**Cyber Project**

We are supposing the architecture that multiple layers will take place while encrypting and decrypting the given text using a given key.

The structure looks something looks like this:

Encryption 1

Encryption 2

Encryption 3

Decryption 3

Decryption 2

Decryption 1

Encryption 1:

In this preliminary encryption, we performed substitution encryption with some modifications. We have added some encryption for special characters. For this, we have created a list of special characters and this character is replaced by the adjacent special character picked from the list. Each character in the string is shifted by some shift value ‘s’, provided it is not a special character. This value is calculated by adding all the ASCII values of the text and then transforming that number between 0-25. If the character in the string is uppercase, then the formula is (original\_character + s - 65) % 26 + 65). If the character is lowercase, then the formula used is: (original\_character + s - 97) % 26 + 97). Now this encrypted text is ready for the next layer of encryption.

Decryption 1:

This is the last part of the algorithm. It reverses what we have done in the first encryption just by modifying the shift value we used in **Encryption 1**. If the encrypted character is a special character then it is replaced by previous adjacent character. For lowercase character, the formula used is: (original\_character + (26-s) - 97) % 26 + 97). For the uppercase, the formula used is: (original\_character + (26-s) - 65) % 26 + 65). After this step, the original text string is retrieved.

Encryption 2:

In this encryption, we are breaking the text we get from previous encryptions in blocks having size equals to the length of the key. We then xor the ascii values of each letter in the block with the corresponding letter of the key. We also xor the indices of each letter in the text with the letter. This provides us with the final encrypted text.

For Example,

Text = “ABCDEFGHI”

Key = “XYZ”

Now text will be broken into,

[A^X^0] [B^Y^1] [C^Z^2]

[D^X^3] [E^Y^4] [F^Z^5]

[G^X^6] [H^Y^7] [I^Z^8]

Decryption 2:

XOR follows the property,

If A^B = C

Then C^A = B and C^B=A

Thus by the same property, if we XOR every letter in encrypted text with its index and the corresponding letter in the key, we will get the original text.

Thus following the previous example,

Let [A^X^0] = C

Then C^X^0 = A