Assignment No.

Title: Password data encryption

Aim: Write a program in python/ Java/ Scala/ C++/ HTML5 to implement password data encryption. Use encryption method overloading (any two methods studied).

Objective:

- Students learn about the relationship between cryptographic keys and passwords.
- Students explore the DES with a widget to examine how a cryptographic "key" can be used to encrypt and decrypt a message.

Theory:

Cryptography: The art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form.

Different techniques used for cryptography are:

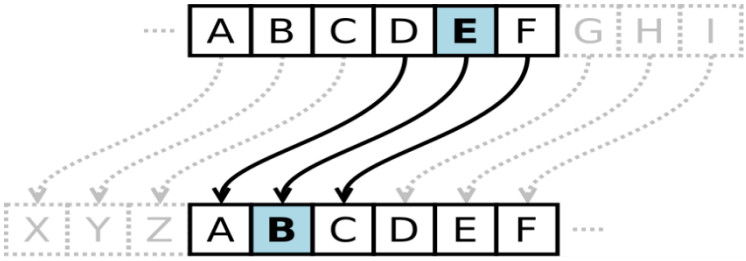
1) Substitution Techniques
    - Ceaser Cipher
    - Playfair Cipher
2) Transposition Techniques
    - Rail Fence Cipher
3) Hybrid
    - Data Encryption Standard

The **Caesar Cipher**, also known as a shift cipher, is one of the oldest and simplest forms of encrypting a message. It is a type of substitution cipher where each letter in the original message (which in cryptography is called the plaintext) is replaced with a letter corresponding to a certain number of letters shifted up or down in the alphabet.

For each letter of the alphabet, you would take its position in the alphabet, say 3 for the letter 'C', and shift it by the key number. If we had a key of +3, that 'C' would be shifted down to an 'F' - and that same process would be applied to every letter in the plaintext.

In this way, a message that initially was quite readable, ends up in a form that cannot be understood at a simple glance.

For example, here's the Caesar Cipher encryption of a full message, using a left shift of 3.



As unreadable as the resulting ciphertext may appear, the Caesar Cipher is one of the weakest forms of encryption one can employ for the following reasons:

- The key space is very small. Using a brute force attack method, one could easily try all (25) possible combinations to decrypt the message without initially knowing the key.
- The structure of the original plaintext remains intact. This makes the encryption method vulnerable to frequency analysis - by looking at h ow often certain characters or sequences of characters appear, one can discover patterns and potentially discover the key without having to perform a full brute force search.

The Caesar Cipher can be expressed in a more mathematical form as follows:

$$E_n(x) = (x + n)\ mod\ 26$$

In plain terms, this means that the encryption of a letter x is equal to a shift of x + n, where n is the number of letters shifted. The result of the process is then taken under modulo division, essentially meaning that if a letter is shifted past the end of the alphabet, it wraps around to the beginning.

Decryption of the encrypted text (called the ciphertext) would be carried out similarly, subtracting the shift amount.

$$D_n(x) = (x - n)\ mod\ 26$$

First used by Julius Caesar, the Caesar Cipher is one of the more well-known older historical encryption methods. While you certainly wouldn't want to use it in today's modern world, a long time ago it might have done the trick.

The **Playfair Cipher** is a manual symmetric encryption cipher invented in 1854 by Charles Wheatstone, however it's name and popularity came from the endorsement of Lord Playfair.

The Playfair cipher encrypts pairs of letters (digraphs), instead of single letters as is the case with simpler substitution ciphers such as the Caesar Cipher. Frequency analysis is still possible on the Playfair cipher, however it would be against 600 possible pairs of letters instead of 26 different possible letters. For this reason the Playfair cipher is much more secure than older substitution ciphers, and it's use continued up.

The playfair cipher starts with creating a key table. The key table is a 5×5 grid of letters that will act as the key for encrypting your plaintext. Each of the 25 letters must be unique and one letter of the alphabet (usually Q) is omitted from the table (as there are 25 spots and 26 letters in the alphabet).

Let's say we wanted to use the phrase "Hello World" as our key. The first characters (going left to right) in the table will be the phrase, with duplicate letters removed. The rest of the table will be filled with the remaining letters of the alphabet, in order. Our key table would look like this:

```
H E L O W
R D A B C
F G I J K
M N P S T
```

Now, we need a message to encrypt. In a playfair cipher the message is split into digraphs, pairs of two letters. If there is an odd number of letters, a Z is added to the last letter.

Let's say we want to encrypt the message "hide the gold".

```
HI DE TH EG OL DZ
```

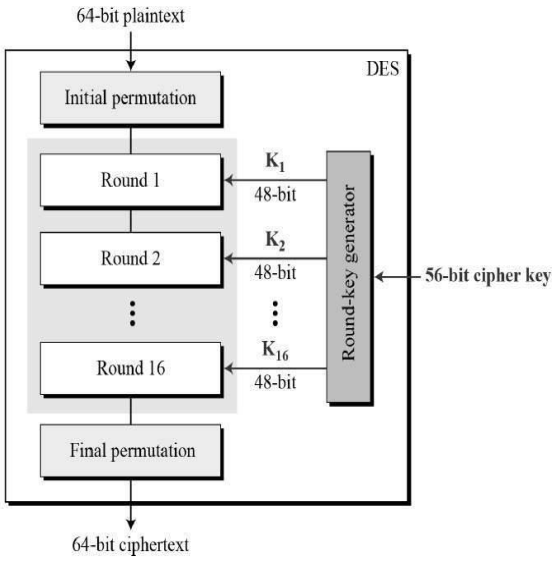
Now for the actual encryption process. The Playfair cipher uses a few simple rules relating to where the letters of each digraph are in relation to each other. The rules are:

- If both letters are in the same column, take the letter below each one (going back to the top if at the bottom)
- If both letters are in the same row, take the letter to the right of each one (going back to the left if at the farthest right)

- If neither of the preceding two rules are true, form a rectangle with the two letters and take the letters on the horizontal opposite corner of the rectangle

Using these rules, the result of the encryption of “hide the gold” with the key of “hello world” would be “LF GD MW DN WO CV”.

The **Data Encryption Standard (DES)** is a symmetric-key algorithm for the encryption of electronic data.DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only).Block diagram of DES is shown below –

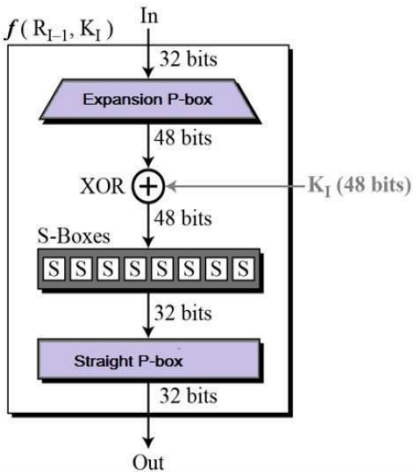


Since DES is based on the Feistel Cipher, all that is required to specify DES is −

- Round function
- Key schedule
- Any additional processing − Initial and final permutation

Initial and Final Permutation:-The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other.

Round Function:-The heart of this cipher is the DES function, f. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.



- **Expansion Permutation Box:-** right input is 32-bit and round key is a 48-bit, we first need to expand right input to 48 bits.
- **Substitution Boxes**:-The S-boxes carry out the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output.

  There are a total of eight S-box tables. The output of all eight s-boxes is then combined in to 32 bit section.

- **Straight Permutation:-** the 32 bit output of S-boxes is then subjected to the straight permutation with rule.

Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.

Conclusion: Thus we have studied and implemented Ceaser, Play fair, & DES Cipher for password data encryption.