

# 北方工业大学

## 2011 年第七届程序设计大赛试题

### 1、可爱的甲虫

#### Description

有  $n$  只甲虫 ( $0 < n < 10000$ ), 依次编号为 1,2,3, ...,  $n$ , 开始时头都朝东。第一秒钟, 编号为 1 的倍数的甲虫向右转 90 度; 第二秒钟, 编号为 2 的倍数的甲虫向右转 90 度; 第三秒钟, 编号为 3 的倍数的甲虫向右转 90 度, ..., 如此进行。那么,  $n$  秒钟后, 第  $n$  号甲虫头朝哪个方向?

#### Input

输入  $m$  个数 ( $0 < m < 10$ ), 表示有  $m$  个测试数据, 以下输入  $m$  个甲虫的编号  $n_1$ 、 $n_2$ 、...、 $n_m$ 。

#### Output

输出共  $m$  行, 分别表示编号为  $n_1$ 、 $n_2$ 、...、 $n_m$  的甲虫在第  $n_1$ 、 $n_2$ 、...、 $n_m$  秒后头的朝向, 用 East、West、South、North 来表示。

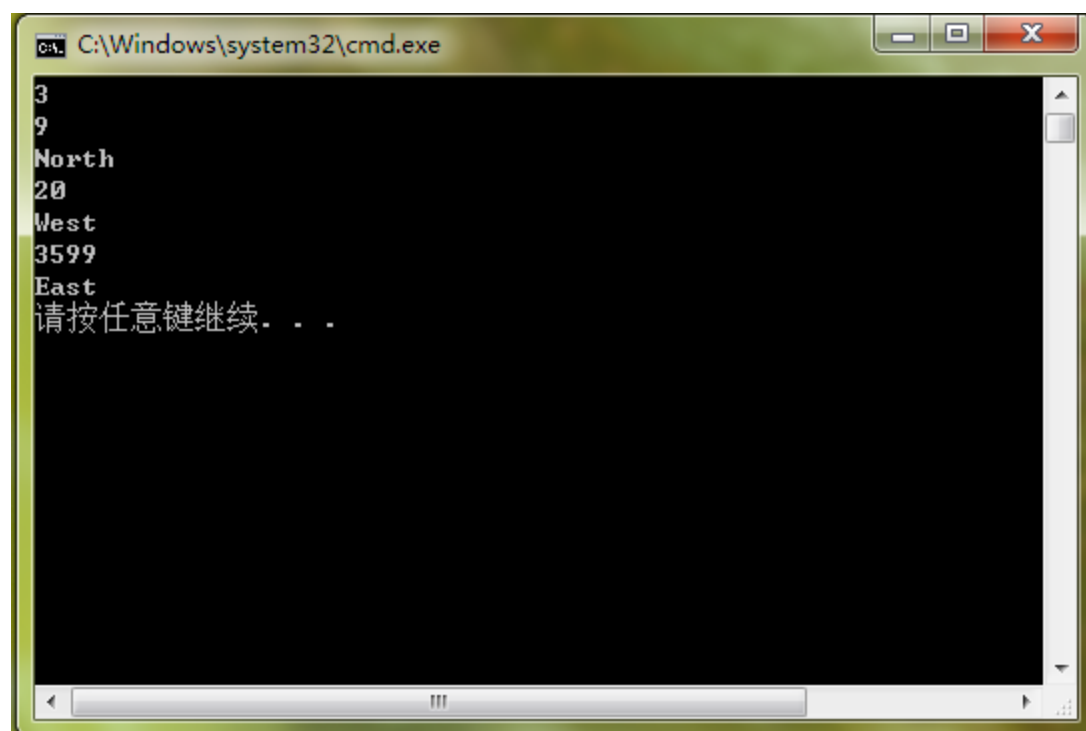
#### Sample Input

```
3
9 20 3599
```

#### Sample Output

```
North
West
East
```

附实际测试截图:



## 2、水果开会时段

### Description

每个百度工程师团队都有一笔还算丰裕的食品经费，足够每天购置多种水果。水果往往下午送达公司前台。前台的姐姐们只要看到同时出现五种或以上的水果，就称之为“水果开会”。

