WICHITA STATE UNIVERSITY – CS 211

Homework #6 – Hail Caesar (shift)!

DUE: October 22 by 11:59:59 PM **Assigned**: October 15

Background

A Caesar shift is one of the oldest forms of encryption. It was originally implemented by using two strips of paper with the alphabet written on them. One of the strips was then shifted a certain number of spaces over. For example, a shift of 3 would change the letter ‘A’ -> ‘D’, ‘B’ -> ‘E’, and so on. ‘Z’ -> ‘C’, meaning that the alphabet wrapped back around to the beginning.

This assignment will allow the user to enter a message, we will then encode it by shifting it 4 characters over and decode it again.

Assignment Requirements

- The name of your source code file shall be `hw06.cpp`
- Declare 2 functions, one to encode and the other to decode the message
  - **IMPORTANT**: Any circumvention of these functions for the sake of “correct output” will result in a grade of 0 for this assignment.
- Declare the necessary constants in your main function
- When the user is done typing the message, they will hit the Enter key. That keystroke is **not** to be saved to the array
- Non-letter characters will be ignored and not encrypted nor decrypted

Sample Run

```
Please enter a message: zip zap zooey

ZIP ZAP ZOOEY
DMT DET DSSIC
ZIP ZAP ZOOEY
```

Hints

- Recall that a character can be thought of as a small integer. We can use this to our advantage. https://www.asciitable.com
- For simple encryptions, decoding is simply reversing the encoding process
  - Encoding will likely be a mix of the actual encode process and a character control process
- The requirement concerning the Enter key can be taken care of for you by your method of input

Reminders

- Be sure to include a comment block at the top of every file with the required information

    - Refer to the General Homework Requirements handout on Blackboard
- Provide meaningful comments
    - If you think a comment is redundant, it probably is
    - If you think a comment is helpful, it probably is
    - Remember that you are writing comments for other programmers, not people who know nothing (obligatory Jon Snow) about coding
    - Comments are more helpful when they explain why, not what or how
- There will be no extensions

Preparing and Submitting

- Your code must be able to compile and run on the EECS lab machines
    - You are responsible for testing your code
    - "But it runs fine on my machine!" will **not** earn you any points
- Submit **ONLY** your source code file
- Homework submission will be handled exclusively through Blackboard