

IMPLEMENTATION OF CRYPTOGRAPHIC SOLUTIONS USING SYMMETRIC ENCRYPTION

AIM: To implement Caesar Cipher, Shift Cipher and the Brute Force attack on Shift Cipher.

CODE:

CAESAR CIPHER

```
#include <iostream>

#include <string>

using namespace std;

string Encrypt(string message, int key) {
    string encryptedMessage = "";

    for (char ch : message) {
        if (isupper(ch)) {
            encryptedMessage += char(int(ch + key - 65) % 26 + 65);
        }
        else if (islower(ch)) {
            encryptedMessage += char(int(ch + key - 97) % 26 + 97);
        }
        else {
            encryptedMessage += ch;
        }
    }

    return encryptedMessage;
}

int main() {
    string plaintext;
    int key;
```

21IT077

```
cout << "Enter the plaintext message: ";
getline(cin, plaintext);
//Setting key to be 3 (for Ceasar Cipher)
key=3;
string ciphertext = Encrypt(plaintext, key);
cout << "Encrypted ciphertext: " << ciphertext << endl;
return 0;
}
```

OUTPUT:

```
Enter the plaintext message: truestofallmembersunite
Encrypted ciphertext: wuxhvwridoophpehuvxqlwh
```

SHIFT CIPHER

```
#include <iostream>
#include <string>
using namespace std;

string encrypt(string plaintext, int shift) {
    string encryptedText = "";
    for (size_t i = 0; i < plaintext.length(); i++) {
        if (isupper(plaintext[i]))
            encryptedText += char(int(plaintext[i] + shift - 65) % 26 + 65);
        else if (islower(plaintext[i]))
            encryptedText += char(int(plaintext[i] + shift - 97) % 26 + 97);
        else
            encryptedText += plaintext[i];
    }
    return encryptedText;
}
```

21IT077

```
}
```

```
string decrypt(string ciphertext, int shift) {  
    string decryptedText = "";  
    for (size_t i = 0; i < ciphertext.length(); i++) {  
        if (isupper(ciphertext[i]))  
            decryptedText += char(int(ciphertext[i] - shift - 65 + 26) % 26 + 65);  
        else if (islower(ciphertext[i]))  
            decryptedText += char(int(ciphertext[i] - shift - 97 + 26) % 26 + 97);  
        else  
            decryptedText += ciphertext[i];  
    }  
    return decryptedText;  
}
```

```
int main() {  
    int shiftKey;  
    string message;  
  
    cout << "Enter the message to encrypt: ";  
    getline(cin, message);  
  
    cout << "Enter the shift key: ";  
    cin >> shiftKey;  
  
    // Encryption part  
    string encryptedMessage = encrypt(message, shiftKey);  
    cout << "Encrypted message: " << encryptedMessage << endl;
```

21IT077

```
// Decryption part  
string decryptedMessage = decrypt(encryptedMessage, shiftKey);  
cout << "Decrypted message: " << decryptedMessage << endl;  
  
return 0;  
}
```

OUTPUT:

```
Enter the message to encrypt: survivalofthefittest  
Enter the shift key (a positive integer): 8  
Encrypted message: aczdqditwnbpmnqbbmab  
Decrypted message: survivalofthefittest
```

BRUTE FORCE ATTACK (ON SHIFT CIPHER)

```
#include <iostream>  
#include <string>  
using namespace std;  
  
string decrypt(string ciphertext, int shift) {  
    string decryptedText = "";  
    for (size_t i = 0; i < ciphertext.length(); i++) {  
        if (isupper(ciphertext[i]))  
            decryptedText += char(int(ciphertext[i] - shift - 65 + 26) % 26 + 65);  
        else if (islower(ciphertext[i]))  
            decryptedText += char(int(ciphertext[i] - shift - 97 + 26) % 26 + 97);  
        else  
            decryptedText += ciphertext[i];  
    }  
    return decryptedText;  
}
```

21IT077

```
int main() {  
    string encryptedMessage;  
  
    cout << "Enter the encrypted message: ";  
    getline(cin, encryptedMessage);  
  
    // Perform brute-force attack  
    cout << "Check the results:" << endl;  
    for (int shiftKey = 1; shiftKey <= 25; shiftKey++) {  
        string decryptedMessage = decrypt(encryptedMessage, shiftKey);  
        cout << "Key num " << shiftKey << ": " << decryptedMessage << endl;  
    }  
    return 0;  
}
```

OUTPUT:

```
Enter the encrypted message: ywzjxytkfq
Check the results:
Key num 1: xvyiwxsjep
Key num 2: wuxhvwridoo
Key num 3: vtwguvqhcnn
Key num 4: usvftupgbmm
Key num 5: truestofall
Key num 6: sqtdrsnezkk
Key num 7: rpsecrmdyjj
Key num 8: qorbpqlcxii
Key num 9: pnqaopkbwhh
Key num 10: ompznojavgg
Key num 11: nloymnizuff
Key num 12: mknxlmhytee
Key num 13: ljmwnlgxsdd
Key num 14: kilvjkfwrcc
Key num 15: jhkuijevqbb
Key num 16: igjthidupaa
Key num 17: hfisghctoza
Key num 18: gehrfgbsnyy
Key num 19: fdgqefarmxx
Key num 20: ecfpdezqlww
Key num 21: dbeocdypkvv
Key num 22: cadnbcxojuu
Key num 23: bzcmaabwnitt
Key num 24: ayblzavmhss
Key num 25: zxakyzulgrr
```

RESULT:

Thus, Caesar Cipher, Shift Cipher and Brute Force attack have been implemented accordingly.