

# NOI2017模拟题

JOHNKRAM

题目名称	Left	Mid	Right
源文件名称	left	mid	right
输入文件名	left.in	mid.in	right.in
输出文件名	left.out	mid.out	right.out
每个测试点时限	1s	2s	4s
测试点数目	10	10	10
每个测试点分值	10	10	10
内存限制	512MB	512MB	512MB
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型
是否有SPJ	否	否	否
编译优化	-O2	-O2	-O2

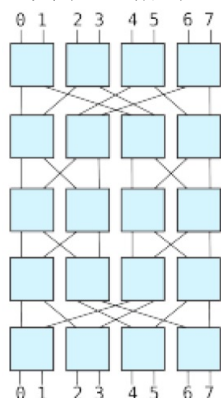
# 1 Left

## 1.1 问题描述

JOHNKRAM最近在研究排序网络，但他发现他不会制作比较器，于是他用交换器来代替比较器。

一个交换器有两个输入端X,Y和两个输出端X',Y'。如果交换器处于关闭状态，则X收到的信号会从X'发出，Y收到的信号会从Y'发出。如果交换器处于开启状态，则X收到的信号会从Y'发出，Y收到的信号会从X'发出。

JOHNKRAM设计了这样一个递归定义的网络：1阶网络就是一个交换器。 $n(n > 1)$ 阶网络的第一排是 $2^{n-1}$ 个交换器，接下来是两个 $n-1$ 阶网络，最后一排也是 $2^{n-1}$ 个交换器。将第一排的输出端和第二排的输入端分别从左到右标号为 $0 \sim 2^n - 1$ ，第一排的 $i$ 输出端连接到第二排的 $i \gg 1$ 输入端，其中 $\gg$ 指 $n$ 位二进制数的循环右移。类似，将倒数第一排的输入端和倒数第二排的输出端分别从左到右标号为 $0 \sim 2^n - 1$ ，倒数第二排的 $i$ 输出端连接到倒数第一排的 $i \ll 1$ 输入端，其中 $\ll$ 指 $n$ 位二进制数的循环左移。一个3阶的网络如下图所示：



JOHNKRAM通过开关交换器来调整网络。现在他对一个 $n$ 阶网络的 $2^n$ 个输入端分别输入了一个数，第 $i(0 \leq i < 2^n)$ 个输入端输入的是 $i$ 。然后他给出了一个长度为 $2^n$ 的排列 $p$ 。他希望你给出一种网络的状态，使得第 $i(0 \leq i < 2^n)$ 个输出端输出的是 $p_i$ 。

## 1.2 输入格式

输入文件包含不超过10组测试数据。

每个测试数据包含两行，第一行一个整数 $n$ ，表示是一个 $n$ 阶网络。

第二行 $2^n$ 个整数，表示排列 $p$ 。

输入文件以一个0结尾。

## 1.3 输出格式

对于每组数据，如果没有合法的解，则输出 $-1$ ，否则输出 $2n+1$ 行 $2^{n-1}$ 位二进制数，表示网络状态。如果一个交换器是开启的，则对应的位置上是1，否则是0。如果有多解，输出字典序最小的。

每个答案后打印一个空行。

#### 1.4 样例输入

```
2
3 2 1 0
3
3 7 4 0 2 6 1 5
0
```

#### 1.5 样例输出

```
00
11
11

0011
0000
0110
1111
1101
```

#### 1.6 数据规模与约定

对于20%的数据，保证 $n \leq 3$ 。

对于100%的数据，保证 $1 \leq n \leq 13$ 。

## 2 Mid

### 2.1 问题描述

JOHNKRAM在冬令营的时候被wys的卡常题坑了。他表示非常不爽，于是决定也出一道卡常题来祸害人。

有一个多重集，初始时空。JOHNKRAM进行了 $n$ 次操作，每次操作往多重集内插入一个整数。第 $i$  ( $1 \leq i \leq n$ )次操作完之后他会问你这个多重集内第 $\lfloor \frac{i+1}{2} \rfloor$ 小的数是多少。为了防止你的工作量过大，你只需要把每次询问的答案异或起来得到的值告诉他即可。

