**周期字串**

问题描述

　　右右喜欢听故事，但是右右的妈妈总是讲一些“从前有座山，山里有座庙，庙里有个老和尚给小和尚讲故事，讲的什么呢？从前有座山……”这样循环的故事来搪塞右右。  
　　我们定义，如果一个字符串是以一个或者一个以上的长度为k的重复字符串所连接成的，那么这个字符串就叫做周期为k的串。  
　　例如:  
　　字符串’abcabcabcabc’周期为3，因为它是由4个循环’abc’组成的。它同样是以6为周期（两个重复的’abcabc’）和以12为周期（一个循环’abcabcabcabc’）。  
　　右右现在想给他的朋友大灰狼转述妈妈讲的故事，请帮他写一个程序，可以测定一个字符串的最小周期。

输入格式

　　一个最大长度为100的无空格的字符串。

输出格式

　　一个整数，表示输入的字符串的最小周期。

样例输入

HaHaHa

样例输出

2

样例输入

Return0

样例输出

7

C++代码：

#include <iostream>

#include <string.h>

using namespace std;

int count;

char a[100];

void f(int n,int k)

{

int i,j,zhouqi,c,fla=0;

for(i=1;i<=n;i++)

{

if(n%i==0)

{

for(j=0;j<n/i;j++)

{

for(c=0;c<i;c++)

{

if(a[c]!=a[j\*i+c])

{

fla=1;

break;

}

}

if(fla)

break;

}

if(fla==0)

{

count=i;

return ;

}

}

fla=0;

}

}

int main()

{

int n;

cin>>a;

f(strlen(a),1);

cout<<count<<endl;

return 0;

}

C代码：

#include<stdio.h>

#include<string.h>

int min=100;

int a[28];

struct num{

int a[4];

};

int prime(int n)

{

int i;

if(n==7)return 1;

for(i=3;i<n/2;i++)

{

if(n%i==0)break;

}

if(i==n/2)return 1;

return 0;

}

void prime1()

{

a[0]=2;

a[1]=3;

a[2]=5;

int i,j=3;

for(i=7;i<110;i=i+2)

{

if(prime(i))

{

a[j]=i;

j++;

}

}

}

struct num prime2(int n)

{

struct num p;

int i,j;

for(i=0;i<4;i++)

p.a[i]=0;

j=0;

for(i=0;i<28;i++)

{

if(n%a[i]==0)

p.a[j]=a[i];

j++;

}

return p;

}

int xunhuan(char a[],int n)

{

int i,m,l,j,k;

int cnt=1;

struct num p;

if(n==1)

return 1;

p=prime2(n);

for(i=0;i<4;i++)

{

if(p.a[i]!=0)

{

l=n/p.a[i];

for(j=1;j<p.a[i];j++)

for(k=0;k<l;k++)

if(a[k]!=a[j\*l+k])

{

cnt=0;

break;

}

if(cnt==1)

{

l=xunhuan(a,l);

if(l<min)

min=l;

}

cnt=1;///////////////

}

else if(min>n)

min=n;

}//for

return min;

}

int main()

{

char a[120];

int m,n;

int l;

prime1();

gets(a);

n=strlen(a);

m=xunhuan(a,n);

printf("%d",m);

return 0;

}

JAVA代码：

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

String s=new String();

int k = 0,m=0;

Scanner sc=new Scanner(System.in);

s=sc.nextLine();

for (int i = 1; i <= s.length(); i++) {

String str=s.substring(0, i);

for (int j = 0; j < s.length(); j+=i,k=j,m=i) {

if(j+i>s.length()) break;

if(!str.equals(s.substring(j, j+i))){

break;

}

str=s.substring(j, j+i);

}

if(k>=s.length()){

System.out.println(m);

break;

}

}

}

}