

# 2023 重庆大学 ACM 程序设计挑战赛 暨社团新生赛

CQU-ACM Challenge 2023

## 正式赛

时间：2023 年 10 月 1 日 15:00 ~ 8 日 18:00

### 目 录

A. CQUer 最喜欢的一门课 / Course	2
B. 货物运输 / Transport	4
C. 梦不凌乱 / Dream	6
D. 化溪字哪里贵了? / Bargain	7
E. UNO! / Uno	9
F. 随机数表 / Random	11
G. 星途 / Travel	13
H. 回到过去 / Past	15
I. 穿越 / Stay	17
J. 数数 / Counting	19
K. 重逢 / Reunion	20
L. 紧急支援 / Emergency	23
M. 流动的城市 / Flow	25

N. 浸泡！脱水！ / Cube	27
O. 最尴尬的一集 / Awkward	28
P. 轮回的房间 / Loop	30
Q. 彗星坠落的那一天 / Comet	33

## A. CQUer 最喜欢的一门课 / Course

时间限制：1.0 秒

空间限制：512 MiB

**【题目描述】**

作为一名学习委员，小 P 在暑假期间收到了一项特殊的任务——调查大一学生本学年最喜爱的课程。

以下是本次调查的课程范围及其相关信息：

课程名称	英文代号	课程代码	开课学院	学分	开课专业
高等数学 II-1	gaoshu2-1	MATH10821	数学与统计学院	5	30
高等数学 II-2	gaoshu2-2			6	27
文明经典系列 A	wmjd-A	HG00080	博雅学院 (高研院)	3	67
文明经典系列 B	wmjd-B	HG00081			
大学物理 II-1	dawu2-1	PHYS10013	物理学院	3.5	16
大学物理 II-2	dawu2-2	PHYS10023		4	36
中国近现代史纲要	zgjxds	MT10200	马克思主义学院	3	77
思想道德与法治	sxddfz	MT10101		2	73

办事向来勤奋的小 P 很快在群里发起了问卷调查。然而，整整一天时间，小 P 只收到了寥寥几份样本。

小 P 知道，要在假期期间收集到所有同学的样本是毫无希望的——毕竟一到暑假，群消息这种东西也就没有人会再看了。于是，他决定将自己最喜欢的一门课程当作调查结果汇报上去。

小 P 认为，一门好的课程需要满足以下几个条件：

- 首先，这门课程学分占比要大，具体而言，它的学分要在 3 分及以上。
- 其次，这门课程面向的群体要广，具体而言，它开课专业的数量至少要占该年级专业总数的一半以上（已知小 P 所在年级的专业总数为 77）。
- 在所有满足以上条件的课程之中，小 P 会选择课程中文名称字典序最小的一个作为自己最喜欢的课程。

这样一来，小 P 最喜欢的课程便呼之欲出了。但在汇报前，他想考考屏幕前的你：你知道他最喜爱的课程是哪一门吗？

请输出对应课程的英文代号。

**【输入格式】**

本题无输入。

**【输出格式】**

输出到标准输出。

输出一个字符串，表示小 P 最喜欢的课程的英文代号。如若答案是思想道德与法治，应当输出 `sxddfz`。

**【提示】**

或许你只看标题都知道答案是什么（doge）。

## B. 货物运输 / Transport

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

C 国的某中心市区近期需要紧急从隔壁的 D 国进口一批货物。

已知本次运输计划共计需要运进  $n$  箱货物，物流公司为本次运输提供了 A 和 B 两种方案：

- 方案 A：第一次提供的货车可以最多运送任意  $a_1$  箱货物，随着货车的不断召集，下一次运输的货物的数目可以提高  $a_2$ 。
- 方案 B：第一次提供的货车只能运送  $b_1$  ( $b_1 < a_1$ ) 箱货物，但公司花费了更多资金用于招募货车，这使得每次完成运输后，下一次可运输的货物提高  $b_2$  ( $b_2 > a_2$ )。为了能够尽快完成本次运输计划，请你帮助 C 国选出耗费运送次数最小的方案。

### 【输入格式】

从标准输入读入数据。

输入仅一行，包含五个正整数  $n, a_1, a_2, b_1, b_2$ ，分别表示需要运送货物的箱数及方案 A、方案 B 中分别提及的常量  $a_1, a_2$  与  $b_1, b_2$ 。

