Practical no: 2

Name: Saloni Vishwakarma
Batch-Roll no: C1-13
Subject: Cryptography Lab

Aim: Perform encryption and decryption using One Time Pad Cipher technique.

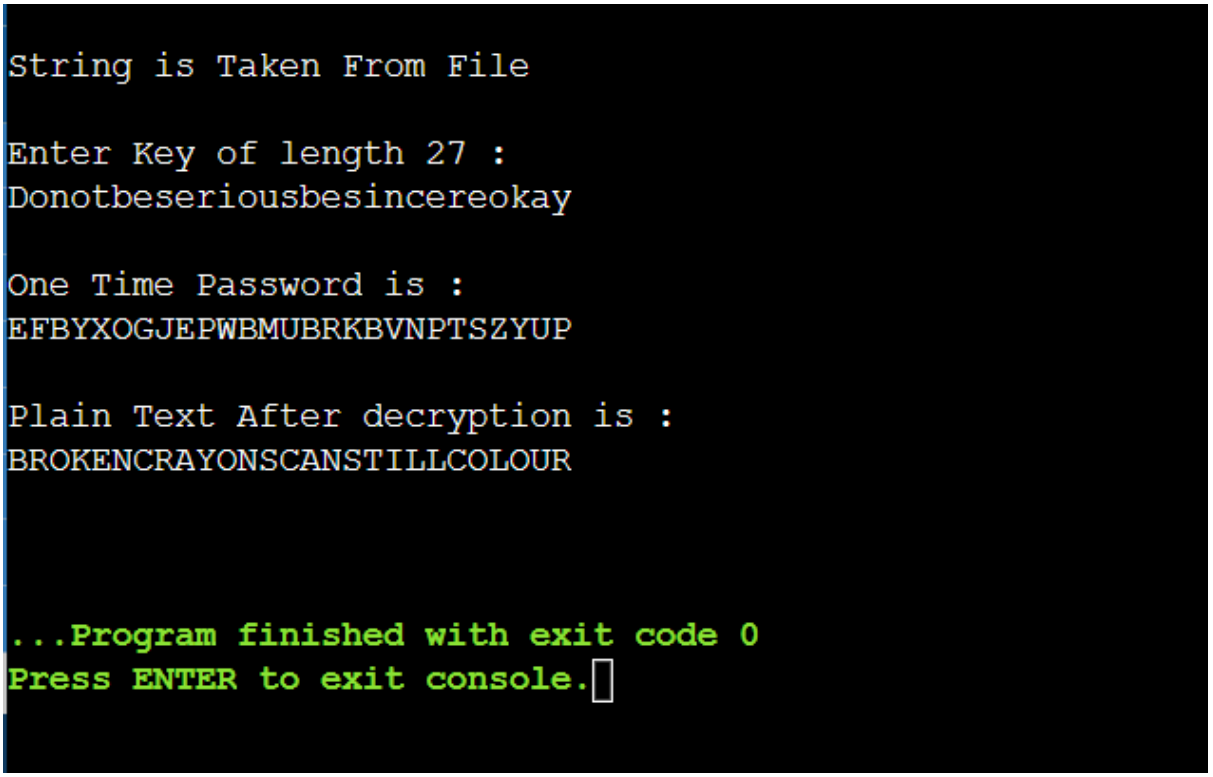
CODE:

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

#define MAX_LEN 100

int main() {
    char str[MAX_LEN], key[MAX_LEN], cipher[MAX_LEN], plaintext[MAX_LEN];
    int numstr[MAX_LEN], numkey[MAX_LEN], numcipher[MAX_LEN],
numtext[MAX_LEN];
    int i, j, count = 1;

    //Creating File And and reading data from it..
    FILE *myFile;
    myFile = fopen("mydoc.txt", "w");
    fprintf(myFile, "BrokenCrayonsCanStillColour");
    fclose(myFile);

    myFile = fopen("mydoc.txt", "r");
    fgets(str, MAX_LEN, myFile);
    fclose(myFile);

