CSCI 1300 Introduction to Programming [Fall 2014]
Instructor: Boese
Assignment #9
Due: Friday **Nov 7, by 10am**.

Cryptography

Objectives

- Create functions
- Read and write to files
- Manipulate characters as integers

Encryption

Encrypting messages is one of the ways people and organizations can exchange information without allowing a third party to intercept and interpret them.

Background

The cryptography algorithm we will use is called a Caesar Cipher. With a Caesar cipher, you replace each letter in a message with a letter further along in the alphabet. A Caesar cipher shifts the alphabet and is therefore also called a Shift Cipher. The key is the number of letters you shift. Caesar cipher is one of the oldest types of ciphers. It is named after Julius Caesar, who is said to have used it to send messages to his generals over 2,000 years ago.

So, cryptology has existed for more than 2000 years. But, what is cryptology? The word cryptology is derived from two Greek words: kryptos, which means "hidden or secret," and logos, which means, "description." Cryptology means secret speech or communication

Mathematics of the Caesar Cipher

The mathematical transformation that shifts the alphabet is called a *translation*. In order to shift the letters, we first need to convert the characters to integer. Then you can add the offset key to the integer, and convert it back to a character to display the encrypted (or decrypted character). This works because characters are represented as numbers inside the computer, using the ASCII format. In ASCII, both the upper-case and lower-case letters are represented.

'A'	=	65	'a'	=	97
'B'	=	66	'b'	=	98
'C'	=	67	'c'	=	99
… etc.			… etc.		
'Z'	=	90	'z'	=	122

Note: In C++, chars are represented as integers so you can add an integer to a char variable!

In order to convert beyond the last letter of the alphabet, you will need to restart back at the start of the alphabet. For Example, adding 3 to 'y' should wrap back to 'b'.

For our cryptography, we are only going to encrypt the alphabet (not numbers or symbols). Therefore, any numbers or symbols in the plain text message will also show up in the encrypted text. You will need to handle both the lower-case and upper-case letters.

For this assignment you will write a program that takes three command-line arguments:

- the first one will be either "-e" to encrypt the file or "-d" to decrypt the file,
- second one will be the key to use (offset)
- and the third command-line argument will be the name of the file.

You will read in the file, and then based on the first command-line argument either encrypt or decrypt the message in the file. Then write the output to a new file.
For example,

```
./a.out  -e  5  message.txt
```

This program run will encrypt the contents in the file message.txt using an offset key of 5. With an offset key of 5, 'a' will become 'f', 'b' will be come 'g', etc.
An example run follows

message.txt:

```
These are fun ways to learn C++!
```

./a.out –e 5 message.txt

```
Encrypting file: message.txt
These are fun ways to learn C++!
Ymjxj fwj kzs bfdx yt qjfws H++!
```

The output file message.txt.enc now has:

```
Ymjxj fwj kzs bfdx yt qjfws H++!
```

Another example run,

This program run will decrypt the contents in the file using the offset key of 5. The offset key is subtracted this time, such that 'g' will become 'b'.

message.txt.enc

```
Ymjxj fwj kzs bfdx yt qjfws H++!
```

./a.out –d 5 message.txt.enc

```
Decrypting file: message.txt.enc
Ymjxj fwj kzs bfdx yt qjfws H++!
These are fun ways to learn C++!
```

The output file message.txt.enc.dec now has:

```
These are fun ways to learn C++!
```

Assignment Details:

- The name of the file must be called **Crypto.cpp**
- Comments at the top of your program
    - Your name
    - Date
    - Assignment #9
    - Brief description of the assignment (one or two lines max)
- Comments throughout your program explaining what it is doing
- The program takes three command-line arguments:

    [-e or –d]   offset_key   responses_filename

- Assume the message file is all in one line in the file.
- You must write the following functions (and name them exactly as specified)
    - `readFile` that takes the name of the file as a parameter and returns the contents of the file as a string.
    - `encryptChar` takes a single character to encrypt and the offset key as parameters and returns the encrypted character
    - `decryptChar` takes a single character to decrypt and the offset key as parameters and returns the decrypted character
    - `encryptMessage` that takes a string with the contents of the message to encrypt and the offset key as parameters and returns a string of the message encrypted. This function should call encryptChar.
    - `decryptMessage` that takes a string with the contents of the message to decrypt and the offset key as parameters and returns a string of the message decrypted. This function should call decryptChar.
    - `writeToFile` that takes the string of the either encrypted or decrypted message, the original filename, and the command type ("-e" or "–d"), returns nothing, and writes the message to a file with a filename the same as the original with an additional ".dec" extension added if the message was decrypted, or ".enc" extension added if the message was encrypted.
    - `main` function that coordinates call the other functions.
- You may not write your own classes, linked lists, or Vectors (e.g., do not use things that we have not covered yet).
- The output must match exactly to the examples provided (given appropriate inputs).
- The name of the functions must be exactly as specified, and the parameters and return values must be exactly as specified.
- If you do not use command-line arguments as specified then you will receive a zero for this assignment.
- You may only read the data file once!
- *We will run your program with different input files. So test by changing and using different example files!*
- Program must be written in C++ and submitted in Moodle.
- You may use the COG system to get feedback. COG will not give grades.
- If your code does not compile you will get a zero for this assignment.
- Zip the Crypto.cpp and submit to Moodle as ***Firstname_Lastname_HW9.zip***.