数据保证  $1 \leq n \leq 100000, 1 \leq b_1 < a_1 \leq 1000, 0 \leq a_2 < b_2 \leq 100$ 。

### 【输出格式】

输出到标准输出。

输出一行一个字符串。若方案 A 耗费的次数更少，输出 **Plan A**；反之，若方案 B 耗费的次数更少，则输出 **Plan B**；特别地，若两种方案所需运送次数恰好相同，则输出 **Same**。

### 【样例 1 输入】

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

### 【样例 1 输出】

```
1 Plan A
```

**【样例 1 解释】**

虽然方案 B 每次可运输的货物数增速更快，但方案 A 仅需 2 次就恰好能完成本次运输任务，而方案 B 两次之内最多只能够运输 4 箱货物。

**【样例 2 输入】**

```
1 7 2 0 1 2
```

**【样例 2 输出】**

```
1 Plan B
```

**【样例 3 输入】**

```
1 9 3 1 1 2
```

**【样例 3 输出】**

```
1 Same
```

## C. 梦不凌乱 / Dream

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

某天夜深人静时，小 P 做了一个奇特的梦。梦里有一个画面令他印象极为深刻。但醒来时，他已经忘记了这个画面的具体形状，只记得这个画面的一些特征。请你帮他还原这个梦境。

具体来说，你需要构造一个  $n \times m$  的矩阵  $\{c_{i,j}\}$ ，使得：

- 对于任意  $i, j$  满足  $1 \leq i \leq n, 1 \leq j \leq m$ ，则有  $c_{i,j} \in \{R, G, B\}$ 。
- 对于任意  $i, j$  满足  $1 \leq i < n, 1 \leq j < m$ ，元素  $c_{i,j}, c_{i+1,j}, c_{i,j+1}$  互不相同。
- 对于任意  $i, j$  满足  $2 \leq i \leq n, 2 \leq j \leq m$ ，元素  $c_{i,j}, c_{i-1,j}, c_{i,j-1}$  也互不相同。

### 【输入格式】

从标准输入读入数据。

输入仅一行两个整数  $n, m$  ( $1 \leq n, m \leq 2000$ ) 分别表示你需要构造的矩阵的长和宽。

### 【输出格式】

输出到标准输出。

输出包含  $n$  行，每行为一个长度为  $m$  的字符串，表示你构造的矩阵  $\{c_{i,j}\}$ 。

### 【样例 1 输入】

```
1 2 2
```

### 【样例 1 输出】

```
1 RG
2 BR
```

### 【样例 1 解释】

样例仅展示了合法的一组解。只要你构造的矩阵符合题目的限制，均会视为正确。

## D. 化溪字哪里贵了? / Bargain

时间限制: 2.0 秒

空间限制: 512 MiB

### 【题目背景】

化溪字哪里贵了? 这么多年都是这个价格好吧! 不要睁着眼睛乱说! 有的时候  
#&#&!#%#^%...

### 【题目描述】

小 P 是一位热衷于观看网络直播的正义青年, 某天他突然发现某带货主播的商品价格已超越黄金。为此, 他决定带上自己的死神镰刀和雷神之锤去找他砍一砍价。

由于小 P 体力有限, 他只能和对方砍价  $n$  轮。在第  $i$  轮, 小 P 可以有两种选择:

- 挥动死神镰刀, 若当前价格为  $x$ , 则变为  $\lfloor \frac{x}{a_i} \rfloor$ ;
- 挥动雷神之锤, 若当前价格为  $x$ , 则变为  $x - b_i$ 。

小 P 希望最终商品的价格尽可能的低, 但他不善计算, 因此这个问题交给你。请你帮小 P 算出该商品可能的最低价格。

另外, 小 P 也不愿意出现商家倒贴钱的情况。因此, 若在某一轮结束时, 商品价格小于 0, 小 P 会让价格强制变为 0。

### 【输入格式】

从标准输入读入数据。

第一行输入两个整数  $n, k$ 。其中  $n$  表示小 P 砍价的轮数,  $k$  表示商品的初始价格。接下来第 2 行至第  $n$  行每行包括两个正整数, 其中第  $i+1$  行的两个数分别为  $a_i, b_i$ 。数据保证  $1 \leq n \leq 10^6, 79 \leq k \leq 10^9, 1 \leq a_i, b_i \leq 10^9$ 。

### 【输出格式】

输出到标准输出。

输出一行一个正整数  $x$ , 表示最终商品所能变成的最低价格。

### 【样例 1 输入】

```
1 3 79
2 2 10
3 1 15
4 2 20
```



**【样例 1 输出】**

```
1 4
```

**【样例 1 解释】**

第一轮挥动死神镰刀，随后商品价格变为  $\max(\lfloor \frac{79}{2} \rfloor, 0) = 39$ ；

第二轮挥动雷神之锤，随后商品价格变为  $\max(39 - 15, 0) = 24$ ；

第三轮挥动雷神之锤，随后商品价格变为  $\max(24 - 20, 0) = 4$ 。

**【样例 2 输入】**

```
1 1 79
2 20 150
```

**【样例 2 输出】**

```
1 0
```

**【样例 2 解释】**

小 P 挥动雷神之锤，商品价格直接变为  $\max(79 - 150, 0) = 0$ 。

## E. UNO! / Uno

时间限制： 1.0 秒

空间限制： 512 MiB

**【题目描述】**

小 C 已经厌倦了做这些无聊的题目。于是，她掏出了大家喜闻乐见的 UNO！

众所周知，一套完整的 UNO 牌内包含有 4 张 +4 类型的卡牌和 8 张 +2 类型的卡牌，其中 +4 类型的卡牌只有一种花色；+2 类型卡牌包含红、黄、蓝、绿四种花色各两张。它可以让下家强制摸相应的牌数，同时也能继续向下传递叠加效果。因此便产生了一种名为“叠加”的特殊玩法。

具体的，我们定义一轮合法的用于叠加判定的出牌需要满足如下条件：

- 整轮判定中 +4 类型的卡牌不超过 4 张；+2 类型的卡牌不超过 8 张，且每种颜色最多存在两张。
- 若上一回合所出卡牌的类型为 +2，则本回合出牌可为 +2 或 +4；若上一回合所出卡牌的类型为 +4，则本回合只能出 +4 类型的卡牌。
- 若当前玩家不能或不想打出符合判定要求的卡牌，此时该玩家必须在数字牌堆（假设数量不限）中取出与本轮所有出牌的牌面数字总和数目相等的卡牌作为惩罚。惩罚完成后，本轮判定**随即结束**。

例如，一轮判定中玩家可能先后打出：+2, +2, +4, +4。此时第五回合对应玩家没有打出符合判定要求的 +4 类型卡牌，故罚牌 12 张，并结束本轮判定。

现在给你一个出牌序列，请你判定其是否为一种合法的用于单轮叠加判定的出牌顺序。

**【输入格式】**

从标准输入读入数据。

第一行包含一行一个正整数  $n(1 \leq n \leq 20)$ ，表示序列中出现的牌数。

接下来  $n$  行，每行给出一张牌的信息，有如下两种情况：

- 4: 表示这张牌是 +4 类型的卡牌。
- 2  $c_i$ : 表示这张牌是 +2 类型的卡牌，其中  $c_i$  表示花色，且  $c_i \in \{R, Y, B, G\}$ 。

**【输出格式】**

输出到标准输出。

输出一行一个字符串，表示该出牌序列是否符合题意。若是，输出 Yes；否则输出 No。

**【样例 1 输入】**

```
1 5
2 2 R
3 2 B
4 4
5 4
6 4
```

**【样例 1 输出】**

```
1 Yes
```

**【样例 2 输入】**

```
1 5
2 2 R
3 2 G
4 4
5 4
6 2 B
```

**【样例 2 输出】**

```
1 No
```

## F. 随机数表 / Random

时间限制：1.0 秒

空间限制：512 MiB

### 【题目背景】

随机数表是由随机生成的从 0 到 9 十个数字所组成的数表，是统计工作者用计算机生成的随机数组成，以此保证各个个体被抽取的概率相等的有效工具。

在本题中，你便需要编程实现利用随机数表实现不重复抽样的过程。

如果你在高中数学中并未学习过随机数表抽样的方法，可以[点击这里](#)进行快速了解。

### 【题目描述】

给你一个  $n$  行  $m$  列的随机数表  $\{a_{i,j}\}_{n \times m}$ 。

现在工厂有  $x$  个零件需要进行测试，将其编号为 01, 02, 03, ...,  $x$ 。

假设现在需要抽取  $y$  个不重复的样本，从该随机数表的第  $x_0$  行  $y_0$  列开始读取向右依次读取 2 个数据。请求出得到的  $y$  个样本编号分别为多少？

