

# 必可赛前公益众筹赛

## 第十试

时间：2024 年 11 月 5 日 08:00 ~ 12:00

题目名称	序列	道路	天真	牛仔
题目类型	传统型	传统型	传统型	传统型
目录	sequence	road	naive	nz
可执行文件名	sequence	road	naive	nz
输入文件名	sequence.in	road.in	naive.in	nz.in
输出文件名	sequence.out	road.out	naive.out	nz.in
每个测试点时限	3.0 秒	1.0 秒	1.0 秒	2.0 秒
内存限制	512 MB	512 MB	128 MB	512 MB
测试点数目	25	20	10	5
测试点是否等分	是	是	是	否

提交源程序文件名

对于 C++ 语言	sequence.cpp	road.cpp	naive.cpp	nz.cpp
-----------	--------------	----------	-----------	--------

编译选项

对于 C++ 语言	-lm -O2 -std=c++14
-----------	--------------------

### 注意事项与提醒（请选手务必仔细阅读）

1. 选手请直接提交源程序至 becoder.com.cn 上的对应比赛。
2. 输入输出文件名必须使用英文小写。
3. 选手提交的源程序必须存放在**已建立好的，且带有样例文件和下发文件**的文件夹中，文件夹名称与对应试题英文名一致。
4. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
5. C++ 中函数 main() 的返回值类型必须是 int，值必须为 0。
6. 若无特殊说明，结果比较方式为**忽略行末空格、文末回车后的全文比较**。
7. 程序可使用的栈空间大小与该题内存空间限制一致。
8. 在终端中执行命令 ulimit -s unlimited 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
9. 每道题目所提交的**代码文件大小限制为 100KB**。
10. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。

11. 输入文件中可能存在行末空格，请选手使用更完善的读入方式（例如 `scanf` 函数）避免出错。
12. 直接复制 PDF 题面中的多行样例，数据将带有行号，建议选手直接使用对应目录下的样例文件进行测试。
13. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
14. 请务必使用题面中规定的编译参数，保证你的程序在本机能够通过编译。此外**不允许在程序中手动开启其他编译选项**，一经发现，本题成绩以 0 分处理。
15. **对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。**

## 【前言】

弄一套合适的题目是很费时间和精力事情，这是我们第一次尝试公益活动，活动的初衷是为自觉比赛的选手提供一些比赛选项「能够离线完成最佳」，想把我们训练中遇到的较为合适或者很有启发性的题目尽量展示给大家，这个过程不可避免地遭遇原题，事实上，任何一个能够稳定输出的学校，都是依据大量的、成熟的原题进行训练。这是无可厚非的。

哪怕是组一套试题，其实也具有很大的工作量，部分分的配置、标程、数据、交互库或 SPJ、题解、题面等等。

希望大家认真对待每一道题目，即使知道来源，君勿多言，自己做自己的就行了，在市面价 200 500 一场的现状下，包含退费和奖励的 9.9 元报名费实在是微不足道的。

最后，我们欣赏行胜于言的作风，通过行动来改变，能够提供哪怕一道题目也远胜喋喋不休的抱怨。

## 序列 (sequence)

### 【题目描述】

你有一个长度为  $n$  的序列  $A$ 。

你喜欢 MEX，定义  $MEX(S)$  为最小的没有在  $S$  集中出现的且大于 0 的自然数。

定义区间  $[l, r]$  的美丽度为  $f(l, r) = MEX(A_l, \gcd(A_l, A_{l+1}), \dots, \gcd(A_l, A_{l+1}, \dots, A_r))$ 。

你想要知道序列  $A$  中  $\frac{n \times (n+1)}{2}$  个子区间的美丽度之和。

### 【输入格式】

从文件 `sequence.in` 中读入数据。

第一行一个整数  $n$ ，表示序列的长度。

第二行包含  $n$  个整数，第  $i$  个整数表示  $A_i$ 。

### 【输出格式】

输出到文件 `sequence.out` 中。

一行一个整数，表示答案。

### 【样例 1 输入】

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

### 【样例 1 输出】

```
1 30
```

见选手目录下的 `sequence/sequence1.in` 与 `sequence/sequence1.ans`。

### 【样例 1 解释】

下面仅举几个区间解释其  $f$  值的计算。

对于区间  $[1, 2]$ ， $f(1, 2) = MEX(3, \gcd(3, 2)) = MEX(3, 1) = 2$ 。

对于区间  $[3, 3]$ ， $f(3, 3) = MEX(1) = 2$ 。

对于区间  $[4, 4]$ ， $f(4, 4) = MEX(2) = 1$ 。

### 【样例 2】

见选手目录下的 `sequence/sequence2.in` 与 `sequence/sequence2.ans`。

该样例约束与测试点 1 ~ 3 一致。

### 【样例 3】

见选手目录下的 *sequence/sequence3.in* 与 *sequence/sequence3.ans*。  
该样例约束与测试点 4 ~ 7 一致。

