#include <bits/stdc++.h>

using namespace std;

int Next[10000];

char t[10000]; //主串

char p[10000]; //模式串

int lenp,lent;

void get\_next()

{

int i,j;

Next[0] = j = -1;

i = 0;

while(i<lenp) //最后一位的判断其实是多余的

{

while(j!=-1&&p[j]!=p[i])

j = Next[j];

Next[++i] = ++j;

}

}

int kmp1() //在t串找p串 返回下标

{

int i,j;

i = j = 0; //两个下标指针 i为主串的指针 j为模式串的指针

while(i<lent&&j<lenp)

{

while(j!=-1&&t[i]!=p[j])

j = Next[j];

i++;

j++;

}

if(j==lenp)

return i-j; //若找到返回开始下标（从0开始）

return -1; //找不到返回-1

}

int kmp2() //返回匹配次数

{

int i,j;

i = j = 0;

int ans = 0;

while(i<lent)

{

while(j!=-1&&t[i]!=p[j])

j = Next[j];

if(j==lenp-1)

{

ans++;

j = Next[j];

}

i++;

j++;

}

return ans;

}

int kmp3() //返回t串中有多少个p串

{

int i,j;

i = j = 0;

int ans = 0;

while(i<lent)

{

while(j!=-1&&t[i]!=p[j])

j = Next[j];

if(j==lenp-1)

{

ans++;

j = -1; // j指针改变为-1 然后++ 从0重新查找

}

i++;

j++;

}

return ans;

}

int main()

{

scanf("%s",t);

scanf("%s",p);

lenp = strlen(p);

lent = strlen(t);

get\_next();

int ans = kmp3();

cout<<ans<<endl;

return 0;

}