AP Computer Science A@Beijing National Day School

Lab 6: `Caesar Cipher`

Due date: Friday, November 16, 2018
Instructor: Mr. Alwin Tareen
Total Points: 15

Task Overview

- Implement a program that encrypts a message using the caesar cipher.

Background

- The main idea behind the Caesar Cipher is to shift each letter in a secret message by a fixed number of positions. If this shifting behaviour goes further than the end of the alphabet, then it "wraps around" to the beginning, and continues from there.
- The security of this crypto-system relies on having only the sender and the recipient know the secret **key**, which is the number of places by which the letters have been shifted.

Table 1. Encrypting HELLO with a key of 1 yields IFMMP.

plaintext	H	E	L	L	O
+ key	1	1	1	1	1
= ciphertext	I	F	M	M	P

Specification

The Information Box Which Includes Your Name[5 points]

- Type your English and Pinyin name into the `Author` field, where it says: `YOUR NAME HERE`

Encrypt a Message with the Caesar Cipher [10 points]

- Write a `Java` program in the file `CaesarCipher.java` that encrypts a message using the caesar cipher.
- You will write your solution in a function called: `public static String encrypt(String message, int key)` right below the place where it says: `YOUR CODE HERE`.
- Make sure that you run your `Java` program, and ensure that it is free of errors. When the following statements are executed:

  ```
  String result = encrypt("hello", 1);
  System.out.println(result);
  ```

  The output of your program should be: `ifmmp`

Hints

- Unencrypted text is generally called **plaintext**, and encrypted text is generally known as **ciphertext**. The quantity by which the letters have been shifted is called a **key**.
- In general, the Caesar Cipher encrypts messages by "rotating" each letter by `key` positions. More formally, if p is the `alphabet` index of a plaintext letter, and `key` is the amount by which that letter is shifted, then the `alphabet` index of the corresponding letter in the ciphertext c, is computed by the following equation:

$$c \quad = \quad (p + \text{key}) \bmod 26$$

- You may assume that all of the characters in the plaintext messages are in **lowercase**.
- There will be no punctuation present in any of the plaintext messages, with the exception of the space character. You should design your program so that any spaces in the plaintext message are transferred into the encrypted ciphertext.
- Note that the complete lowercase `alphabet` has been provided for you:

  ```
  String alphabet = "abcdefghijklmnopqrstuvwxyz";
  ```

  This means that `"a"` corresponds to index `0`, `"b"` corresponds to index `1`, etc.

Testing

- The file `CaesarCipherJUnitTest.java` contains the `JUnit` test cases which verify the correct functionality of the program.

Submission

- Submit your `Java` program by uploading it to the `Web-CAT` automated grading platform: `http://ec2-54-65-207-33.ap-northeast-1.compute.amazonaws.com:8080/Web-CAT/WebObjects/Web-CAT.woa`