Lab 06 - Modules & Namespaces

Instructions:

- **The lab requires completing a few tasks.**
- **Your submissions must be submitted to the GitHub repository in the Lab06 directory.**
- **Cheating of any kind is prohibited and will not be tolerated.**
- **Violating and failing to follow any of the rules will result in an automatic zero (0) for the lab.**

TO ACKNOWLEDGE THAT YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS ABOVE, AT THE BEGINNING OF YOUR SUBMISSION(S), ADD A COMMENT THAT CONSISTS OF YOUR NAME AND THE DATE.

Grading

Task	Maximum Points	Points Earned
1	2.5	
2	2.5	
Total	5	

There are several avaialble ciphers. The some simple ciphers are

Casear Cipher
E = (P + 3) % M
Affine Cipher
E = (P * A + K) % M

where E, P, M, A, and K are encoded character, plaintext character, size of the plaintext, multiplier, and shifter, respectively. Additionally, A and M must be relatively prime (that is, they are no common factors except for 1). Their deciphers are

Casear Cipher
P = (E + -3) % M
Affine Cipher
P = A^{-1} * (E + -K) % M

where -3 and -K are the additive inverses of 3 and K, respectively, modulo M and A^{-1} is the multiplicative inverse of A modulo M. Your objective is to

1. Create a string constant named *PLAINTEXT* that is initialized to the concatenation of the uppercase alphabet and the digits within a namespace named **Ciphers**.
2. Create namespaces **Casear** and **Affine** within the namespace **Ciphers**.
3. In each namespace **Casear** and **Affine**, define the following functions
   - A string function named `encode()` that takes a string parameter. It returns an encrypted string of the parameter by the appropriate cipher if the parameter only contains case-insensitive characters of the *PLAINTEXT* and spaces; otherwise, it returns an empty string. Spaces are not encrypted.
   - A string function named `decode()` that takes a string parameter. It returns a decrypted string of the parameter by the appropriate cipher if the parameter only contains case-insensitive characters of the *PLAINTEXT* and spaces; otherwise, it returns an empty string. Spaces are not decrypted.

   Furthermore, for the affine cipher, you choose A and K.
4. Within a main function,
   a. Prompt the user to enter a string.
   b. For each cipher, assign the encryption of the user's input to a variable.
   c. Display each encryption with its cipher name preceding it.
   d. For each encryption, assign its decryption to a variable.
   e. Display each decryption with its cipher name preceding it.

Task 1

Write the above program in C++.

Task 2

Write the above program in Ruby.