从搜索引擎切词的语法角度，只要两种水果的名字中有一个字相同就属于同样的类别。例如“小雪梨”和“大雪梨”是同一种水果，而“核桃”和“水蜜桃”也被认为是同一种水果。尤其要指出的是，如果有三种水果  $x$ ,  $y$ ,  $z$  同时在前台出现，且  $x$  和  $y$  是同一种水果， $y$  和  $z$  也是同一种水果的时候， $x$  和  $z$  在此时也被认为是同一种水果（即使  $x$  和  $z$  并不包含相同的字）。现在前台的姐姐们想知道，今天是否有“水果开会”——五种或更多的水果同时在前台出现。

### Input

输入的第一行只有一个整数  $n$ ，表示购置水果的组数。接下来的  $n$  行表示水果的到达时间、取走时间（时间用 1200 到 1900 之间的正整数表示，保证取走时间大于到达时间）。剩下的字符串以空格分割每一种水果。如“1400 1600 雪梨 水蜜桃”，表示下午两点到四点（包含两点和四点这两个时间点），雪梨和水蜜桃会在前台等待开会。每种水果名称由不超过十个汉字组成。

### Output

输出仅一行，包含一个字符串 Yes 或 No，分别表示今天水果开会与否。

### Sample Input

```
3
1200 1400 雪梨 柠檬
1300 1400 西瓜 苹果
1400 1800 花生 水蜜桃
```

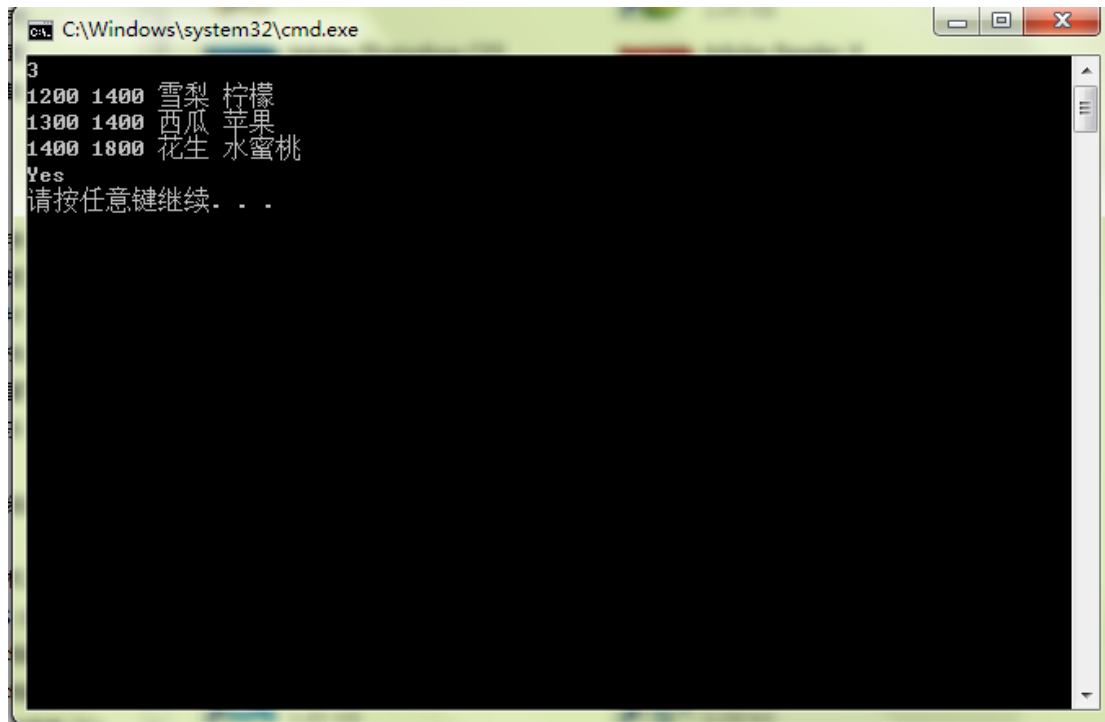
### Sample Output

Yes

### 样例解释

在样例中，时刻 1400 有六种水果在前台。

附实际测试截图：



### 3.简单的语音计算器

#### Description

Suzumiya 最近开始无端刁难她的同学 ViVo，总是莫名其妙的问他一些简单而又离谱、没有实际意义的数学问题，比如三千加上五百等于多少。回答一次两次还可以，但不断这样的纠缠致使 ViVo 已经无法忍受了。所以 ViVo 决定制作一个语音装置来自动回答 Suzumiya 提出的无聊问题。

ViVo 知道你是个伟大的算法艺术家，所以希望该装置核心的数学计算模块你能来帮忙。装置接收到的语音会自动为你转化为对应的中文字符串，经过你的模块处理完成后输出中文字符串，传递给装置来朗读出来。

为了给你带来方便，ViVo 已经提前收集好了 Suzumiya 可能会问到的问题，发现这些问题都是数学中的加减乘除问题，并且题目中可能包含的数字都不超过 10，答案都可以用数字来表示。

#### Input

输入的第一行是一个数字  $n$ ，表示接下来有  $n$  个问题，每个问题占一行。如果遇到一行运算非法（如除数为 0）则显示信息：输入非法！如果运算结果为负数则不用计算出结果，只显示信息：结果为负数！

#### Output

每行包含一个没有语病的中文表示最终的结果。

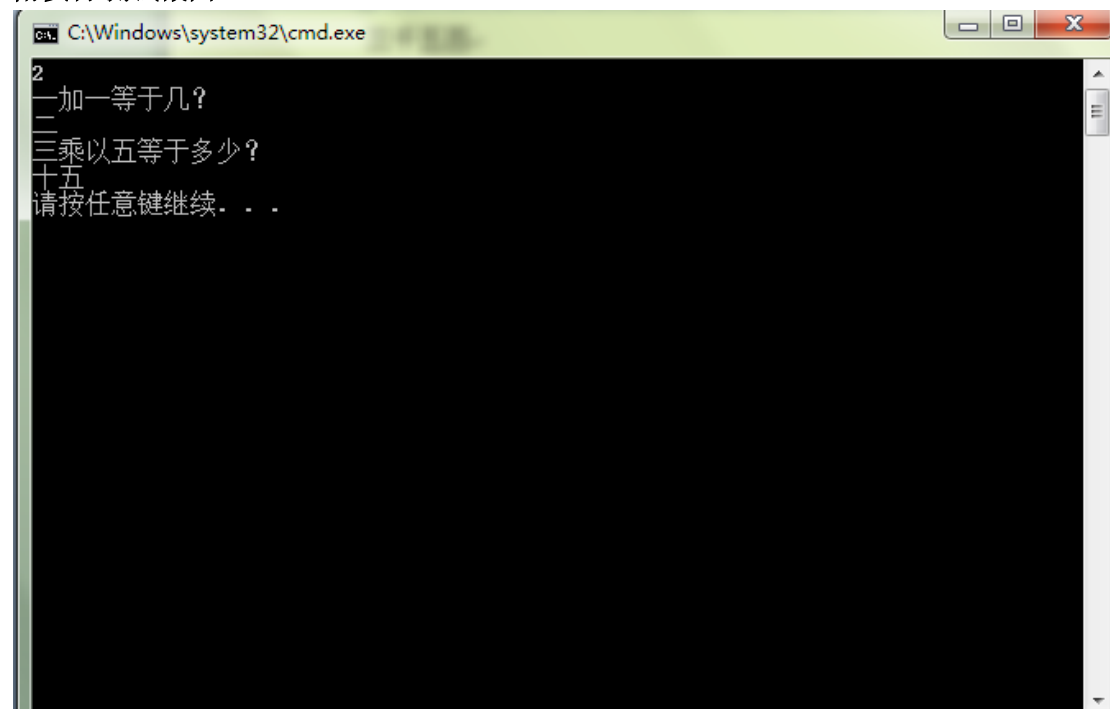
### Sample Input

2  
一加一等于几？  
三乘以五等于多少？

### Sample Output

二  
十五

附实际测试截图：



## 4、角谷猜想

### Description

考拉兹猜想，又称为  $3n+1$  猜想、角谷猜想、哈塞猜想、乌拉姆猜想或叙拉古猜想，是由日本数学家角谷静夫发现，是指对于每一个正整数，如果它是奇数，则对它乘 3 再加 1，如果它是偶数，则对它除以 2，如此循环，最终都能够得到 1。取一个数字。

如  $n = 6$ ，根据上述公式，得出  $6 \rightarrow 3 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ 。（步骤中最大的数是 16，共有 7 个步骤）。

如  $n = 11$ ，根据上述公式，得出

$11 \rightarrow 34 \rightarrow 17 \rightarrow 52 \rightarrow 26 \rightarrow 13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ 。（步骤中最大的数是 52，共有 13 个步骤）

如  $n = 27$ ，根据上述公式，得出：

$27 \rightarrow 82 \rightarrow 41 \rightarrow 124 \rightarrow 62 \rightarrow 31 \rightarrow 94 \rightarrow 47 \rightarrow 142 \rightarrow 71 \rightarrow 214 \rightarrow 107 \rightarrow 322 \rightarrow 161 \rightarrow 484 \rightarrow 242 \rightarrow 121 \rightarrow 364 \rightarrow 182 \rightarrow 91 \rightarrow 274 \rightarrow 137 \rightarrow 412 \rightarrow 206 \rightarrow 103 \rightarrow 310 \rightarrow 155 \rightarrow 466 \rightarrow 233 \rightarrow 70$

0→350→175→526→263→790→395→1186→593→1780→890→445→1336→668  
→334→167→502→251→754→377→1132→566→283→850→425→1276→638→  
319→958→479→1438→719→2158→1079→3238→1619→4858→2429→7288→3644  
→1822→911→2734→1367→4102→2051→6154→3077→9232→4616→2308→11  
54→577→1732→866→433→1300→650→325→976→488→244→122→61→184  
→92→46→23→70→35→106→53→160→80→40→20→10→5→16→8→4→2→1  
。(步骤中最大的数是 9232，共有 111 个步骤)

考拉兹猜想称，任何正整数，经过上述计算步骤后，最终都会得到 1。

#### Input

输入仅包含一个需要验证角谷猜想的数  $n$ ， $0 < n < 100$ 。

#### Output

从数字  $n$  开始，依次输出经过变换后得到的每一个数字，数与数之间用“->”连接，直到输出 1 为止步。统计步骤中最大的数字  $m$  以及所需的步骤数  $x$ ，然后在下一行输出如下信息：(步骤中最大的数是  $m$ ，共有  $x$  个步骤)

#### Sample Input

11

#### Sample Output

11->34->17->52->26->13->40->20->10->5->16->8->4->2->1

(步骤中最大的数是 52，共有 13 个步骤)

附实际测试截图：

```
C:\Windows\system32\cmd.exe
11
11->34->17->52->26->13->40->20->10->5->16->8->4->2->1
<步骤中最大的数是52，共有13个步骤>
请按任意键继续...
```

## 5、打印自方幂数

#### Description

一个  $n$  位正整数如果等于它的  $n$  个数字的  $n$  次方和，该数称为  $n$  位自方幂数，也称为  $n$  位  $n$  次幂回归数。

一位自方幂数又叫花数。

两位自方幂数又叫花朵数。

三位自方幂数又叫水仙花数。

四位自方幂数又叫玫瑰花数。

五位自方幂数又叫五角星数。

六位自方幂数又叫六合数。

叫法很多，实在不知道怎么称呼。此外还有广义的定义。

维基百科上是统称为水仙花数 Narcissistic number, PPDI, 阿姆斯特朗数

Armstrong\_number。(sequence A005188 in OEIS)

