

# NOIP模拟赛

(请选手务必仔细阅读本页内容)

## 一、题目概况

中文题目名称	绘制二叉树	魔术师	环上排序信息最优分割	野猪
英文题目名称	binary	magic	ring	wildboar
可执行文件名	binary	magic	ring	wildboar
输入文件名	binary.in	magic.in	ring.in	wildboar.in
输出文件名	binary.out	magic.out	ring.out	wildboar.out
提交文件名	binary.cpp	magic.cpp	ring.cpp	wildboar.cpp
每个测试点时限	1秒	3秒	2秒	6秒
测试点数目	20	20	20	20
每个测试点分值	5	5	5	5
内存限制	512MB	512MB	1024MB	1024MB
题目类型	传统题	传统题	传统题	传统题

## 二、编译命令

题目名称	binary	magic	ring	wildboar
对于C++语言	-o binary binary.cpp -lm -std=c++14 -O2 -Wl,--stack=2147483647	-o magic magic.cpp -lm -std=c++14 -O2 -Wl,--stack=2147483647	-o ring ring.cpp -lm -std=c++14 -O2 -Wl,--stack=2147483647	-o wildboar wildboar.cpp -lm -std=c++14 -O2 -Wl,--stack=2147483647

## 三、注意事项

- 文件夹名、文件名(程序名和输入输出文件名)必须使用英文小写。
- C/C++中函数main()的返回值类型必须是int,程序正常结束时的返回值必须是0。
- 统一评测时采用的机器配置为：windows下lemon评测。
- 请尽力优化，会收获更多的部分得分。
- AK了不要大声喧哗，没AK也不要。

## 绘制二叉树(binary)

### 题目描述

二叉树是一种基本的数据结构，它要么为空，要么由根结点，左子树和右子树组成，同时左子树和右子树也分别是二叉树。

当一颗二叉树高度为  $m - 1$  时（根节点高度为 0），共有  $m$  层。若一棵二叉树除第  $m$  层外，其他各层的结点都拥有两个子节点，且第  $m$  层的节点无任何子节点，则其为一棵满二叉树。

现在，需要你用程序来绘制一棵二叉树，它由一棵满二叉树去掉若干结点而成。对于一棵满二叉树，我们需要按照以下要求绘制：

- 1. 结点用小写字母 `o` 表示，对于一个父亲结点，用 `/` 连接左子树，用 `\` 连接右子树。
- 2. 定义  $[i, j]$  为位于第  $i$  行第  $j$  列的某个字符。若  $[i, j]$  为 `/`，那么  $[i - 1, j + 1]$  与  $[i + 1, j - 1]$  要么为 `o`，要么为 `/`。若  $[i, j]$  为 `\`，那么  $[i - 1, j - 1]$  与  $[i + 1, j + 1]$  要么为 `o`，要么为 `\`。同样，若  $[i, j]$  为第  $1 \sim m - 1$  层的某个结点 `o`，那么  $[i + 1, j - 1]$  为 `/`， $[i + 1, j + 1]$  为 `\`。
- 3. 对于第  $m$  层结点也就是叶子结点点，若两个属于同一个父亲，那么它们之间由 3 个空格隔开；若两个结点相邻但不属于同一个父亲，那么它们之间由 1 个空格隔开。第  $m$  层左数第 1 个结点之前没有空格。

最后需要在一棵绘制好的满二叉树上删除  $n$  个结点（包括这个结点的左右子树，以及与父亲的连接），原有的字符用空格替换（空格为 ASCII 32，若输出 ASCII 0 会被算作错误答案）。

