C：

#include <stdio.h>

#include <stdlib.h>

int turn(char a[], char b[])

{

int i;

int n = 0;

for(i = 0;a[i]!='\0';i++)

{

if(a[i] == b[i])

{

continue;

}

else

{

b[i+1]=(b[i+1]=='\*'?'o':'\*');

n++;

}

}

return n;

}

int main(void)

{

char a[1000];

char b[1000];

gets(a);

gets(b);

printf("%d\n", turn(a, b));

return 0;

}

C++：

#include <iostream>

using namespace std;

char reverse(char c)

{

if(c=='\*')

c='o';

else

c='\*';

return c;

}

int main()

{

char a[1000],b[1000];

cin>>a>>b;

int move=0;

for(int i=0;a[i]!='\0';i++)

{

if(a[i]!=b[i])

{

a[i]=reverse(a[i]);

a[i+1]=reverse(a[i+1]);

move++;

}

}

cout<<move<<endl;

return 0;

}

Java：

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Main {

public static int cnt = 0;

public static void reverse(char[] origin, int k) {

origin[k] = (origin[k] == '\*') ? 'o' : '\*';

origin[k+1] = (origin[k+1] == '\*') ? 'o' : '\*';

++cnt;

}

public static void main(String[] args) throws IOException {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

char[] origin = br.readLine().toCharArray();

char[] target = br.readLine().toCharArray();

int k = 0;

while(k < target.length)

if(k+1 < target.length) {

if(origin[k] != target[k] && origin[k+1] == target[k+1]) {

reverse(origin, k);

k = k+1;

} else if(origin[k] != target[k] && origin[k+1] != target[k+1]) {

reverse(origin, k);

k = k+2;

} else if(origin[k] == target[k] && origin[k+1] == target[k+1])

k = k+2;

else

k = k+1;

} else

k = k+1;

System.out.println(cnt);

}

}