Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Experiment 4

Date of Performance : 27-02-2023 **Date of Submission: 04-03-2023**

SAP Id: 60004200107 **Name : Kartik Jolapara**

Div: B **Batch : B1**

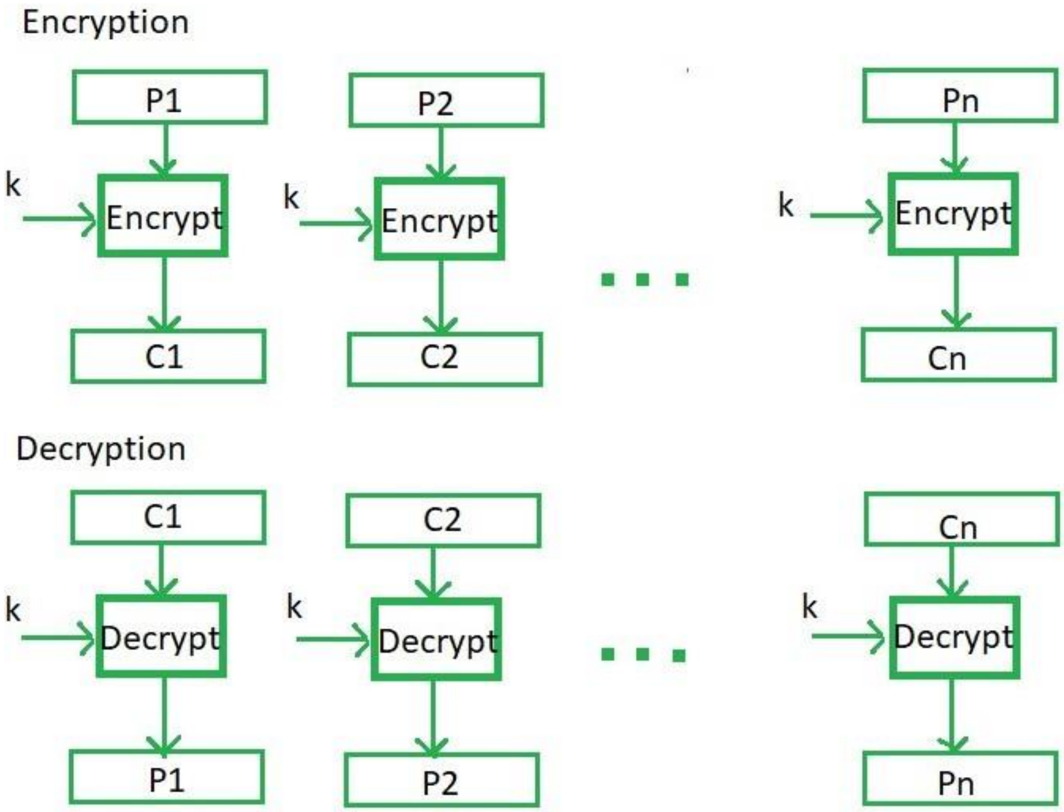
Aim of Experiment

Design and Implement Electronic Code Book(ECB) algorithmic mode. Plaintext is given as a paragraph. First, convert the given paragraph into ASCII values and then Binary. Use 128 bits of a block as input to ECB and encrypt using “t” bits shifter(Left/Right). Display encrypted paragraph.

Theory / Algorithm / Conceptual Description

Encryption algorithms are divided into two categories based on the input type, as block cipher and stream cipher. Block cipher is an encryption algorithm that takes a fixed size of input say b bits and produces a ciphertext of b bits again. If the input is larger than b bits it can be divided further. For different applications and uses, there are several modes of operation for a block cipher.

Electronic Code Book (ECB):
An electronic code book is the easiest block cipher mode of functioning. It is easier because of the direct encryption of each block of input plaintext and output is in form of blocks of encrypted ciphertext. Generally, if a message is larger than b bits in size, it can be broken down into a bunch of blocks and the procedure is repeated.



Advantages of using ECB:

- Parallel encryption of blocks of bits is possible, thus it is a faster way of encryption.
- Simple way of the block cipher.

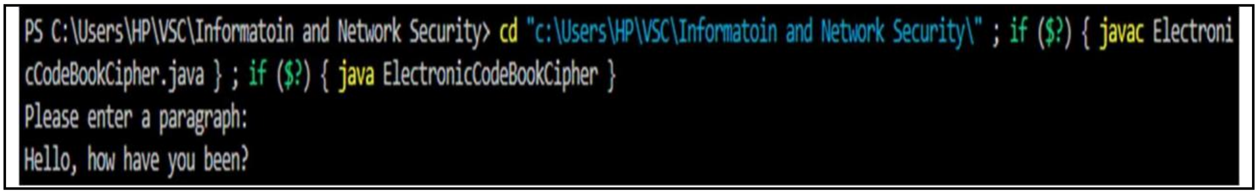
Disadvantages of using ECB:

- Prone to cryptanalysis since there is a direct relationship between plaintext and ciphertext.