## 输入格式

第 1 行包含 2 个正整数  $m$  和  $n$ ，为需要绘制的二叉树层数和需要删除的结点数。

接下来  $n$  行，每行两个正整数，表示删除第  $i$  层从左往右数第  $j$  个结点。

## 输出格式

按照题目要求绘制的二叉树。

## 样例

### 样例1输入

2 0

### 样例1输出

```
  o
 / \
o   o
```

### 样例2输入

4 0

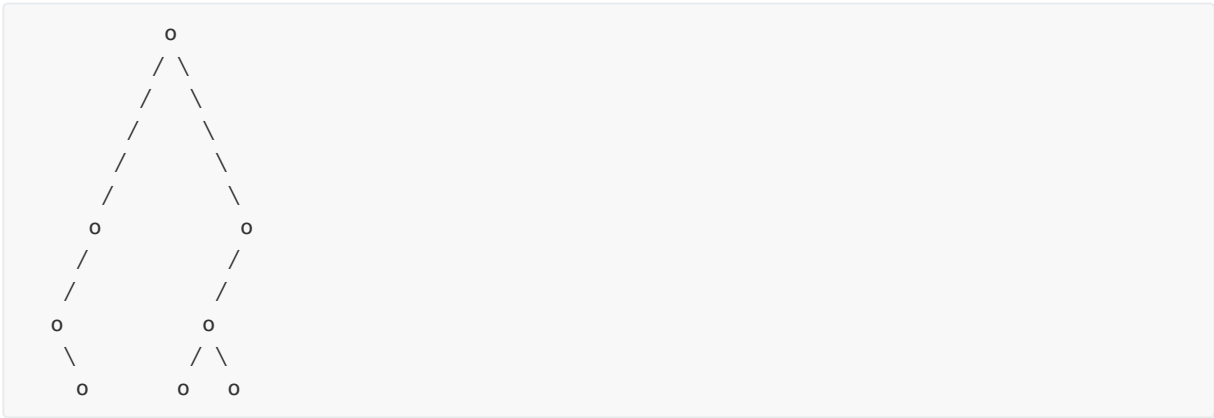
### 样例2输出

```
      o
     / \
    /   \
   /     \
  /       \
 o         o
 / \       / \
o  o     o  o
/ \ / \  / \ / \
o o o o o o o o
```

### 样例3输入

4 3
3 2
4 1
3 4

样例3输出



数据范围

对于所有数据，满足： $2 \leq m \leq 10, 0 \leq n \leq 10, 1 < i \leq m, j \leq 2^{i-1}$ 。

测试点编号	特殊性质
1 ~ 6	$n = 0$
7 ~ 12	$m \leq 5$
13 ~ 20	无特殊限制

魔法师(magic)

题目描述

Tin 是一位著名的魔术师，他的一个经典魔术与洗牌有关。

Tin 会准备一套牌，总共  $n$  张（保证  $n$  为偶数），各编号为  $1 \sim n$ ，一开始的时候牌是乱的且倒扣在桌子上。紧接着他开始表演洗牌，在洗牌的任意时刻，观众都可以向 Tin 询问从底往上数第  $t$  张牌是什么牌，很显然 Tin 一定会立即回答出正确答案。

事实上，Tin 采用如下方式来完成这个魔术，首先他记下了一开始的  $n$  张牌的顺序，接着采用如下技巧洗牌：

1. 拿起自顶向下  $\frac{n}{2}$  张牌放在右手，自底向上  $\frac{n}{2}$  张牌放在左手，牌的正面对着桌子。
2. 借助他的记忆，将左右手最底下的牌进行比较，将编号较小的那张牌放下，重复这个操作直到左右手一边为空。
3. 将还有牌的那只手手上的所有牌放下。

请你写一个程序模拟 Tin 的魔术。

输入格式

第一行两个整数  $N, Q$ 。

接下来一行  $N$  个整数  $p_i$ ，从底向上描述了整个牌堆。

接下来  $Q$  行，一行一个询问  $t, i$ ，表示询问  $t$  次洗牌后自底向上第  $i$  张牌编号是多少。

输出格式

对于每一个询问，输出你的答案。

样例

样例1输入

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

样例1输出

```
2
2
5
```

样例2输入

```
6 6
2 1 5 4 6 3
0 1
1 1
0 3
1 3
0 6
10 6
```

样例2输出

```
2
2
5
4
3
3
```

样例3输入

```
10 10
7 5 2 9 10 8 4 3 6 1
3 1
3 2
3 3
3 4
3 5
3 6
3 7
3 8
3 9
3 10
```

