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

void railFenceEncrypt(char text[], int rails) {
    int len = strlen(text), i, j, dir = 1, row = 0;
    char fence[rails][len];
    memset(fence, '\n', sizeof(fence));

    for (i = 0; i < len; i++) {
        fence[row][i] = text[i];
        row += dir;
        if (row == 0 || row == rails - 1) dir = -dir;
    }

    for (i = 0; i < rails; i++)
        for (j = 0; j < len; j++)
            if (fence[i][j] != '\n')
                printf("%c", fence[i][j]);
    printf("\n");
}

void railFenceDecrypt(char cipher[], int rails) {
    int len = strlen(cipher), i, j, dir = 1, row = 0, k = 0;
    char fence[rails][len];
    memset(fence, '\n', sizeof(fence));

    for (i = 0; i < len; i++) {
        fence[row][i] = '*';
        row += dir;
        if (row == 0 || row == rails - 1) dir = -dir;
    }

    for (i = 0; i < rails; i++)
        for (j = 0; j < len; j++)
            if (fence[i][j] == '*')
                fence[i][j] = cipher[k++];

    row = 0; dir = 1;
    for (i = 0; i < len; i++) {
        printf("%c", fence[row][i]);
        row += dir;
        if (row == 0 || row == rails - 1) dir = -dir;
    }
    printf("\n");
}

int main() {
    char text[256];
    int rails;
    printf("Enter text: ");
    gets(text);
    printf("Enter number of rails: ");
    scanf("%d", &rails);

    printf("Encrypted: ");
    railFenceEncrypt(text, rails);

    char cipher[256];
    printf("Enter cipher text to decrypt: ");
    scanf("%s", cipher);

    printf("Decrypted: ");
    railFenceDecrypt(cipher, rails);

    return 0;
}
```