Program

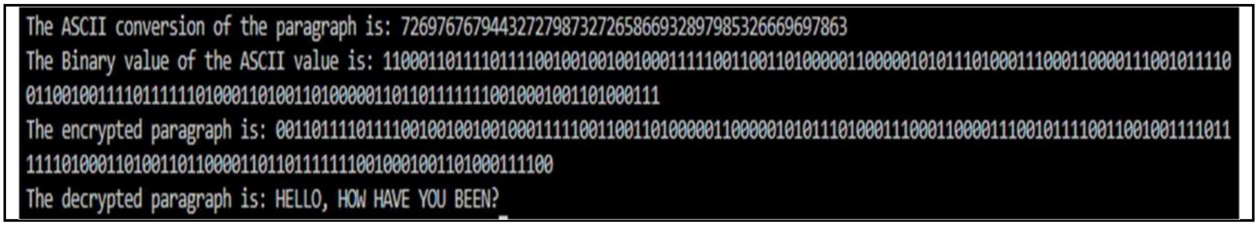
```
import java.math.BigInteger;
import java.util.Scanner;

class ECB {
    String ascii(String paragraph) {
        StringBuilder sb = new StringBuilder();
        for(char i: paragraph.toUpperCase().toCharArray()) {
            sb.append(Integer.valueOf(i));
        }

        return sb.toString();
    }

    String binary(String ascii) {
        return new BigInteger(ascii).toString(2);
    }

    String doLeftShift(String block, int t) {
        return block.substring(t) + block.substring(0, t);
    }

    String encrypt(String binary) {
        int count = 1;
        StringBuilder sb = new StringBuilder(), encryptedParagraph = new StringBuilder();
        for(int i = 0; i < binary.length(); i ++, count ++) {
            sb.append(binary.charAt(i));
            if(count % 128 == 0) {
                encryptedParagraph.append(doLeftShift(sb.toString(), 3));
                sb.setLength(0);
            }
        }
        if(sb.length() != 0) {
            encryptedParagraph.append(doLeftShift(sb.toString(), 3));
        }

        return encryptedParagraph.toString();
    }

    String doRightShift(String block, int t) {
        return block.substring(block.length() - t) + block.substring(0, block.length() - t);
    }

    String decrypt(String encryptedParagraph) {
        int count = 1;
        StringBuilder sb = new StringBuilder(), binary = new StringBuilder(),
decryptedParagraph = new StringBuilder();
        for(int i = 0; i < encryptedParagraph.length(); i ++, count ++) {
            sb.append(encryptedParagraph.charAt(i));
            if(count % 128 == 0) {
                binary.append(doRightShift(sb.toString(), 3));
                sb.setLength(0);
            }
        }
        if(sb.length() != 0) {
            binary.append(doRightShift(sb.toString(), 3));
        }

        String ascii = new BigInteger(binary.toString(), 2).toString(10);

        for(int i = 0; i < ascii.length(); i+= 2) {
            decryptedParagraph.append((char) Integer.valueOf(ascii.substring(i, i + 2)).intValue());
        }

        return decryptedParagraph.toString();
    }
}

public class ElectronicCodeBookCipher {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        ECB ecb = new ECB();

        System.out.println("Please enter a paragraph: ");
        String paragraph = in.nextLine();
        String ascii = ecb.ascii(paragraph);
        String binary = ecb.binary(ascii);
        String encryptedParagraph = ecb.encrypt(binary);

        System.out.println("The ASCII conversion of the paragraph is: " + ascii);
        System.out.println("The Binary value of the ASCII value is: " + binary);
        System.out.println("The encrypted paragraph is: " + encryptedParagraph);
        System.out.println("The decrypted paragraph is: " + ecb.decrypt(encryptedParagraph));
    }
}
```

Input



```
PS C:\Users\HP\VSC\Informatoin and Network Security> cd "c:\Users\HP\VSC\Informatoin and Network Security\" ; if ($?) { javac Electroni
cCodeBookCipher.java } ; if ($?) { java ElectronicCodeBookCipher }
Please enter a paragraph:
Hello, how have you been?
```

Output



```
The ASCII conversion of the paragraph is: 7269767679443272798732726586693289798532666969697863
The Binary value of the ASCII value is: 110001101111011110010010010010001111100110011010000011000001010111010001110001100001110010111110
01100100111101111110100011010011010000011011011111110010001001101000111
The encrypted paragraph is: 00110111101111001001001001000111110011001101000001100000101011101000111000110000111001011110011001001111011
11110100011010011011000011011011111110010001001101000111100
The decrypted paragraph is: HELLO, HOW HAVE YOU BEEN?
```