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

void otp_cipher(char *plaintext, char *key, char *ciphertext) {
    int plaintext_len = strlen(plaintext);
    int key_len = strlen(key);
    int i;

    for (i = 0; i < plaintext_len; i++) {
        if (isalpha(plaintext[i])) {
            int base = isupper(plaintext[i]) ? 'A' : 'a';
            ciphertext[i] = ((plaintext[i] - base + (key[i] - base)) % 26) + base;
        } else {
            ciphertext[i] = plaintext[i];
        }
    }
    ciphertext[i] = '\0';
}

void otp_decipher(char *ciphertext, char *key, char *plaintext) {
    int ciphertext_len = strlen(ciphertext);
    int i;

    for (i = 0; i < ciphertext_len; i++) {
        if (isalpha(ciphertext[i])) {
            int base = isupper(ciphertext[i]) ? 'A' : 'a';
            plaintext[i] = ((ciphertext[i] - base - (key[i] - base) + 26) % 26) + base;
        } else {
            plaintext[i] = ciphertext[i];
        }
    }
    plaintext[i] = '\0';
}

int main() {
    char plaintext[100];
    char key[100];
    char ciphertext[100];
    char decryptedtext[100];

    printf("Enter the plaintext: ");
    scanf("%99s", plaintext);

    srand(time(NULL));
```

```
    for (int i = 0; i < strlen(plaintext); i++) {
        if (isalpha(plaintext[i])) {
            int base = isupper(plaintext[i]) ? 'A' : 'a';
            key[i] = (rand() % 26) + base;
        } else {
            key[i] = plaintext[i];
        }
    }
    key[strlen(plaintext)] = '\0';

    otp_cipher(plaintext, key, ciphertext);
    printf("Ciphertext: %s\n", ciphertext);
    printf("Key: %s\n", key);

    otp_decipher(ciphertext, key, decryptedtext);
    printf("Decrypted text: %s\n", decryptedtext);

    return 0;
}
```