Homework Assignment #3

CS5004 – Object-Oriented Design
Northeastern University – Silicon Valley
Summer 2020

Due Sunday 05/31 at 11:00pm PT

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 for sending messages. Given a string *s* and an integer *k*, replace every letter by the letter than comes *k* positions later in alphabetical 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' (since after 'z' we go back to 'a'). To decrypt the message one needs to know the key; for our example, just "shift" each letter by -5 positions.

In this problem you will write a program called `Cipher.java` which does both encryption and decryption. Some examples follow

```
-- Enter the text below
Hello
-- Enter E for Encryption, D for Decryption, X to exit: E
-- Enter key: 5
-- The encrypted text is below
Mjqqt
```

Now another example

```
-- Enter the text below
Mjqqt
-- Enter E for Encryption, D for Decryption, X to exit: D
-- Enter key: 5
-- The decrypted text is below
Hello
```

Another example

```
-- Enter the text below
Hello
-- Enter E for Encryption, D for Decryption, X to exit: F
-- Input not recognized. Try again.
-- Enter E for Encryption, D for Decryption, X to exit: X
```

NOTE: You can assume that the input contain only alphabet letters, both lower case and upper case, space, and the usual punctuation marks $\{., !?\}$. Non alphabet characters are to be left *unchanged*. Also, 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. Your code is expected to work with any non-negative integer. Please check that input key is valid (that is $key \geq 0$; if not print an error message and exit).

Problem 2 [10pts]. In discrete math, the n-the Harmonic number is defined as

$$H_n = \sum_{k=1}^{n} \frac{1}{k} = \frac{1}{1} + \frac{1}{2} + ... + \frac{1}{n}$$

Write a program that takes as input an integer $n > 0$ and outputs H_n. An example:

```
-- Enter n: 100
5.187377517639621
```

If the input is not valid (e.g. -1), your program should print an error message and exit.