现在要求你打印出 1 到 6 位的自方幂数。

#### Input

首先输入 m，表示有 m 个测试数据。接下来输入 m 个数字，分别表示要打印的 m 个自方幂数的位数。

#### Output

输出这 m 个测试数据的所有自方幂数（如果不存在则显示信息：不存在），中间用换行隔开；n 相邻两个数用空格隔开（见附图）。

#### Sample Input

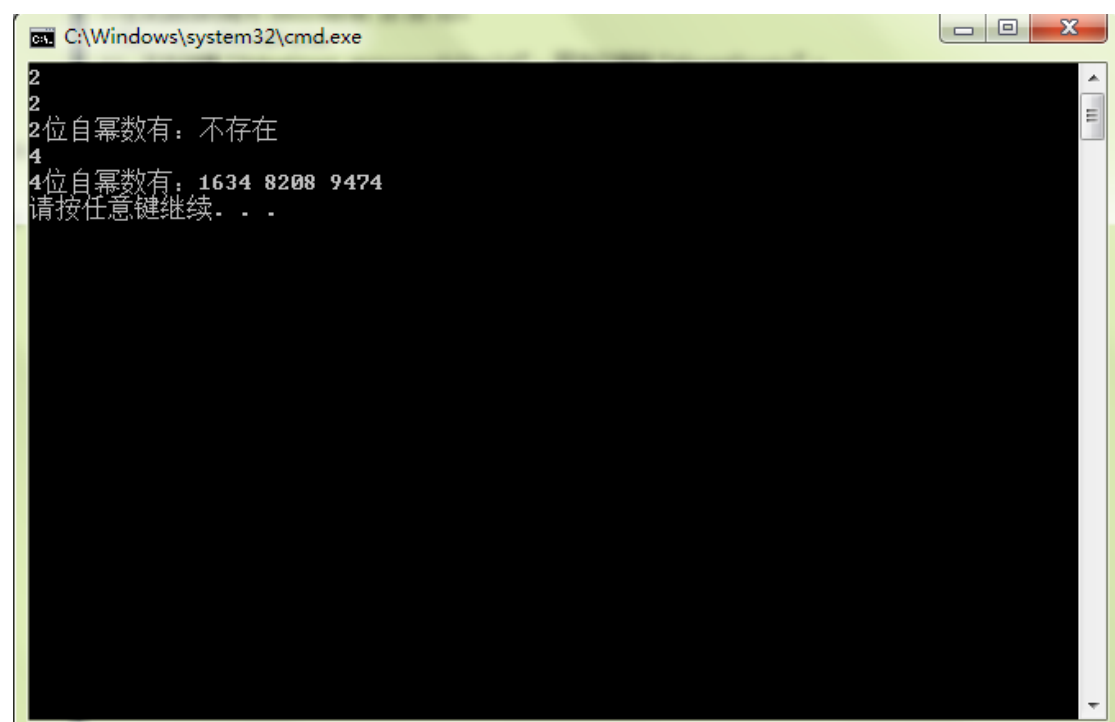
```
2
2
4
```

#### Sample Output

2 位自幂数有：不存在

4 位自幂数有：1634 8208 9474

附实际测试截图：



```
C:\Windows\system32\cmd.exe
2
2
4
2位自幂数有：不存在
4位自幂数有：1634 8208 9474
请按任意键继续...
```

## 6、度度熊大战僵尸

#### Description

僵尸最近老在百度大厦附近出没，因此公司派出了度度熊去消灭他。

度度熊有  $n$  件武器，第  $i$  件武器有物理攻击力  $A_i$  和魔法攻击力  $B_i$ 。在某个时刻  $t$ ，武器能造成的伤害为  $A_i + B_i * t$ 。僵尸有一个初始血量值  $H$ ，受到武器的攻击后，血量会减去武器的当前伤害值。如果某个时刻僵尸的血量值为负，则僵尸将原地满血复活为血量值  $H$ 。因此为了消灭僵尸，度度熊的最后一击，必须恰好使僵尸的血量为 0。

