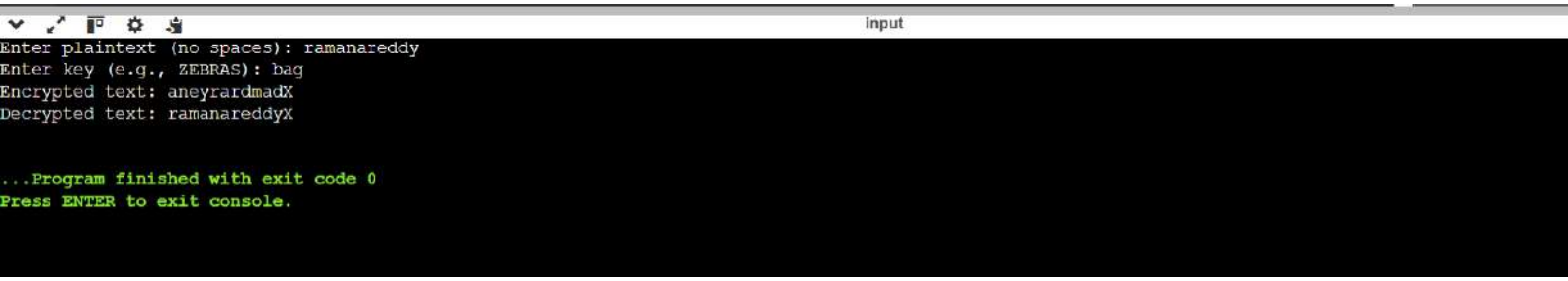
```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
void getColumnOrder(char *key, int *order) {
    int len = strlen(key);
    char tempKey[20];
    strcpy(tempKey, key);
    for (int i = 0; i < len; i++) {
        order[i] = i;
    }
    for (int i = 0; i < len - 1; i++) {
        for (int j = i + 1; j < len; j++) {
            if (tempKey[j] < tempKey[i]) {
                char c = tempKey[i];
                tempKey[i] = tempKey[j];
                tempKey[j] = c;
                int t = order[i];
                order[i] = order[j];
                order[j] = t;
            }
        }
    }
}
void encrypt(char *plaintext, char *key, char *ciphertext) {
    int keyLen = strlen(key);
    int textLen = strlen(plaintext);
    int rows = (textLen + keyLen - 1) / keyLen;
    char matrix[rows][keyLen];
    int k = 0;
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < keyLen; j++)
            matrix[i][j] = (k < textLen) ? plaintext[k++] : 'X';

    int order[keyLen];
    getColumnOrder(key, order);
    k = 0;
    for (int i = 0; i < keyLen; i++) {
        int col = order[i];
        for (int j = 0; j < rows; j++) {
            ciphertext[k++] = matrix[j][col];
        }
    }
```

```
    }
    ciphertext[k] = '\0';
}
void decrypt(char *ciphertext, char *key, char *plaintext) {
    int keyLen = strlen(key);
    int textLen = strlen(ciphertext);
    int rows = (textLen + keyLen - 1) / keyLen;

    char matrix[rows][keyLen];
    int order[keyLen];
    getColumnOrder(key, order);
    int k = 0;
    for (int i = 0; i < keyLen; i++) {
        int col = order[i];
        for (int j = 0; j < rows; j++) {
            matrix[j][col] = ciphertext[k++];
        }
    }
    k = 0;
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < keyLen; j++)
            plaintext[k++] = matrix[i][j];
    plaintext[textLen] = '\0';
}
int main() {
    char plaintext[100], ciphertext[100], decrypted[100], key[20];
    printf("Enter plaintext (no spaces): ");
    scanf("%s", plaintext);
    printf("Enter key (e.g., ZEBRAS): ");
    scanf("%s", key);
    encrypt(plaintext, key, ciphertext);
    printf("Encrypted text: %s\n", ciphertext)
    decrypt(ciphertext, key, decrypted);
    printf("Decrypted text: %s\n", decrypted);
    return 0;
}
```


input
Enter plaintext (no spaces): ramanareddy
Enter key (e.g., ZEBRAS): bag
Encrypted text: aneyrardmadX
Decrypted text: ramanareddyX
...Program finished with exit code 0
Press ENTER to exit console.