

EX. NO: 1(B)

IMPLEMENTATION OF PLAYFAIR CIPHER

AIM:

To write a C program to implement the Playfair Substitution technique.

DESCRIPTION:

The Playfair cipher starts with creating a key table. The key table is a 5×5 grid of letters that will act as the key for encrypting your plaintext. Each of the 25 letters must be unique and one letter of the alphabet is omitted from the table (as there are 25 spots and 26 letters in the alphabet).

To encrypt a message, one would break the message into digrams (groups of 2 letters) such that, for example, "HelloWorld" becomes "HE LL OW OR LD", and map them out on the key table. The two letters of the diagram are considered as the opposite corners of a rectangle in the key table. Note the relative position of the corners of this rectangle. Then apply the following 4 rules, in order, to each pair of letters in the plaintext:

1. If both letters are the same (or only one letter is left), add an "X" after the first letter
2. If the letters appear on the same row of your table, replace them with the letters to their immediate right respectively
3. If the letters appear on the same column of your table, replace them with the letters immediately below respectively
4. If the letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair.

EXAMPLE:

D. Playfair Cipher

Example1: Plaintext: CRYPTO IS TOO EASY **Key =** INFOSEC **Ciphertext: ??**

Grouped text: CR YP TO IS TO XO EA SY

Ciphertext: AQ TV YB NI YB YF CB OZ

I / J	N	F	O	S
E	C	A	B	D
G	H	K	L	M
P	Q	R	T	U
V	W	X	Y	Z

ALGORITHM:

STEP-1: Read the plain text from the user.

STEP-2: Read the keyword from the user.

STEP-3: Arrange the keyword without duplicates in a 5*5 matrix in the row order and fill the remaining cells with missed out letters in alphabetical order. Note that 'i' and 'j' takes the same cell.

STEP-4: Group the plain text in pairs and match the corresponding corner letters by forming a rectangular grid.

STEP-5: Display the obtained cipher text.

PROGRAM: (Playfair Cipher)

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<ctype.h>
#define MX 5
void playfair(char ch1,char ch2, char key[MX][MX])
{
    int i,j,w,x,y,z;
    FILE *out;
    if((out=fopen("cipher.txt","a+"))==NULL)
    {
        printf("File Corrupted.");
    }
    for(i=0;i<MX;i++)
    {
        for(j=0;j<MX;j++)
        {
            if(ch1==key[i][j])
            {
                w=i;
                x=j;
            }
            else if(ch2==key[i][j])
            {
                y=i;
                z=j;
            }
        }
    }

    //printf("%d%d %d%d",w,x,y,z);
    if(w==y)
    {
        x=(x+1)%5;z=(z+1)%5;
        printf("%c%c",key[w][x],key[y][z]);
        fprintf(out, "%c%c",key[w][x],key[y][z]);
    }
    else if(x==z)
    {

```

```

        w=(w+1)%5;y=(y+1)%5;
        printf("%c%c",key[w][x],key[y][z]);
        fprintf(out, "%c%c",key[w][x],key[y][z]);
    }
    else
    {
        printf("%c%c",key[w][z],key[y][x]);
        fprintf(out, "%c%c",key[w][z],key[y][x]);
    }

    fclose(out);
}
void main()
{
    int i,j,k=0,l,m=0,n;
    char key[MX][MX],keyminus[25],keyst[10],str[25]={0};
    char
    alpa[26]={'A','B','C','D','E','F','G','H','I','J','K','L',
    ,'M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'}
    ;
    clrscr();
    printf("\nEnter key:");
    gets(keyst);
    printf("\nEnter the plain text:");
    gets(str);
    n=strlen(keyst);
    //convert the characters to uppertext
    for (i=0; i<n; i++)
    {
        if(keyst[i]=='j')keyst[i]='i';
        else if(keyst[i]=='J')keyst[i]='I';
        keyst[i] = toupper(keyst[i]);
    }
    //convert all the characters of plaintext to uppertext
    for (i=0; i<strlen(str); i++)
    {
        if(str[i]=='j')str[i]='i';
        else if(str[i]=='J')str[i]='I';
        str[i] = toupper(str[i]);
    }
    j=0;

    for(i=0;i<26;i++)
    {
        for(k=0;k<n;k++)
        {
            if(keyst[k]==alpa[i])
            break;
            else if(alpa[i]=='J')
            break;
        }
        if(k==n)
        {
            keyminus[j]=alpa[i];j++;
        }
    }
}

