CS336 Homework #2

Due: Friday, Mar 24th, 2023, 5:00pm
Turn in: C/C++ code file and a word/pdf file contains one or more screenshots to show your program is running correctly. Please name your C++ file with **first name_last name-hw2_code.cpp,** and name your word/pdf with **first name_last name-hw2_file.doc (or .pdf). Don't zip them together, just turn in both files.**
For example, Alice_Zhang-hw2_code.cpp and Alice_Zhang-hw2_file.doc
Points: 50 pts

In this assignment, you will implement the CAESAR cipher with C/C++.

Note: Assume users enter letters (uppercase and/or lowercase letters) only (No digits or symbols). And your cipher program will generate ciphertext/plaintext in UPPERCASE LETTERS only. That is, if user enters lowercase letters, covert them to uppercase letters first and then run through the encryption or decryption algorithm. The output should be UPPERCASE LETTERS only.

(10 pts) Part 1: CAESAR Cipher encryption

- ⇒ Ask user to input plaintext and a key (key should be a number).
- ⇒ Then your program should feed the plaintext and the key to the Caesar cipher encryption algorithm and display the ciphertext.

(10 pts) Part 2: CAESAR Cipher decryption with a given key

- ⇒ Ask user to input ciphertext and a key.
- ⇒ Then your program should feed the ciphertext and the key to the Caesar cipher decryption algorithm and display the plaintext.

(10 pts) Part 3: CAESAR Cipher decryption without a known key

Assume the user doesn't know the key, your program should do brute-force attack (try all possible keys) on the ciphertext and show all possible plaintext.

- ⇒ Ask user to input ciphertext.
- ⇒ Your program should try all possible keys and display all possible plaintext (and the key for each one).

(5 pts) Overall code structure.

You may need to have a menu the let users pick what they want to do.
Loops - Users will choose to continue the program or exit the program.

(15 pts) Test your program and take one or more screenshots when running your program:

- ⇒ To test part 1 and part 2, try with your own messages and keys.
- ⇒ To test part 3, use the following ciphertext and display all possible plaintext (and the corresponding key)

```
SK S UK KLMVWFL QGM FWWV LG UGVW JWYMDSJDQ SFV XJWIMWFLDQ
```