### 【输入格式】

从标准输入读入数据。

第一行两个整数  $n, m$  ( $5 \leq n, m \leq 1000$ )，表示给定随机数表的长和宽。

接下来  $n$  行为一个  $n$  行  $m$  列的随机数表  $\{a_{i,j}\}_{n \times m}$ ，无任何分隔符号。

最后一行包含四个整数  $x_0, y_0, x, y$  ( $1 \leq x_0 \leq n, 1 \leq y_0 \leq m, 10 \leq x < 100, 1 \leq y < x$ )，其中前两个数表示读取的起始位置  $(x_0, y_0)$ ，倒数第二个数  $x$  表示测试零件的数目，最后一个数  $y$  表示需要抽取样本的数目。

数据保证对于任意  $i \in [1, n], j \in [1, m]$ ， $a_{i,j}$  均在  $0 \sim 9$  之间随机生成。

### 【输出格式】

输出到标准输出。

输出一行  $y$  个整数，相邻两个数用单个空格隔开，其中第  $i$  个数表示按照以上方法产生的第  $i$  个样本单位号码，不需要保留编号的前导零。

特别地，如果不能得到相应的  $y$  个样本编号，请输出 -1。

### 【样例 1 输入】

```
1 6 10
2 0347437386
3 3696473661
4 4699698162
5 9774246762
6 4281145720
7 4253323732
8 1 6 95 10
```

**【样例 1 输出】**

```
1 37 38 63 69 64 73 66 14 16 29
```

**【样例 1 解释】**

样例即所给链接中【方法介绍】板块给出的例子（由于篇幅原因，对随机数表的大小有所调整），你可以结合该阐述对样例进行更深入的理解。

## G. 星途 / Travel

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

身处多条交通航线和运输路线的交汇之处，R 星上部署着大量中转仓库及备用物资供往来飞船使用。我们假设 R 星上顺次排列着  $n$  个基站，它们从左往右依次编号为  $1, 2, 3, \dots, n$ ，每个基站上都部署着大量的运输飞船。

L 是 R 星上一位资深的运筹学家。最近，他研究出了一种新的运输策略，可以使得运输飞船在整个运输过程中消耗的能源达到最低。他是这样规定的每个飞船的运输策略的：当停留在某个基站的某个运输飞船被下达工作命令后，飞船会不断往右飞行若干次，并装载上它所经过基站上的全部物资（包含起点）。其中第  $i$  次飞行时，假设当前在第  $x$  个基站起飞，那么它将会在第  $x+i$  号基站降落。如果不存在第  $x+i$  号基站，飞船将会自动飞往运输中心并结束整个任务。

为了评估这一运输策略的实际运输效率，L 计划模拟一遍 R 星上某一天的运输过程。因此，他给出了  $n$  个基站初始时分别需要送出的物资量  $w_i$ ，并提出了一天中依次发生的  $Q$  个事件：

- **1 x v**，表示第  $x$  个基站需要送出的物资量立即增加  $v$ ，即  $w'_x = w_x + v$ 。
- **2 x**，表示立即向第  $x$  个基站的某个运输飞船下达工作命令，并会在极短的时间内完成此次任务。

L 希望他在每次向某个运输飞船下达工作命令时，你都能迅速给出当前飞船最终送往运输中心的资源总数。你能做到这点吗？

### 【输入格式】

从标准输入读入数据。

第一行包含一个正整数  $n$  ( $1 \leq n \leq 10^5$ )，表示航空基站的总数。

第二行有  $n$  个正整数，其中第  $i$  个整数表示第  $i$  个基站需要送出的物资量  $w_i$  ( $0 \leq w_i \leq 10^9$ )。

第三行有一个整数  $Q$  ( $1 \leq Q \leq 10^5$ )，表示一天内发生事件的总数。

接下来  $Q$  行依次表示当天依次发生的每个时间，其中每行的一个数字  $op$  ( $op \in \{1, 2\}$ ) 表示事件的类型，接下来：

- 如果  $op = 1$ ，则后面两个整数  $x, v$  ( $1 \leq x \leq n, 0 \leq v \leq 10^9$ ) 表示物资量发生变化的基站的编号的和增加的数目。
- 如果  $op = 2$ ，则后面一个整数  $x$  ( $1 \leq x \leq n$ ) 表示被下达工作命令的基站的编号。