#### 【样例 4】

见选手目录下的 *sequence/sequence4.in* 与 *sequence/sequence4.ans*。  
该样例约束与测试点 8 ~ 11 一致。

#### 【样例 5】

见选手目录下的 *sequence/sequence5.in* 与 *sequence/sequence5.ans*。  
该样例约束与测试点 12 ~ 14 一致。

#### 【样例 6】

见选手目录下的 *sequence/sequence6.in* 与 *sequence/sequence6.ans*。  
该样例约束与测试点 18 ~ 20 一致。

#### 【样例 7】

见选手目录下的 *sequence/sequence7.in* 与 *sequence/sequence7.ans*。  
该样例约束与测试点 21 ~ 25 一致。

#### 【数据范围】

对于 1000% 的数据,  $1 \leq n \leq 5 \times 10^6, 1 \leq A_i \leq 5 \times 10^6, 1 \leq l \leq r \leq n$ 。

测试点编号	$n \leq$	$\max A_i \leq$	特殊性质
1 ~ 3	2000	$5 \times 10^6$	无
4 ~ 7	$10^5$	$5 \times 10^6$	无
8 ~ 11	$5 \times 10^5$	200	无
12 ~ 14	$5 \times 10^6$	$5 \times 10^6$	A
15 ~ 17	$10^6$	$5 \times 10^6$	B
18 ~ 20	$5 \times 10^6$	$5 \times 10^6$	C
21 ~ 25	$5 \times 10^6$	$5 \times 10^6$	无

特殊性质 A: 保证  $\forall 1 \leq i \leq n, A_i \in \{1, 2, 3\}$ 。

特殊性质 B: 保证  $A_i$  在区间  $[1, 5 \times 10^6]$  内均匀随机生成。

特殊性质 C: 保证  $\forall 1 \leq i \leq n, A_i \bmod 2 = 1$ 。

请注意: 本题输入量过大, 请使用快读进行读入, 避免因为读入超时。

## 道路 (road)

### 【题目描述】

很久很久以前，中原地区分成了  $N$  个国家，编号为 1 到  $N$ ，任意两个国家都可互达。每个国家有一个攻击值  $A[i]$  和防御值  $B[i]$ 。

定义一个人从  $i$  国去  $j$  国的危险值为：假如  $A[i] > B[j]$ ，则危险值为  $A[i]^2 - B[j]^2$ ，否则危险值为 0。

现在，Nan 从国家 1 出发，经过每一个国家有且仅有一次，最后回到国家 1，要求找出一种方案，使得其中危险值的最大值最小。

### 【输入格式】

从文件 *road.in* 中读入数据。

第一行正整数  $N$ ，表示有  $N$  个国家。

第二行正整数  $A[1], A[2], x, y, z$ ，有等式  $A[i] = (x * A[i - 1] + y * A[i - 2] + z) \bmod 32767$ 。

第三行正整数  $B[1], B[2], x, y, z$ ，有等式  $B[i] = (x * B[i - 1] + y * B[i - 2] + z) \bmod 32767$ 。