    //Removing spaces and making all char in uppercase for string
    for (i = 0, j = 0; i < strlen(str); i++) {
        if (str[i] != ' ') {
            str[j] = toupper(str[i]);
            j++;
        }
    }
    str[j] = '\0';
```

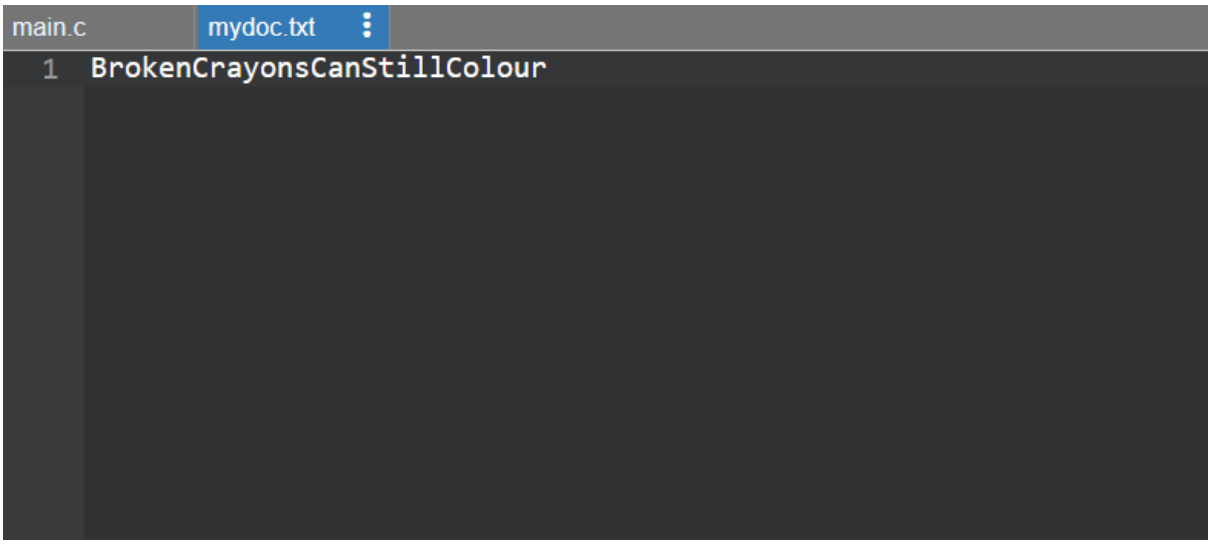
```
for (i = 0; i < strlen(str); i++) {
    numstr[i] = str[i] - 'A';
}

do {
    printf("\nString is Taken From File \n\n");
    printf("Enter Key of length %ld : \n", strlen(str));
    scanf("%s", key);

    if (strlen(key) == strlen(str)) {
        count = 0;
    }
} while (count);

//Removing spaces and making all char in uppercase for key
for (i = 0, j = 0; i < strlen(key); i++) {
    if (key[i] != ' ') {
        key[j] = toupper(key[i]);
        j++;
    }
}
key[j] = '\0';

for (i = 0; i < strlen(key); i++) {
    numkey[i] = key[i] - 'A';
}

//Adding both string numeric values
for (i = 0; i < strlen(str); i++) {
    numcipher[i] = numstr[i] + numkey[i];
}

//Converting the above 26 values in range of 25
for (i = 0; i < strlen(str); i++) {
    if (numcipher[i] > 25) {
        numcipher[i] = numcipher[i] - 26;
    }
}

//Printing cipher text
printf("\nOne Time Password is : \n");
for (i = 0; i < strlen(str); i++) {
    cipher[i] = numcipher[i] + 'A';
    printf("%c", cipher[i]);
```

```
    }

    //Decryption for cipher
    for (i = 0; i < strlen(str); i++) {
        numtext[i] = numcipher[i] - numkey[i];
    }

    for (i = 0; i < strlen(str); i++) {
        if (numtext[i] < 0) {
            numtext[i] = numtext[i] + 26;
        }
    }

    printf("\n\nPlain Text After decryption is : \n");
    for (i = 0; i < strlen(str); i++) {
        plaintext[i] = numtext[i] + 'A';
        printf("%c", plaintext[i]);
    }

    printf("\n\n");
    return 0;
}
```

OUTPUT:



Input File:



Output File: