**Period**

**Time Limit: 2000/1000 MS (Java/Others)    Memory Limit: 65536/32768 K (Java/Others)  
Total Submission(s): 5842    Accepted Submission(s): 2832**

Problem Description

For each prefix of a given string S with N characters (each character has an ASCII code between 97 and 126, inclusive), we want to know whether the prefix is a periodic string. That is, for each i (2 <= i <= N) we want to know the largest K > 1 (if there is one) such that the prefix of S with length i can be written as AK , that is A concatenated K times, for some string A. Of course, we also want to know the period K.

Input

The input file consists of several test cases. Each test case consists of two lines. The first one contains N (2 <= N <= 1 000 000) – the size of the string S. The second line contains the string S. The input file ends with a line, having the number zero on it.

Output

For each test case, output “Test case #” and the consecutive test case number on a single line; then, for each prefix with length i that has a period K > 1, output the prefix size i and the period K separated by a single space; the prefix sizes must be in increasing order. Print a blank line after each test case.

Sample Input

3

aaa

12

aabaabaabaab

0

Sample Output

Test case #1

2 2

3 3

Test case #2

2 2

6 2

9 3

12 4

分析：给你一个字符串，让你找它的每一个前缀是否可以由某个循环节构成 可以的话 就输出前缀的长度以及循环节循环的次数

由于k大于1 所以循环节不会是前缀本身

AC代码：

#include <stdio.h>

#include <string.h>

#define MOD 10007

**char** a**[**2000005**],**b**[**2000005**],**c**[**2000005**];**

**int** next**[**2000005**];**

**int** n**,**m**,**sum**;**

**void** Next**()** ///求next数组

**{**

next**[**0**] =** next**[**1**] =** 0**;**

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

**{**

**int** j **=** next**[**i**];**

**while(**j**&&**a**[**j**]!=**a**[**i**])**

j **=** next**[**j**];**

next**[**i**+**1**] =** a**[**i**]==**a**[**j**]?**j**+**1**:**0**;**

**}**

**}**

**int main()**

**{**

**int** k **=** 1**;**

**while(~**scanf**(**"%d"**,&**n**))**

**{**

**if(**n**==**0**)**

**break;**

scanf**(**"%s"**,**a**);**

Next**();**

//find1();

printf**(**"Test case #%d\n"**,**k**++);**

**for(int** i **=** 2**;** i **<=** n**;** i**++)**

**{**

**int** len **=** i**;**

**int** cir **=** len **-** next**[**len**];**

**if(**cir**==**1**)**

printf**(**"%d %d\n"**,**len**,**len**);** ///输出当前前缀的长度以及循环节的次数

**else if(**cir**!=**len**&&**len**%**cir**==**0**)**

printf**(**"%d %d\n"**,**len**,**len**/**cir**);**

**else**

**continue;**

**}**

printf**(**"\n"**);**

**}**

**return** 0**;**

**}**