### 【输出格式】

输出到文件 *road.out* 中。

输出一个数，表示危险值的最大值最小是多少。

### 【样例 1 输入】

```
1 5
2 2 4 1231 4432 123
3 123 45 3245 555 6676
```

### 【样例 1 输出】

```
1 9171832
```

见选手目录下的 *road/road1.in* 与 *road/road1.ans*。

### 【样例 1 解释】

$A$  数组为 2, 4, 13911, 5151, 3031

$B$  数组为 123, 45, 24364, 26060, 21765

其中一种最优方案为 1 - 2 - 4 - 3 - 5 - 1，危险值分别为 0, 0, 0, 0, 9171832

**【样例 2】**

见选手目录下的 *road/road2.in* 与 *road/road2.ans*。

**【样例 3】**

见选手目录下的 *road/road3.in* 与 *road/road3.ans*。

**【样例 4】**

见选手目录下的 *road/road4.in* 与 *road/road4.ans*。

**【样例 5】**

见选手目录下的 *road/road5.in* 与 *road/road5.ans*。

**【数据范围】**

保证 100% 的数据,  $A, B$  数组和  $x, y, z \in [0, 32767)$ 。

测试点编号	$n \leq$
1 ~ 4	10
5 ~ 10	$10^3$
11 ~ 16	$10^5$
17 ~ 20	$10^6$

## 天真 (naive)

### 【题目描述】

这是一个经典问题，放在这里只是检验一下大家会不会而已，别无他意！

这个问题是这样子的：有一个  $n$  层的大楼，存在这样一个  $x$  ( $0 \leq x \leq n$ )，在  $x$  层及以下你是天真的，在  $x$  层以上就会不天真。

现在有  $k$  个操作，每次操作可以去到大楼某一层，但是如果不天真了就不能再操作了，你想找到这个  $x$ ，问最坏情况下最少要找多少次。

### 【输入格式】

从文件 *naive.in* 中读入数据。

一行两个正整数，分别表示  $n, m$ 。

### 【输出格式】

输出到文件 *naive.out* 中。

一行一个正整数表示答案。

### 【样例 1 输入】

```
1 2
2 10 1
3 5 2
```

### 【样例 1 输出】

```
1 10
2 3
```

见选手目录下的 *naive/naive1.in* 与 *naive/naive1.ans*。

### 【样例 2】

见选手目录下的 *naive/naive2.in* 与 *naive/naive2.ans*。

### 【数据范围】

对于 30% 的数据， $n \times 1000$ ；

对于另外 20% 的数据， $k \leq 2$ ；

对于另外 30% 的数据， $T \leq 10000, n \leq 10^{18}, k \leq 64$ ；

对于 100% 的数据， $T \leq 100000, n \leq 10^{18}, k \leq 64$ ；

## 牛仔 (nz)

### 【题目描述】

nk 这次想创建一个长度为  $n$  的序列, 其中元素均在  $[1, K]$  中, 如果序列中有连续子序列长度为  $K$  且元素互不相同, 我们则称其为牛仔序列。

但 nk 觉得这样不好玩, 于是有生成了一个长度为  $m$  的序列  $A$ , 元素均在  $[1, K]$  中。定义一个长度为  $n$  的序列  $S$  的「牛度」为  $A$  在其中匹配的次数。

询问所有满足元素均在  $[1, K]$  中的牛仔序列「牛度」之和。

### 【输入格式】

从文件 *nz.in* 中读入数据。

第一行, 三个整数  $N$ ,  $K$  和  $M$ ;

第二行,  $M$  个整数, 表示题意中的  $A$

### 【输出格式】

输出到文件 *nz.out* 中。

一个整数, 表示「牛度」之和。

### 【样例 1 输入】

```
1 3 2 1
2 1
```

### 【样例 1 输出】

```
1 9
```

见选手目录下的 *nz/nz1.in* 与 *nz/nz1.ans*。

### 【样例 1 解释】

有  $(1, 1, 2), (1, 2, 1), (1, 2, 2), (2, 1, 1), (2, 1, 2), (2, 2, 1)$  总共 6 种牛仔序列, 它们分别能产生 2, 2, 1, 2, 1, 1 的「牛度」, 因此答案为  $2 + 2 + 1 + 2 + 1 + 1 = 9$ 。

### 【样例 2 输入】

```
1 10 3 5
2 1 1 2 3 3
```

### 【样例 2 输出】



1 1458

见选手目录下的  $nz/nz2.in$  与  $nz/nz2.ans$ 。

**【样例 3】**

见选手目录下的  $nz/nz3.in$  与  $nz/nz3.ans$ 。

**【数据范围】**

对于 100% 的数据，有  $1 \leq m \leq n \leq 25000$ ,  $1 \leq A_i \leq K \leq 400$ 。

子任务编号	特殊性质	分值
1	$n, K \leq 20$	10
2	$n \leq 20$	10
3	保证 $A$ 是牛仔序列	10
4	保证 $A$ 中元素互不相同	20
5	无	50