11.

标题：统计回文

“回文串”是一个正读和反读都一样的字符串，比如“level”或者“noon”等等就是回文串。花花非常喜欢这种拥有对称美的回文串，生日的时候她得到两个礼物分别是字符串A和字符串B。现在她非常好奇有没有办法将字符串B插入字符串A使产生的字符串是一个回文串。你接受花花的请求，帮助她寻找有多少种插入办法可以使新串是一个回文串。如果字符串B插入的位置不同就考虑为不一样的办法。

例如：

A = “aba”，B = “b”。这里有4种把B插入A的办法：

\* 在A的第一个字母之前: "baba" 不是回文

\* 在第一个字母‘a’之后: "abba" 是回文

\* 在字母‘b’之后: "abba" 是回文

\* 在第二个字母'a'之后 "abab" 不是回文

所以满足条件的答案为2

输入描述：

每组输入数据共两行。

第一行为字符串A

第二行为字符串B

字符串长度均小于100且只包含小写字母

输出描述：

输出一个数字，表示把字符串B插入字符串A之后构成一个回文串的方法数

示例1:

输入

aba

b

输出

2

import java.util.\*;

public class Main {

public static boolean isHuiwen(String s){

int i = 0;

int j = s.length()-1;

while(i<j){

if(s.charAt(i)!=s.charAt(j)){

return false;

}

i++;

j--;

}

return true;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

String str1 = sc.nextLine();

String str2 = sc.nextLine();

int count = 0;

for(int i = 0; i <= str1.length();i++){

StringBuilder sb = new StringBuilder(str1);

sb.insert(i, str2);

if(isHuiwen(sb.toString())){

count++;

}

}

System.out.println(count);

}