人随着岁数的增长是越大越聪明还是越大越笨，这是一个值得全世界科学家思考的问题,同样的问题Eddy也一直在思考，因为他在很小的时候就知道亲和串如何判断了，但是发现，现在长大了却不知道怎么去判断亲和串了，于是他只好又再一次来请教聪明且乐于助人的你来解决这个问题。   
 亲和串的定义是这样的：给定两个字符串s1和s2,如果能通过s1循环移位，使s2包含在s1中，那么我们就说s2 是s1的亲和串。

**Input**

本题有多组测试数据，每组数据的第一行包含输入字符串s1,第二行包含输入字符串s2，s1与s2的长度均小于100000。

**Output**

如果s2是s1的亲和串，则输出"yes"，反之，输出"no"。每组测试的输出占一行。

**Sample Input**

AABCD

CDAA

ASD

ASDF

**Sample Output**

yes

no

多个字符串函数应用的典型例题

#include<bits/stdc++.h>

#define maxn 100005

using namespace std;

int main()

{

//freopen("input.txt","r",stdin);

char a[maxn],b[maxn];

char c[maxn\*2];

while(cin>>a>>b)

{

strcpy(c,a);

strcat(c,a);

if(strstr(c,b)!=0)

cout<<"yes\n";

else

cout<<"no\n";

}

return 0;

}

在字符串s1后再接上s1,形成新的字符串,再在s1中寻找是否有s2的子串,若有则为亲和串.

**用到的函数原型:**

**1.strcat() 连接两个字符串**

   #include <string.h>  
   char \*strcat( char \*str1, const char \*str2 );  
   功能：函数将字符串str2 连接到str1的末端，并返回指针str1. 例如：  
    printf( "Enter your name: " );  
    scanf( "%s", name );  
    title = strcat( name, " the Great" );  
    printf( "Hello, %s\n", title );

**2.strcpy() 拷贝字符串**

 #include <string.h>  
 char \*strcpy( char \*to, const char \*from );  
 功能：复制字符串from 中的字符到字符串to，包括空值结束符。返回值为指针to。

**3.strstr() 在一字符串中查找指定的子串首次出现的位置**

#include <string.h>  
char \*strstr( const char \*str1, const char \*str2 );  
功能：函数返回一个指针，它指向字符串str2 首次出现于字符串str1中的位置，如果没有找到，返回NULL。

用string

#include<iostream>

#include<stdio.h>

#include<cstring>

using namespace std;

int main()

{

string a,b,c="";

while(cin>>a>>b)

{

c=a;

c+=a;

if((c.find(b)+1)!=0)cout<<"yes\n";else cout<<"no\n";

}

return 0;

}