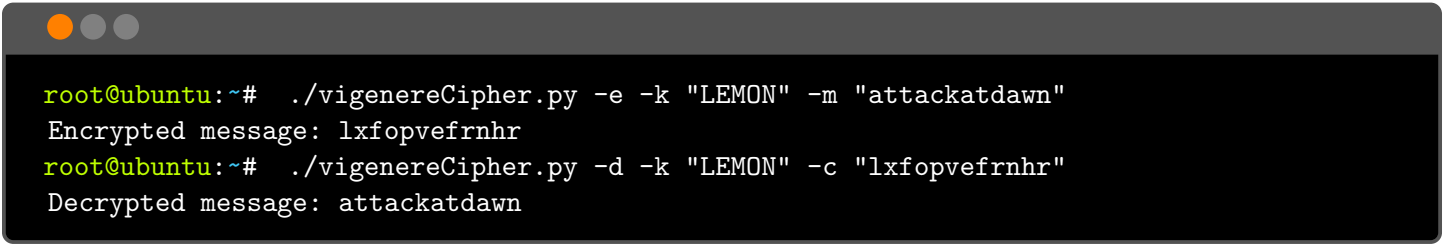
Create a program named "substitutionCipher" that requires the argument "-e" for encryption or "-d" for decryption, along with "-k" followed by the key. If the encryption flag is set, include the "-m" flag followed by the message to encrypt. If the decryption flag is set, include the "-c" flag with the ciphertext to decrypt. Ensure the program implements the substitution cipher algorithm and performs encryption or decryption accordingly based on the specified flags and print the encrypted or decrypted text to the standard output. An example usage for the program is provided below. The key is a 26 character string representing a different alphabet or the same alphabet in a different order.

A terminal window with a dark background and three colored window control buttons (orange, grey, grey) in the top-left corner. The terminal shows the execution of the substitutionCipher.py program for both encryption and decryption using a 26-character key.

```
root@ubuntu:~# ./substitutionCipher.py -e -k zyxwvutsrqponmlkjihgfedcba -m Hello
Encrypted message: Svool
root@ubuntu:~# ./substitutionCipher.py -d -k zyxwvutsrqponmlkjihgfedcba -c Svool
Decrypted message: Hello
```

3 Vigenere cipher

Create a program named "vigenereCipher" that requires the argument "-e" for encryption or "-d" for decryption, along with "-k" followed by the key. If the encryption flag is set, include the "-m" flag followed by the message to encrypt. If the decryption flag is set, include the "-c" flag with the ciphertext to decrypt. Ensure the program implements the vigenere cipher algorithm and performs encryption or decryption accordingly based on the specified flags and print the encrypted or decrypted text to the standard output. An example usage for the program is provided below. The key is a string containing only upper or lower case letters and needs to be bigger than 1.



```
root@ubuntu:~# ./vigenereCipher.py -e -k "LEMON" -m "attackatdawn"  
Encrypted message: lxfopvefrnhr  
root@ubuntu:~# ./vigenereCipher.py -d -k "LEMON" -c "lxfopvefrnhr"  
Decrypted message: attackatdawn
```