从时刻 1 开始的每个整数时刻，度度熊可以从自己的武器中挑选一个武器攻击僵尸一次，也可以攻击僵尸。一件武器可以在不同的时刻使用多次。

由于度度熊武器的限制，不是每个血量的僵尸都能杀死。度度熊希望能知道能杀死的僵尸中第  $k$  小的血量值是多少。

#### Input

输入的第一行包含两个整数  $n, k$ ，分别表示度度熊拥有的武器数和要求的血量是第几小的。

接下来  $n$  行表示度度熊拥有的武器，其中第  $i$  行包含两个整数  $A_i, B_i$ ，表示第  $i$  个武器的物理和魔法攻击力。

#### Output

输出包含一个整数，表示度度熊能杀死的僵尸中第  $k$  小的血量值。

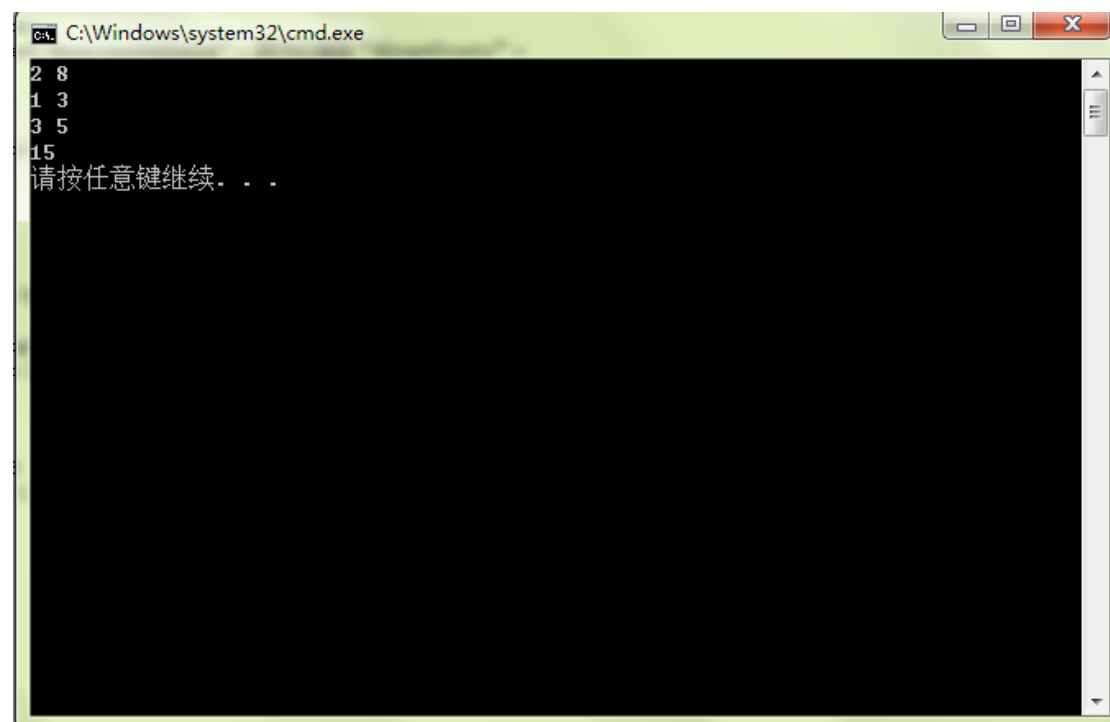
#### Sample Input

```
2 8
1 3
3 5
```

#### Sample Output

```
15
```

附实际测试截图：



```
C:\Windows\system32\cmd.exe
2 8
1 3
3 5
15
请按任意键继续. . .
```

## 7、DNA Sorting

### Description

One measure of "unsortedness" in a sequence is the number of pairs of entries that are out of order with respect to each other. For instance, in the letter sequence "DAABEC", this measure is 5, since D is greater than four letters to its right and E is greater than one letter to its right. This measure is called the number of inversions in the sequence. The sequence "AACEDGG" has only one inversion (E and D)---it is nearly sorted---while the sequence "ZWQM" has 6 inversions (it is as unsorted as can be---exactly the reverse of sorted).

You are responsible for cataloguing a sequence of DNA strings (sequences containing only the four letters A, C, G, and T). However, you want to catalog them, not in alphabetical order, but rather in order of "sortedness", from "most sorted" to "least sorted". All the strings are of the same length.

### Input

The first line contains two integers: a positive integer  $n$  ( $0 < n \leq 50$ ) giving the length of the strings; and a positive integer  $m$  ( $0 < m \leq 100$ ) giving the number of strings. These are followed by  $m$  lines, each containing a string of length  $n$ .

### Output

Output the list of input strings, arranged from "most sorted" to "least sorted". Since two strings can be equally sorted, then output them according to the original order.

### Sample Input

```
10 6
AACATGAAGG
TTTTGGCCAA
TTTGGCCAAA
GATCAGATTT
CCCGGGGGGA
ATCGATGCAT
```

### Sample Output

```
The result is:
CCCGGGGGGA
AACATGAAGG
GATCAGATTT
ATCGATGCAT
TTTTGGCCAA
TTTGGCCAAA
附实际测试截图:
```



```
C:\Windows\system32\cmd.exe
10 6
AACATGAAGG
TTTGGCCAA
TTTGGCCAA
GATCAGATT
CCCGGGGGGA
ATCGATGCAT
The result is:
CCCGGGGGGA
AACATGAAGG
GATCAGATT
ATCGATGCAT
TTTGGCCAA
TTTGGCCAA
请按任意键继续. . .
```