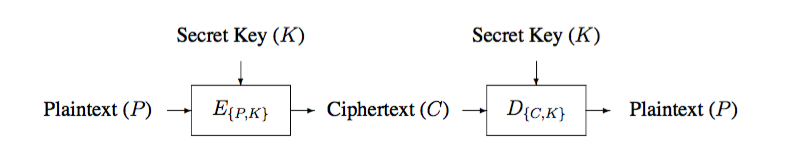
**Pavan Kumar Revanuru – 01578776**

**Computer Network Security - Project 1**

**Objective:**

To implement DES with EBC and CBC operations in Java from scratch.

**DES Algorithm:**

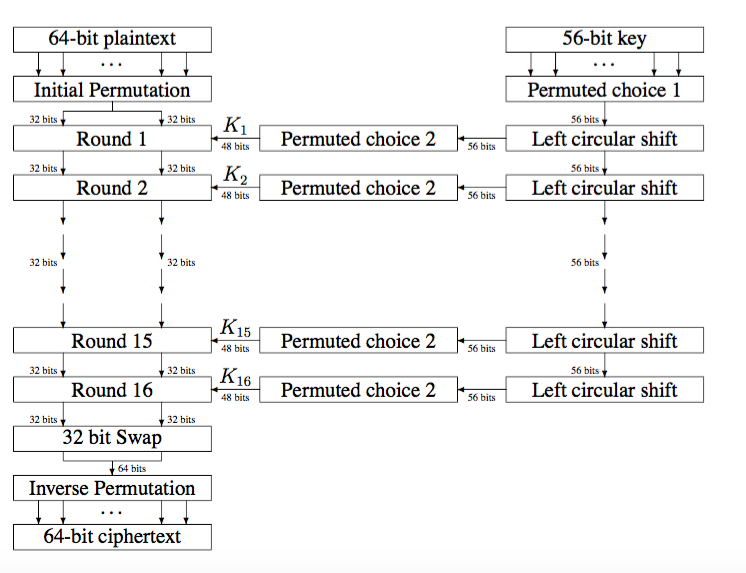


Initial permutation is used for rearranging the bits to form the “permuted input” followed by 16 iterations of the same function. The output of the last iteration consists of 64 bits which is a function of the plaintext and key. The left and right halves are swapped to produce the preoutput. Finally, the preoutput is passed through a permutation (IP−1) which is simply the inverse of the initial permutation (IP). The output of IP−1 is the 64-bit ciphertext.

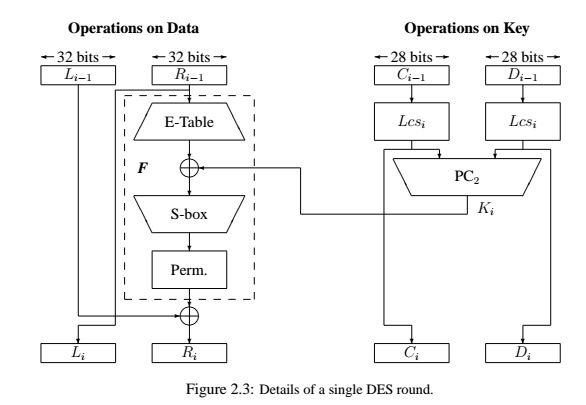
.

**Flow Chart of DES algorithm:**

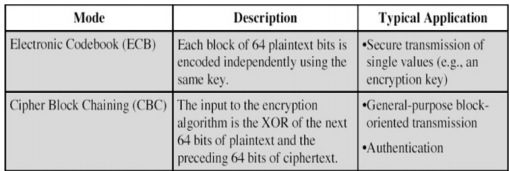
This is the flow chart followed to implement DES Encryption Algorithm.



The per-round algorithm is depicted below:



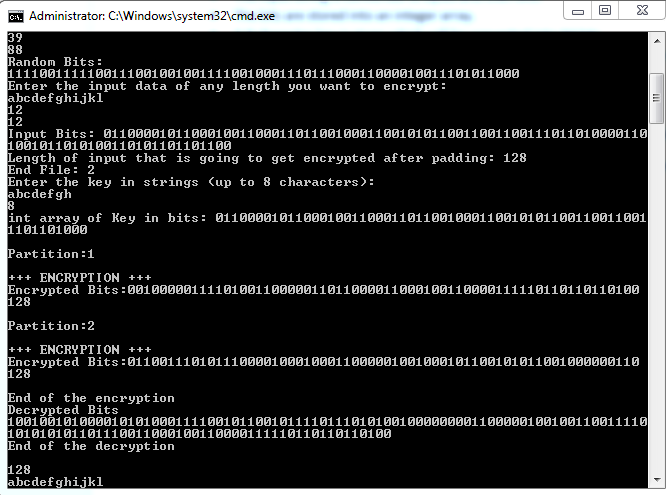
The typical difference between ECB and CBC is shown below:



The general working of code is as follows:

* First the random numbers are generated using the random function.
* The random numbers are converted into the bits.
* The input string that has to be encrypted is given when asked in the command prompt window.
* The input string is converted to bytes and then bits.
* The bits are stored into an integer array.
* A 8 character key is given which will be converted into 64 bits.
* The functions are executed in order such that each round is executed for sixteen times.
* Each round is executed as per DES.
* ECB is applied when there is no repetition of data in the input string.
* CBC is implemented when the data is repeated along the bits.
* The cipher text is again xor with the random number and the 64 bit blocks are decrypted.

The output is shown below:



**Conclusion:**

The given data is encrypted and decrypted using EBC and CBC successfully.