# Information and Network Security
# 2CSDE54
# Practical 1

1. Perform encryption, decryption using the following substitution techniques
   a) Caeser cipher
   b) ROT-13

**CAESER CIPHER**

```
#include <bits/stdc++.h>
using namespace std;

string cc_encrypt(string text, int key)
{
    char c;
    for (int i = 0; i < text.length(); i++)
    {
        c = text[i];
        if (isalpha(c))
        {
            if (islower(c))
            {
                c = (c - 'a' + key) % 26 + 'a';
            }
            if (isupper(c))
            {
                c = (c - 'A' + key) % 26 + 'A';
            }
        }
        text[i] = c;
    }
    return text;
}
string cc_decrypt(string text, int key)
{
    char c;
    for (int i = 0; i < text.length(); i++)
    {

        c = text[i];
```

```
        if (isalpha(c))
        {
            if (islower(c))
            {
                c = (c - 'a' - key + 26) % 26 + 'a';
            }
            if (isupper(c))
            {
                c = (c - 'A' - key + 26) % 26 + 'A';
            }
        }
        text[i] = c;
    }
    return text;
}
int main()
{

    FILE *ptr;
    fstream f;
    char ch;
    string s = "";
    string cipher_text, plain_text;
    ptr = fopen("input.txt", "r");
    f.open("output.txt", fstream::out);
    if (NULL == ptr)
    {
        printf("Cannot open\n");
    }
    printf("Plain text: \n");
    while (!feof(ptr))
    {
        ch = fgetc(ptr);
        s += ch;
        printf("%c", ch);
    }
    cout << "\n";
    int key;
    cout << "Enter key: \n";

    cin >> key;
    cout << cc_encrypt(s, key);
```

```
    cipher_text = cc_encrypt(s, key);
    cout << "\nEncryption"
         << "\n"
         << "Cipher Text: " << cipher_text;
    for (int i = 0; i < cipher_text.length(); i++)
    {
        f << cipher_text[i];
    }
    plain_text = cc_decrypt(cipher_text, key);
    cout << "\nDecryption"
         << "\n"
         << "Plain Text: " << plain_text;
    f.close();
    fclose(ptr);
    return 0;
}
```

**O/P:**

```
Plain text:
abc
Enter key:
1
bcd
Encryption
Cipher Text: bcd
Decryption
Plain Text: abc
```

---

**ROT CIPHER**

```
#include <bits/stdc++.h>
using namespace std;
string cc_encrypt(string text, int key)
{
    char c;
    for (int i = 0; i < text.length(); i++)
    {
        c = text[i];
        if (isalpha(c))
        {
            if (islower(c))
```

```
            {
                c = (c - 'a' + key) % 26 + 'a';
            }
            if (isupper(c))
            {
                c = (c - 'A' + key) % 26 + 'A';
            }
        }
        text[i] = c;
    }
    return text;
}
string cc_decrypt(string text, int key)
{
    char c;
    for (int i = 0; i < text.length(); i++)
    {

        c = text[i];
        if (isalpha(c))
        {
            if (islower(c))
            {
                c = (c - 'a' - key + 26) % 26 + 'a';
            }
            if (isupper(c))
            {
                c = (c - 'A' - key + 26) % 26 + 'A';
            }
        }
        text[i] = c;
    }
    return text;
}
int main()
{

    FILE *ptr;
    fstream f;
    char ch;
    string s = "";
    string cipher_text, plain_text;
```

```
    ptr = fopen("input.txt", "r");
    f.open("output.txt", fstream::out);
    if (NULL == ptr)
    {
        printf("Cannot open\n");
    }
    printf("Plain text: \n");
    while (!feof(ptr))
    {
        ch = fgetc(ptr);
        s += ch;
        printf("%c", ch);
    }
    cout << "\n";
    int key;
    cout << "Enter key: \n";

    cin >> key;
    cout << cc_encrypt(s, key);
    cipher_text = cc_encrypt(s, key);
    cout << "\nEncryption"
         << "\n"
         << "Cipher Text: " << cipher_text;
    for (int i = 0; i < cipher_text.length(); i++)
    {
        f << cipher_text[i];
    }
    plain_text = cc_decrypt(cipher_text, key);
    cout << "\nDecryption"
         << "\n"
         << "Plain Text: " << plain_text;
    f.close();
    fclose(ptr);
    return 0;
}
```

**O/P:**

```
Plain text:
qhpx
Encryption
Cipher Text: duck
Decryption
Plain Text: qhpx
```

---

**Questions**

1. Crack the following plaintext TRVJRI TZGYVIJ RIV HLZKV VRJP KF
TIRTB
Ans: Plain text = “CAESAR CIPHERS ARE QUITE EASY TO CRACK” and Key = “R”. “V” and “R” are the most common letters in the ciphertext.

2. What key do we need to make “CAESAR” become “MKOCKB”?
Ans: Key = “K”

3. What key do we need to make “CIPHER” become “SYFXUH”?
Ans: Key = “Q”

4. Use the Caesar cipher to encrypt your first name
Ans: Considering Key = 1,
First Name: AZIM
Last Name: BAJN

5. How can we find the decryption key from the encryption key?
Ans: Encryption and decryption keys occur in pairs.