**【输出格式】**

输出到标准输出。

输出若干行，对于每个  $op = 2$  的事件，输出一行一个整数表示答案。

**【样例 1 输入】**

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

**【样例 1 输出】**

```
1 14
2 22
3 18
```

## H. 回到过去 / Past

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

L 在搜集资料时意外发现了一页陈旧的笔记，笔记上歪歪斜斜画着一些简单的函数图像。他知道，这是他初中的时候学习二次函数记下的。

因为没有妥善保存，笔记本上用铅笔绘制的函数图像都变得模糊不清，只有其中  $n$  个整点能够被确定下来，其中第  $i$  个整点在二维平面上的坐标被确定为  $(x_i, y_i)$ 。

根据图像下的标注，L 能够确定这个函数图像对应的函数解析式一定属于以下类型之一：

- $y = a$ ，其中  $a$  为实常数，即常函数。
- $y = kx + b$ ，其中  $k, b$  为实常数且  $k \neq 0$ ，即一次函数。
- $y = ax^2 + bx + c$ ，其中  $a, b, c$  为实常数且  $a \neq 0$ ，即二次函数。

现在，给出这  $n$  个点的坐标信息，你能帮助 L 确定这个图像对应的函数解析式的类型吗？

### 【输入格式】

从标准输入读入数据。

输入第一行包含一个整数  $n$  ( $1 \leq n \leq 100$ )，表示 L 确定的点的数目。

接下来  $n$  行每行两个整数  $x_i, y_i$  ( $|x_i|, |y_i| \leq 10^{12}$ )，表示 L 确定的第  $i$  个点的坐标  $(x_i, y_i)$ 。

数据保证给定点的坐标互不相同且至少存在一条常函数、一次函数或者二次函数，使得给定的  $n$  个点都在函数图像上。

### 【输出格式】

输出到标准输出。

输出一行包含一个字符串。如果你能确定函数解析式为常函数，输出 **Const**；同理，若确定函数解析式为一次函数，输出 **Line**；若确定函数解析式为二次函数，输出 **Parabola**；特别地，如果不能确定函数解析式的类型，输出 **-1**。

### 【样例 1 输入】

```
1 4
2 0 1
3 1 3
```



```
4 2 5
5 4 9
```

**【样例 1 输出】**

```
1 Line
```

**【样例 1 解释】**

当函数解析式为  $y = 2x + 1$  时，此时对应的函数图像恰好能够经过 L 确定的 4 个点。

**【样例 2 输入】**

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

**【样例 2 输出】**

```
1 -1
```

**【样例 2 解释】**

满足题目条件的函数解析式有很多。

例如，它可能是常函数  $y = 1$ ；也可能是二次函数  $y = x^2$ ；同时，它也可能是二次函数  $y = 2 - x^2$ 。

因此我们不能够确定该函数解析式的类型。

## I. 穿越 / Stay

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

某天夜里，在 R 星基站的工作的 L 突然收到了一连串来自外太空的神秘信号。这一连串信号通过计算机处理后可以被转化为一串长度为  $4n$  的 01 序列  $S$  和一系列附加信息。

经过对这些附加信息的研究，L 认为这样一串 01 序列  $S$  极有可能可以通过如下转化规则最终转化为一个大写英文字符串  $T$ ：

- 设  $S$  的长度  $4n$ 。对于任意整数  $k$  满足  $0 \leq k < n$ ，取 01 字符串  $S$  中第  $4k+1$  至第  $4k+4$  项连成一个子串  $T$ ，设  $T$  在二进制下表示的数为  $x$ ，则我们确定英文字符串  $T$  的第  $k$  个英文字符为顺次第  $x+1$  个大写字母。
- 例如， $S = 000000010010$  经上述转化规则将会转化为英文字符串  $T = ABC$ 。

然而，一切并没有那么简单。根据记载，这类密文的制造者往往会对 01 序列进行一些处理。使得真正的答案成为给出的 01 序列  $S$  在至多  $k$  次交换其中任意相邻字符后，再通过上述规则所能够转化出的字典序最大的英文字符串。

你能帮助 L 破解出最终的英文字符串吗？

### 【输入格式】

从标准输入读入数据。

第一行包含两个整数  $n, k$  ( $1 \leq n \leq 10^5, 0 \leq k \leq 10^{12}$ )，分别表示字符串  $T$  的长度和至多交换次数  $k$ 。

