Homework #6

Hail Caesar (shift)!

Due: November 1 by 11:59:59 PM **Assigned:** October 25, 2018

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' to 'D', 'B' to 'E', and so on. 'Z' would shift back around to 'C'.
This assignment will allow the user to enter a message of up to 200 characters, we will then encode it by shifting it **4** characters over, and decode it again
NOTE: There is extra credit that allows earning up to 120% on this assignment. Refer to the extra credit section

Requirements:

- Name the source file for your program `program6.cpp`
- The use of C-Strings, std::string objects, and all related functions is **not** allowed for this assignment
- Declare two constant integers in the global space:
  Name the first one `CAP` and initialize it to a value of 200
  Name the second one `SHIFT` and initialize it to a value of 4
- Declare two functions, one to encode and the other to decode the message
- The message must be stored in an array of characters, which **must** be declared in your `main` function. The capacity of this array is `CAP`
- The message must be converted to upper case
  - You are only expected to encode and decode **letters**, not numbers or symbols
  - This means, do NOT encode numbers or symbols
- When is the user is done typing the message, they will hit the enter key. That keystroke is **not** to be saved to the array
- A sample run of your program should look like:

```
[WSUID@localhost Homeworks]$ ./prog6
Please enter a message (200 character limit): zip zap zooey

ZIP ZAP ZOOEY
DMT DET DSSIC
ZIP ZAP ZOOEY
[WSUID@localhost Homeworks]$
```

Hints:

- **IMPORTANT**: Any attempt to circumvent the encode/decode process to simply have "correct" output will be considered cheating, and you will fail the course (refer to the syllabus for my policy on academic integrity)
- Recall that `char` is also a small integer. We can use this to our advantage
  - `http://www.asciitable.com/`
- For simple encryptions like this, decoding is simply reversing the encoding process
  - Encoding will likely be a mix of the actual encode process and a character control process
- In order to read in the entire message (including whitespace) and perform proper processing, `cin.get()` can be helpful
- EXTRA CREDIT HINT: You may need to declare more than one array

Extra Credit:

- For 9 points (20%) extra credit on this assignment, perform a rail-fence cipher on top of the Caesar shift (2 layers only)
- Information about rail-fence cipher: `https://en.wikipedia.org/wiki/Rail_fence_cipher`
- The same sample run should look like this:

```
[WSUID@localhost Homeworks]$ ./prog6
Please enter a message (200 character limit): zip zap zooey

ZIP ZAP ZOOEY
DTDTDSCM E SI
ZIP ZAP ZOOEY
[WSUID@localhost Homeworks]$
```

Reminders:

- Be sure that your program includes your name, ID, description, etc. as shown in the General Homework Requirements Handout
- Use good style including indentation, comments, etc. Part of the grade will be for style and quality.
- Carefully test your program.
- You are welcome to write your program at home. If you do, be sure to compile and test it in the lab before submitting it.

How to submit your program:

- Submit the file `program3.cpp` electronically using the following terminal command:
  For the 12:30 lecture section:
  ```
  ~cs211a/bin/handin 6 program6.cpp
  ```

  For the 5:35 lecture section:
  ```
  ~cs211b/bin/handin 6 program6.cpp
  ```