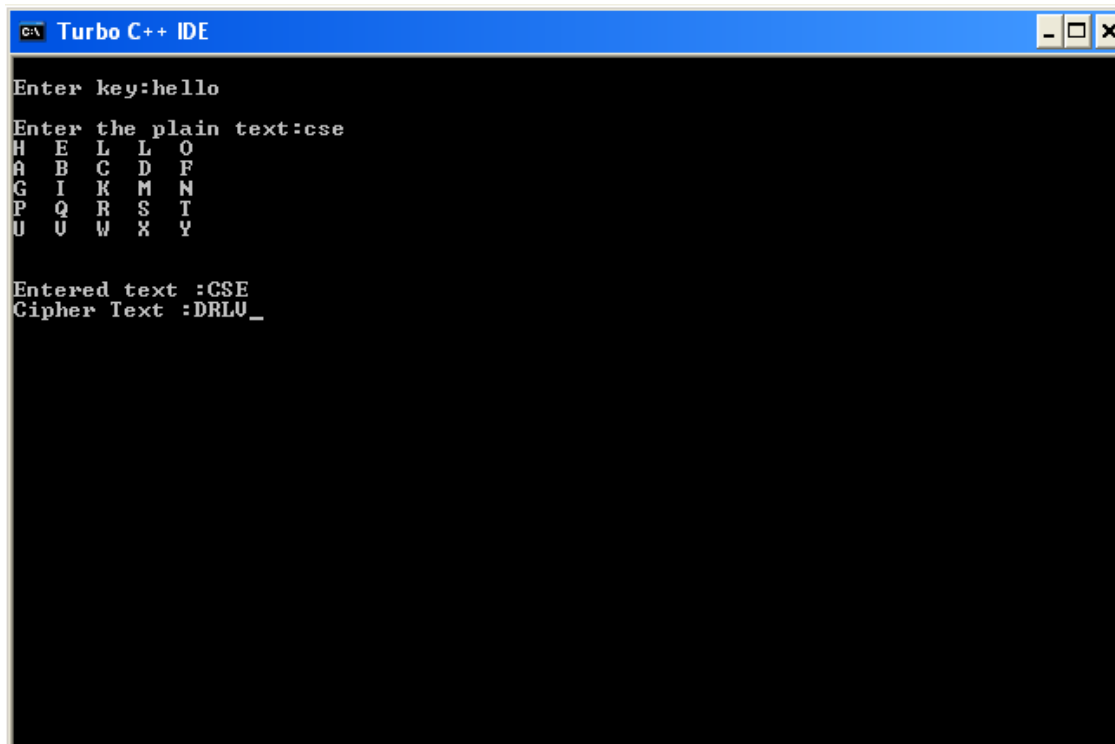
```

```

//construct key keymatrix
k=0;
for(i=0;i<MX;i++)
{
    for(j=0;j<MX;j++)
    {
        if(k<n)
        {
            key[i][j]=keystr[k];
            k++;}
        else
        {
            key[i][j]=keyminus[m];m++;
        }
        printf("%c  ",key[i][j]);
    }
    printf("\n");
}
printf("\n\nEntered text :%s\nCipher Text :",str);
for(i=0;i<strlen(str);i++)
{
    if(str[i]=='J')str[i]='I';
    if(str[i+1]=='\0')
    playfair(str[i],'X',key);
    else
    {
        if(str[i+1]=='J')str[i+1]='I';
        if(str[i]==str[i+1])
        playfair(str[i],'X',key);
        else
        {
            playfair(str[i],str[i+1],key);i++;
        }
    }
}
getch();
}

```

OUTPUT:



```
C:\ Turbo C++ IDE
Enter key:hello
Enter the plain text:cse
H E L L O
A B C D F
G I K M N
P Q R S T
U U W X Y

Entered text :CSE
Cipher Text :DRLU_
```

VIVA QUESTIONS:

1. What is difference between a monoalphabetic and a polyalphabetic cipher?
2. What are stream cipher and block cipher and how are they different?
3. How many possible keys does the playfair cipher have?
4. How to find the keyword of playfair cipher, given the plain text and cipher text?
5. Construct a table for the Playfair Cipher with the keyword EFFECTIVENESS?

RESULT:

Thus the Playfair cipher substitution technique had been implemented successfully.