Playfair Cipher - Implementation

Code

```
#include<stdio.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;
            }
        }
    }
    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 {
```

Dheeraj Lalwani
1902085

```
        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], keystr[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'};

    printf("\nEnter key: ");
    gets(keystr);

    printf("\nEnter the plain text: ");
    gets(str);

    n = strlen(keystr);

    //convert the characters of key to uppertext
    for (i = 0; i < n; i++) {
        if (keystr[i] == 'j') keystr[i] = 'i';
        else if (keystr[i] == 'J') keystr[i] = 'I';
        keystr[i] = toupper(keystr[i]);
    }

    //convert all the characters of plaintext message 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 (keystr[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("\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++;
            }
        }
    }
    printf("\n");
}
```

Output

```
C:\Users\USER\OneDrive\Desktop\practicals\CSS\CeasarCipher>gcc CaesarCipher.c -o CaesarCipher

C:\Users\USER\OneDrive\Desktop\practicals\CSS\CeasarCipher>CaesarCipher -e "Is your name Nikita?" 15
You chose to encrypt data with a shift of 15
Original string: | Is your name Nikita? |
Encrypted string: | X$ *~&# }p|t ]xzx%pN |

C:\Users\USER\OneDrive\Desktop\practicals\CSS\CeasarCipher>CaesarCipher -d "X$ *~&# }p|t ]xzx%pN" 15
You chose to decrypt data with a shift of 15
Original string: | X$ *~&# }p|t ]xzx%pN |
Decryped string: | Is your name Nikita? |
```

Dheeraj Lalwani
1902085