样例3输出

```
2
3
6
1
7
5
8
4
9
10
```

样例3解释

洗牌次数	自底向上的牌堆
0	7 5 2 9 10 8 4 3 6 1
1	7 5 2 8 4 3 6 1 9 10
2	3 6 1 7 5 2 8 4 9 10
3	2 3 6 1 7 5 8 4 9 10

数据范围

对于全部数据，满足  $1 \leq N \leq 2 \times 10^5$ ， $N$  为偶数， $1 \leq Q \leq 10^6$ ， $0 \leq t \leq 10^9$ ， $p$  为  $1 \sim n$  的排列， $1 \leq i \leq N$ 。

测试点编号	约束
1 ~ 6	$N \leq 10^3$
7 ~ 10	每一个询问的 $t$ 相同
11 ~ 15	$N, Q \leq 10^5$
16 ~ 20	无特殊限制

环上排序信息最优分割(ring)

题目描述

给定  $n$  个序列  $\{a_{1,i}\}, \{a_{2,i}\}, \dots, \{a_{n,i}\}$ ，第  $i$  个序列的长度为  $m_i$ ，每个序列的每个元素都是 0 到  $2 \times 10^6$  之间的整数。定义  $i$  的后继为  $i + 1$  ( $1 \leq i \leq n - 1$ )，而  $n$  的后继是 1。 $i$  的后继记作  $\text{succ}(i)$ 。

定义一个序列的代价为，向序列中加入一个 0 和一个  $2 \times 10^6$ ，排序后，相邻两个数的差的平方之和。即若排序后是  $0 = p_0 \leq p_1 \leq p_2 \leq \dots \leq p_{k-1} \leq p_k = 2 \times 10^6$ ，那么代价为  $\sum_{i=0}^{k-1} (p_{i+1} - p_i)^2$ 。

定义一个分割为整数序列  $x_1, x_2, \dots, x_n$ ，满足  $1 \leq x_i \leq m_i$ 。

定义第  $i$  个分割后的序列是由  $a_i$  的  $[x_i, m_i]$  号元素，加上  $a_{\text{succ}(i)}$  的  $[1, x_{\text{succ}(i)} - 1]$  号元素组成的序列。定义一个分割的代价是所有  $n$  个分割后的序列的代价之和。

求代价最小的分割。输出最小代价的值即可。

输入格式

第一行一个整数  $n$ 。

接下来的  $n$  行，每行包含一个整数  $m_i$  和  $m_i$  个整数  $a_{i,1}, \dots, a_{i,m_i}$ 。

输出格式

一行一个整数表示最小代价。

## 样例

### 样例1输入

```
4
5 414276 935411 204664 302847 1142143
5 162307 1199651 1168780 39659 991911
6 1204312 442315 639803 28852 1019073 143732
4 279750 1185347 612942 1086837
```

### 样例1输出

```
4522800735482
```

### 样例1解释

取  $x_1 = 2, x_2 = 3, x_3 = 3, x_4 = 3$ ，得到的四个序列分别为：  
 $\{935411, 204664, 302847, 1142143, 162307, 1199651\}$ ,  $\{1168780, 39659, 991911, 1204312, 442315\}$ ,  
 $\{639803, 28852, 1019073, 143732, 279750, 1185347\}$  和  $\{612942, 1086837, 414276\}$ 。

这四个序列的权值分别为：

- $(162307 - 0)^2 + (204664 - 162307)^2 + (302847 - 204664)^2 + (935411 - 302847)^2 + (1142143 - 935411)^2 + (1199651 - 1142143)^2 + (2000000 - 1199651)^2 = 1124518604972$
- $(39659 - 0)^2 + (442315 - 39659)^2 + (991911 - 442315)^2 + (1168780 - 991911)^2 + (1204312 - 1168780)^2 + (2000000 - 1204312)^2 = 1131425013362$
- $(28852 - 0)^2 + (143732 - 28852)^2 + (279750 - 143732)^2 + (639803 - 279750)^2 + (1019073 - 639803)^2 + (1185347 - 1019073)^2 + (2000000 - 1185347)^2 = 997321197822$
- $(414276 - 0)^2 + (612942 - 414276)^2 + (1086837 - 612942)^2 + (2000000 - 1086837)^2 = 1269535919326$

