Homework Assignment #2

CS5004 – Object-Oriented Design
Northeastern University – Silicon Valley
Summer 2019

Due 05/26 at 11:00pm PDT

Instructions: You must submit a `.java` file per problem. Note that we use Java 8 or higher. Submit your files on Blackboard. Email submissions are not accepted.

Grading: Each programming problem is graded as follows

- A submission which does not compile gets 0.
- A submission which compiles but does something completely irrelevant gets 0.
- A submission which works (partially) correctly, gets (up to) %80 of the total credit.
- %20 is reserved for the coding style. Follow the coding style described in the book.

Problem 1 [30pts]. According to historians, Julius Caesar used the following encryption technique. Given a string s and an integer k, replace every letter by the letter than comes k positions later in alphabetic order. For example the string "hello" with *key* 5, is encrypted into "mjqqt" since letter 'm' is the fifth letter after 'h' and so on. Note that 'z' in this case is mapped to 'e'.

In this problem you will write two programs ENCRYPT.JAVA and DECRYPT.JAVA which are expected to work as follows. We use encrypt as follow

```
encrypt input-file output-file
```

The program must first check if the input file exists. If so, the program prompts the user to enter a key k. Then, it should encrypt the input file and write the encrypted file to the output file. For example, suppose we have a text file "test.txt" residing in the same folder as the java program. Next, we call

```
encrypt test.txt out.txt
```

to get the encrypted output written into the file “out.txt” which should be overwritten if such a file exists.

The decrypt program works in the opposite direction; i.e. it receives as input two files. It asks for a key. Then, it decrypts the first one and write the result in the second file.
NOTE: You can assume that the text files contain only alphabet letters both lower and upper case. You must keep the case; i.e. a lower case letter should be mapped to a lower case letter.
NOTE: The key k is a positive integer. But, it can be a large number. In this case, you must wrap it around. For example if the key is 27 then ’a’ is mapped to ’b’ etc.
NOTE: The input file and the output file are passed as command-line arguments. Do a quick search on the Internet to learn how it is done (use args).
NOTE: Starting with a file “t.txt” and applying encrypt and decrypt back-to-back (with a fixed key!), we expect to get “t.txt” back intact!

Problem 2 [20pts]. Given a text file containing decimal numbers separated by commas, we need to compute the average, the minimum, the maximum, and the variance of the list of numbers. Quick example of an input file is

$$12.2, 33.09, 22.00$$

Of course, the input file can contain hundreds of numbers!