

# Streaming #5

## (NOIP 模拟赛 Day 1)

**竞赛时间：2014 年 10 月 3 日 18:30-22:00**

题目名称	珠	兔农	高维网络
目录	beads	rabit	cube
可执行文件名	beads	rabit	cube
输入文件名	标准输入		
输出文件名	标准输出		
每个测试点时限	1 秒	1 秒	1 秒
内存限制	128M	128M	128M
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	无	无	无
题目类型	传统	传统	传统

特别提示：评测在 Linux 下进行，C/C++语言开启-O2 优化开关，Pascal 语言开启-O1 优化开关。

答疑/题解：<http://tieba.baidu.com/p/3297669984>

提交：

[http://ch.ezoj.tk/contest/CH%20Round%20%2354%20-%20Streaming%20%235%20\(NOIP%E6%A8%A1%E6%8B%9F%E8%B5%9BDay1\)](http://ch.ezoj.tk/contest/CH%20Round%20%2354%20-%20Streaming%20%235%20(NOIP%E6%A8%A1%E6%8B%9F%E8%B5%9BDay1))

# 珠

## 【问题描述】

萌蛋有 $n$ 颗珠子，每一颗珠子都写有一个数字。萌蛋把它们用线串成了环。

我们称一个数字串是有趣的，当且仅当它的第 1 位是 2，且除了第 1 位以外的每一位都是 3。例如，2，233，2333333 都是有趣的数字串。

现在，你可以从这串珠子的任意一颗开始读，沿着顺时针或逆时针方向，到任意一颗珠子停止。这样，你就可以读出一个数字串来。

萌蛋想知道，所有能读出的有趣的数字串当中，最长的是哪一个数字串。当然，你也可能读不出任何一个有趣的数字串，你也需要对这种情况做出判断。

## 【输入文件】

输入只有一行，是一个数字串。这是从这串珠子的某一颗开始，顺时针读取恰好一圈得到的。

## 【输出文件】

输出只有一行，是能读出的最长有趣的数字串。特殊地，如果找不到任何有趣的数字串，应输出“TvT”（不含引号）。

## 【输入样例 1】

323

## 【输出样例 1】

233

## 【输入样例 2】

333

## 【输出样例 2】

TvT

## 【数据规模和约定】

对于 20% 的数据， $n \leq 3$ 。

对于 40% 的数据， $n \leq 100$ 。

对于 60% 的数据， $n \leq 1,000$ 。

另有 20% 的数据，输入的数字串中不含 3。

对于 100% 的数据， $n \leq 100,000$ 。

# 兔农

## 【问题描述】

（如果你想更好地理解本题，请先阅读 NOI2011 第一试“兔农”一题）

萌蛋近年收入不景气，正在她发愁如何能多赚点钱时，她听到隔壁的小朋友在讨论兔子繁殖的问题。（注：兔子是一种简单的单细胞生物）

问题是这样的：时刻 0 有 2 只刚出生的兔子。每一时刻，每只兔子都会分裂成为 2 只兔子。问时刻  $n$  共有多少只兔子？

聪明的你可能已经发现，时刻  $n$  的兔子数正好是第  $n + 1$  个 2 的幂次。萌蛋不懂什么叫幂，但她也发现了规律：时刻  $n + 1$  的兔子数等于时刻  $n$  的兔子数的 2 倍。前几个时刻（从 0 开始）的兔子数依次为：

2 4 8 16 32 64 128 256 512 ...

萌蛋发现越到后面兔子数增长的越快，期待养兔子一定能赚大钱，于是萌蛋在时刻 0 买了 2 只兔子开始培养。

每天，萌蛋都要给兔子们提供营养。兔子的培养基非常特别，每  $k$  只兔子占据一个培养基，最后剩下的不足  $k$  只占据一个培养基。由于兔子特别害怕孤独，如果某个培养基只有 1 只兔子，这只兔子就会很快死掉。

然而，每个时刻的兔子数仍然是可以计算的。例如，当  $k = 7$  时，前几个时刻（从 0 开始）的兔子数依次为：

2 4 7 14 28 56 112 224 448 ...

给定  $n$ ，你能帮助萌蛋计算时刻  $n$  她有多少只兔子么？由于答案可能非常大，你只需要告诉萌蛋时刻  $n$  的兔子只数对  $p$  的余数即可。

## 【输入文件】

输入只有一行，包含三个整数  $n$   $k$   $p$ 。

## 【输出文件】

输出只有一行，为一个整数，表示时刻  $n$  的兔子只数对  $p$  的余数。

## 【输入样例】

6 7 10086

## 【输出样例】

112

## 【数据规模和约定】

对于 30% 的数据， $n \leq 1,000,000$ 。

另有 30% 的数据， $k$  是  $p$  的正整数倍。

对于 100% 的数据， $n \leq 1,000,000,000, 2 \leq k \leq 1,000,000, 1 \leq p \leq 1,000,000$ 。

# 高维网络

## 【问题描述】

现在有一个 $d$ 维的坐标网格，其中第 $i$ 维坐标的范围是 $[0, a_i]$ 。

在这个范围内建立一个有向图：我们把范围内的每个整点（每一维坐标均为整数的点）当做图上的顶点。设点 $A(0,0,\dots,0), B(a_1, a_2, \dots, a_d)$ 。

对于范围内的点 $(x_1, x_2, \dots, x_d)$ ，它会向以下这些点（如果目标点在范围内）： $(x_1 + 1, x_2, \dots, x_d), (x_1, x_2 + 1, \dots, x_d), \dots, (x_1, x_2, \dots, x_d + 1)$ 连有向边。

现在从点 $A$ 到点 $B$ 会有若干条路径，路径的条数可以十分简单地算出。然而不幸的是，范围内有 $p$ 个点被破坏了（点 $A$ 和点 $B$ 不会被破坏），其中第 $i$ 个点的坐标为 $(x_{i,1}, x_{i,2}, \dots, x_{i,d})$ 。你需要算出从点 $A$ 到点 $B$ 剩余的路径条数。

由于答案可能很大，你只需要输出它对 $1,000,000,007$ 取模的结果。

## 【输入文件】

第一行为两个整数 $d, p$ 。

第二行为 $d$ 个整数，其中第 $i$ 个数是 $a_i$ 。

接下来 $p$ 行，每行 $d$ 个整数，其中第 $i$ 行第 $j$ 个数是 $x_{i,j}$ 。

## 【输出文件】

一个整数，表示从点 $A$ 到点 $B$ 剩余的路径条数对 $1,000,000,007$ 取模的结果。

## 【输入样例】

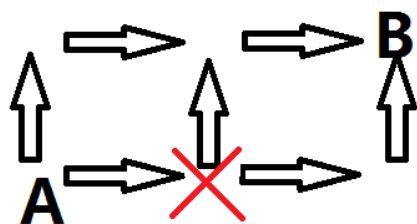
```
2 1
2 1
1 0
```

## 【输出样例】

```
1
```

## 【样例解释】

如图所示，当删掉点 $(1,0)$ 后，从点 $A$ 到点 $B$ 的3条路径只剩下了1条。



【数据规模和约定】

测试点编号	$a_i \leq$	$d =$	$p =$
1	100,000	1	0
2	8	2	1
3	9	2	2
4	10	2	3
5	4	3	2
6	5	3	4
7	100,000	1	5
8	100	2	0
9	500	2	0
10	1,000	2	20
11	50	3	0
12	100	3	20
13	10,000	2	0
14	50,000	2	0
15	100,000	2	500
16	10,000	3	0
17	100,000	3	500
18	100,000	10	500
19	100,000	50	0
20	100,000	100	500