总和为  $1124518604972 + 1131425013362 + 997321197822 + 1269535919326 = 4522800735482$ ，可以证明，不存在更优的构造方法。

## 数据范围

对于所有测试数据： $n \geq 2, m_i \geq 2, \sum m_i \leq 2 \times 10^5, 0 \leq a_{i,j} \leq 2 \times 10^6$ 。

每个测试点的具体限制见下表：

测试点编号	特殊性质
1 ~ 4	$n \leq 6$ 且 $m_i \leq 10$
5 ~ 8	$\sum m_i \leq 100$
9 ~ 12	$\sum m_i \leq 1000$
13 ~ 16	$\sum m_i \leq 5 \times 10^4$
17 ~ 20	无特殊限制

## 野猪(wildboar)

### 题目描述

JOI 君是生活在 IOI 森林里的一头野猪。森林可视为一个包含  $N$  个结点,  $M$  条带权无向边的连通图。结点的编号分别为  $1 \dots N$ 。 $i$  号边连接结点  $A_i$  和  $B_i$ , 权值为  $C_i$ 。保证  $(A_i, B_i) \neq (A_j, B_j)$ , 并且保证: 对于任意两点互相可达。

开始时有一个长度为  $L$  的序列  $X_1, X_2 \dots X_L$ , 表示 JOI 君开始时在  $X_1$ , 它要依次访问结点  $X_2 \dots X_L$ 。序列中可能有重复结点, 但保证序列中相邻两结点不同, 即保证序列中  $X_j \neq X_{j+1}$ 。注意, 不要求从  $X_j$  直达  $X_{j+1}$ , JOI 君可以从  $X_j$  出发, 经过其他结点作为中转, 再到达  $X_{j+1}$ 。但是, JOI 君不能沿原路返回前一个到达的结点。参见样例。

接下来有  $T$  次修改, 每次修改会给出两个整数  $P_k, Q_k$ , 表示将  $X_{P_k}$  修改为  $Q_k$ 。每次修改后, JOI 君想知道: 他能否找到满足要求的路径。如果能, 请输出最短路的长度, 反之则输出  $-1$ 。

## 输入格式。

第一行, 输入四个整数  $N, M, T, L$ 。

接下来的  $M$  行, 每行三个整数  $A_i, B_i, C_i$ 。

接下来的  $L$  行, 每行一个整数  $X_j$ 。

接下来的  $T$  行, 每行两个整数  $P_k, Q_k$ 。

## 输出格式

输出共  $T$  行, 第  $i$  行有一个整数, 表示查询的结果。

## 样例

### 样例1输入

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

### 样例1输出

```
3
```

### 样例1解释

从结点 1 沿着 1 号道路到结点 2, 再沿 2 号道路到结点 3, 再沿 3 号道路到结点 1。

注意 JOI 君在结点 2 时不能沿着 1 号道路直接回到结点 1。

### 样例2输入

```
4 4 4 3
1 2 1
2 3 1
1 3 1
1 4 1
4
1
3
3 4
1 2
3 2
2 4
```

样例2输出

```
5
2
3
-1
```

样例2解释

在第一天,  $\{X_n\} = 4, 1, 4$ , JOI 君可以沿着 4 号道路从结点 4 到 1。然后 JOI 君再依次经过 1, 2, 3, 4 号道路回到结点 4。

注意, 尽管 JOI 君开始沿着 4 号道路从结点 4 到 1, 后来又沿着 4 号道路从结点 1 到 4, 但由于 JOI 君没有沿原路返回前一个到达的结点, 因此这一方案合法。

数据范围

对于所有测试点:  $2 \leq N \leq 2000, N - 1 \leq M \leq 2000, 1 \leq T \leq 10^5, 2 \leq L \leq 10^5, 1 \leq A_i < B_i \leq N, 1 \leq C_i \leq 10^9$ 。

测试点编号	特殊性质
1 ~ 4	$N, M, L, C \leq 10, T = 1$
5 ~ 8	$N, M \leq 500, T = 1$
9 ~ 12	$T = 1$
13 ~ 20	无特殊限制