Assignment Report

Encryption Algorithm:

Input in the algorithm:

- Plain text file in .txt format
- Key from the user

Main logic behind encryption algorithm is Bitwise XORing operation.

Algorithm use block cipher of block size 64 bit.

It takes input from user as ".txt" file format and convert it into bit format and store in block of size 64 bit.

The is also taken input from user and divided into blocks of size 64 bit.

A passphrase Block is generated using a paraphrase string which is already given.

Each plain text block is encrypted it with key block and its previous encrypted block.

Basically, here Differential XORing of bit block is used. Differential XORing means a file is scanned in blocks of bit and the output produced for each block is made a function of the output for the previous block.

One of the reasons behind use of deferential XORing is that it destroys the repetitive patterns in the messages to be encrypted and make it more difficult to break encryption by statistical analysis.

It involves applying the XOR operation between consecutive blocks of plaintext or ciphertext.

- The algorithm implements a block cipher where each block is of length 64.
- It reads plain text stream bitwise in blocks of 64 bits.
- The key is taken as input and is divided into blocks (each of 64 bit) and each block is XORed to get a single 64-bit block (call it keyBlock)
- Similar to the keyBlock generation, a paraphrase Block is generated using a paraphrase string.
- Each block of plain text is encrypted by XORing it with keyBlock and previously encrypted plain text block (except first plain text block)
- In case of first plain text block, the XORing is done with keyBlock and paraphrase Block
- At the end we have a bit stream which is converted into hexadecimal format for writeback in file

Decryption Algorithm:

- The algorithm implements a block cipher where each block is of length 64.
- It reads ciphertext.txt stream bitwise in blocks of 64 bits.
- The key is taken as input and is divided into blocks (each of 64 bit) and each block is XORed to get a single 64-bit block (call it keyBlock)
- Similar to the keyBlock generation, a paraphrase Block is generated using a paraphrase string.
- Each block of cipher text is decrypted by XORing it with keyBlock and previously encrypted plain text block (except first plain text block)
- In case of first plain text block, the XORing is done with keyBlock and paraphrase Block (initialization vector)
- At the end we have a bit stream which is converted into ascii format for write back in file

Crack_classical algorithm:

- Cipher text is read from ciphertext.txt file and convert it to bitvector
- And convert passphrase in to initialization vector and append it at starting of ciphertext block.
- Arrange it 64 bit of row and each 64 bit further divided into 8 column.
- Perform frequency analysis for each column completely one after other
- **Method 1:** Most frequent 8 bitvector is XOR with “00100000” it give key alphbate for that column and repeate this for all other column.
- In this way we get key.
- And using key we decrypt the ciphertext,txt file and write back to file.

Method II: We just replace the key by XORing the most frequent 8 bit by most occurring English alphabet.

In this way we get key and after decryption the file not in much readable form