### 2.2 输入格式

第一行两个整数 $n$ 和 $a_1$ ，指操作的次数和第一次操作插入的数。

接下来插入的数按如下方法生成： $a_i = (1714636915 * a_{i-1} + 1681692777) * (846930886 * ans_{i-1} + 1804289383) \bmod 1000000007$ 。其中 $ans_{i-1}$ 指第 $i-1$ 次询问的答案。

### 2.3 输出格式

输出一个整数，表示所有 $ans_i$  ( $1 \leq i \leq n$ )异或得到的值。

### 2.4 样例输入

```
10 1
```

### 2.5 样例输出

```
943960841
```

### 2.6 数据规模与约定

对于30%的数据，保证 $n \leq 3 * 10^3$ 。

对于50%的数据，保证 $n \leq 1 * 10^6$ 。

对于100%的数据，保证 $1 \leq n \leq 3 * 10^7$ ， $1 \leq a_1 < 1000000007$ 。

## 3 Right

### 3.1 问题描述

JOHNKRAM和C\_SUNSHINE在玩一个游戏。游戏规则如下：有若干堆石子，游戏前选定一个正整数 $p$ ，JOHNKRAM先手，两个人轮流操作。定义一次操作是选择某一堆石子，然后拿出其中的 $p^k (k \in \mathbb{N})$ 个石子扔掉，不能操作者输。

C\_SUNSHINE表示判定谁能赢太简单了，于是他放了 $n$ 堆石子，编号为 $1 \sim n$ 。他每次把编号在某个区间内的石子堆加上若干个石子，或者询问以编号在某个区间内的石子堆进行游戏，是谁胜利。JOHNKRAM表示他不会做，于是他来向你求助。

### 3.2 输入格式

第一行三个数 $n, q, p$ ， $n$ 表示序列的长度， $q$ 表示接下来操作的次数， $p$ 的意义如题目描述中所说。

接下来一行 $n$ 个数，第 $i$ 个数表示初始时第 $i$ 堆石子的石子数量。

接下来 $q$ 行每行第一个数 $tp$ 表示操作类型， $tp = 0$ 表示修改， $tp = 1$ 表示询问。

对于一个修改操作，该行还会有三个数 $l, r, x$ ，表示把 $[l..r]$ 的所有石子堆加上 $x$ 个石子。

对于一个询问操作，该行还会有两个数 $l, r$ ，表示询问以 $[l..r]$ 的所有石子堆进行游戏是谁胜利。

### 3.3 输出格式

对于每一个询问操作，如果JOHNKRAM胜利则输出1，否则输出0。

### 3.4 样例输入

```
10 9 3
2 6 2 5 8 7 4 3 4 1
1 1 10
0 5 7 15
1 1 3
0 3 9 11
1 3 7
0 4 5 53
0 1 2 26
1 6 10
1 4 9
```

### 3.5 样例输出

```
0
```

0  
0  
1  
0

### 3.6 数据规模与约定

对于10%的数据，保证 $n, q \leq 10$ 。

对于20%的数据，保证 $n, q \leq 1 * 10^2$ 。

对于30%的数据，保证 $n, q \leq 5 * 10^3$ 。

对于40%的数据，保证 $n, q \leq 5 * 10^4$ 。

对于50%的数据，保证 $n, q \leq 7 * 10^4$ 。

对于60%的数据，保证 $n, q \leq 8 * 10^4$ 。

对于另外10%的数据，保证所有修改的 $l$ 和 $r$ 相同。

对于另外10%的数据，保证所有询问的 $l$ 和 $r$ 相同。

对于另外10%的数据，保证 $p \leq 1 * 10^2$ 。

对于100%的数据，保证 $1 \leq n, q, p \leq 1 * 10^5$ ，每一堆石子的初始数量 $\leq 1 * 10^5$ 。