**Substitution cypher:**

How does it work?

Basically we make a permutation of the alphabet and you map each character from the alphabet to the permuted alphabet.

Encryption:

You parse the given input (plain text) and you change it character with the corresponding one from the permuted alphabet.

Decryption:

Decryption is the reverse operation of the encryption. You parse the cypher text and you change the character from the cypher text with the corresponding one from the alphabet.

Alphabet:

The alphabet is the set of allowed characters that can be used in composing the plain text, the cypher text obtained from the encryption operation and the original text obtained by the decryption process.

Constraints & Validations:

For our program we have the following constraints:

1. The alphabet is composed by the 26 characters from the English alphabet, plus the space character (\_ in our documentation) thus we have 27 possible characters.
2. The key must have exactly 27 characters from the alphabet mentioned above
3. Both fields for each operation (encryption and decryption) are required
4. Usually the plain text is lower case and the cypher text is upper case

Example:

Plaint\_text1 = “second example”

Key = “pynwlztxrvuosmqfjdhbk icgae”

Cypher Text = encryption (plaint\_text1, key) = BZWFQLPZGYMJSZ

Plain\_text2= decryption(cypher text, key) = Plain\_text1= “second example”