Vigenere Cipher

Vigenere Cipher using with different key values

Name : GANESH D

RegNo: 212223240035

AIM:

To develop a simple C program to implement Vigenere Cipher.

DESIGN STEPS:

Step 1:

Design of Vigenere Cipher algorithnm

Step 2:

Implementation using C or pyhton code

Step 3:

Testing algorithm with different key values. ALGORITHM DESCRIPTION: The Vigenere cipher is a method of encrypting alphabetic text by using a series of different Caesar ciphers based on the letters of a keyword. It is a simple form of polyalphabetic substitution.To encrypt, a table of alphabets can be used, termed a Vigenere square, or Vigenere table. It consists of the alphabet written out 26 times in different rows, each alphabet shifted cyclically to the left compared to the previous alphabet, corresponding to the 26 possible Caesar ciphers. At different points in the encryption process, the cipher uses a different alphabet from one of the rows used. The alphabet at each point depends on a repeating keyword.

PROGRAM:

```c
#include <stdio.h>
#include <string.h>
#include <ctype.h>

#define MAX_LENGTH 100

int main()
{
    char input[MAX_LENGTH];
    char key[MAX_LENGTH];
    char result[MAX_LENGTH];

    printf("Enter the text to encrypt: ");
    fgets(input, MAX_LENGTH, stdin);
    input[strcspn(input, "\n")] = '\0';

    printf("Enter the key: ");
    fgets(key, MAX_LENGTH, stdin);
    key[strcspn(key, "\n")] = '\0';

    int inputLength = strlen(input);
    int keyLength = strlen(key);

    for (int i = 0, j = 0; i < inputLength; ++i)
    {
```

```
        char currentChar = input[i];

        if (isalpha(currentChar))
        {
            int shift = toupper(key[j % keyLength]) - 'A';
            int base = isupper(currentChar) ? 'A' : 'a';

            result[i] = ((currentChar - base + shift + 26) % 26) + base;
            ++j;
        }
        else
        {
            result[i] = currentChar;
        }
    }

    result[inputLength] = '\0';
    printf("Encrypted text: %s\n", result);

    for (int i = 0, j = 0; i < inputLength; ++i)
    {
        char currentChar = result[i];

        if (isalpha(currentChar))
        {
            int shift = toupper(key[j % keyLength]) - 'A';
            int base = isupper(currentChar) ? 'A' : 'a';

            result[i] = ((currentChar - base - shift + 26) % 26) + base;
            ++j;
        }
    }

    result[inputLength] = '\0';
    printf("Decrypted text: %s\n", result);

    return 0;
}
```

OUTPUT:

```
Output

Enter the text to encrypt: Ganesh D
Enter the key: 12
Encrypted text: Qlxpcs N
Decrypted text: Ganesh D
```

RESULT:

The program is executed successfully