第二行包含一个长度为  $4n$  的 01 字符串  $S$ ，仅由字符 **0** 和 **1** 组成。

### 【输出格式】

输出到标准输出。

输出仅一行一个长度为  $n$  字符串  $T$ ，表示你所破译出的英文字符串  $T$ 。

### 【样例 1 输入】

```
1 5 7
2 00111111101101000111
```

**【样例 1 输出】**

```
1 OLLEH
```

**【样例 2 输入】**

```
1 10 7
2 0101111111011000010100111101000011100110
```

**【样例 2 输出】**

```
1 PDNIFDNAOG
```

## J. 数数 / Counting

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

给定  $n, m$ 。

设  $A = \{1, 2, \dots, n\}$ ,  $S = \{(A_1, A_2, \dots, A_m) | A_1 \subseteq A_2 \subseteq \dots \subseteq A_m \subseteq A\}$ 。

现在，请你求出集合  $S$  中的元素个数。

由于答案过大，答案对  $10^9 + 7$  取模。

### 【输入格式】

从标准输入读入数据。

第一行包含一个正整数  $T$  ( $1 \leq T \leq 10^5$ ) 表示数据组数。

每组数据单独一行，包含两个正整数  $n, m$  ( $1 \leq n, m \leq 10^9$ )。

### 【输出格式】

输出到标准输出。

对每一组数据输出一行一个答案，表示集合  $S$  中的元素个数。

答案对  $10^9 + 7$  取模。

### 【样例 1 输入】

```
1 2
2 2 1
3 1 100
```

### 【样例 1 输出】

```
1 4
2 101
```

## K. 重逢 / Reunion

时间限制： 1.0 秒

空间限制： 512 MiB

## 【题目描述】

在二维平面  $xOy$  上有两个点 L 和 P，他们均存在一个以其当前所在位置为圆心，半径随时间变化的圆形信号区域。

初始时，L 点的坐标为  $(x_1, y_1)$ ，且其信号区域的半径为  $R_1$ ；P 点的坐标为  $(x_2, y_2)$ ，且其信号区域的半径为  $R_2$ 。

从  $t = 0$  时刻开始，L 开始朝向量  $\vec{n}_1 = (a, b)$  方向以  $1m/s$  的速度开始匀速运动（忽略加速时间）；P 开始朝向量  $\vec{n}_2 = (c, d)$  方向同样以  $1m/s$  的速度开始匀速运动。于此同时，点 L 与点 P 的信号区域的半径也都在以  $1m/s$  的速度匀速扩展。

在整个运动过程中，若 L 点与 P 点的信号区域在某一时刻存在重叠（包含两个信号区域边界外切的情况），我们则认为二者成功建立信号链接。

现在，给定 L 点与 P 点的初始状态和运动方向，请问他们是否能在某个时刻成功建立信号链接？如果可以，他们建立链接的最早时间是多少？

## 【输入格式】

从标准输入读入数据。

第一行包含三个整数  $x_1, y_1, R_1$  ( $|x_1|, |y_1| \leq 10^4, 0 \leq R_1 \leq 2 \times 10^4$ )，分别表示初始时点 L 的横坐标、纵坐标以及信号区域的半径。

第二行包含三个整数  $x_2, y_2, R_2$  ( $|x_2|, |y_2| \leq 10^4, 0 \leq R_2 \leq 2 \times 10^4$ )，分别表示初始时点 P 的横坐标、纵坐标以及信号区域的半径。

第三行包含四个整数  $a, b, c, d$  ( $|a|, |b|, |c|, |d| \leq 100$  且  $a^2 + b^2 \neq 0, c^2 + d^2 \neq 0$ )，代表点 L 和点 P 运动方向的向量  $\vec{n}_1 = (a, b)$  与  $\vec{n}_2 = (c, d)$ 。

## 【输出格式】

输出到标准输出。

输出仅一行。

- 如果能够成功建立信号连接，输出一个有理数表示他们建立链接的最早时间。题目保证最终输出的答案小于  $10^9$ 。
- 如果始终无法成功建立信号链接，输出  $-1$ 。
- 在本题中，当你的答案与标准答案的绝对误差或者相对误差在  $10^{-4}$  以内时即被视为正确。即，假设你得到的答案为  $a$  而标准答案为  $b$ ，你的答案被视作正确当且仅当  $\frac{|a-b|}{\max\{1, |b|\}} \leq 10^{-4}$ 。

## 【样例 1 输入】

```

1 0 2 1
2 2 0 1
3 0 1 1 0

```

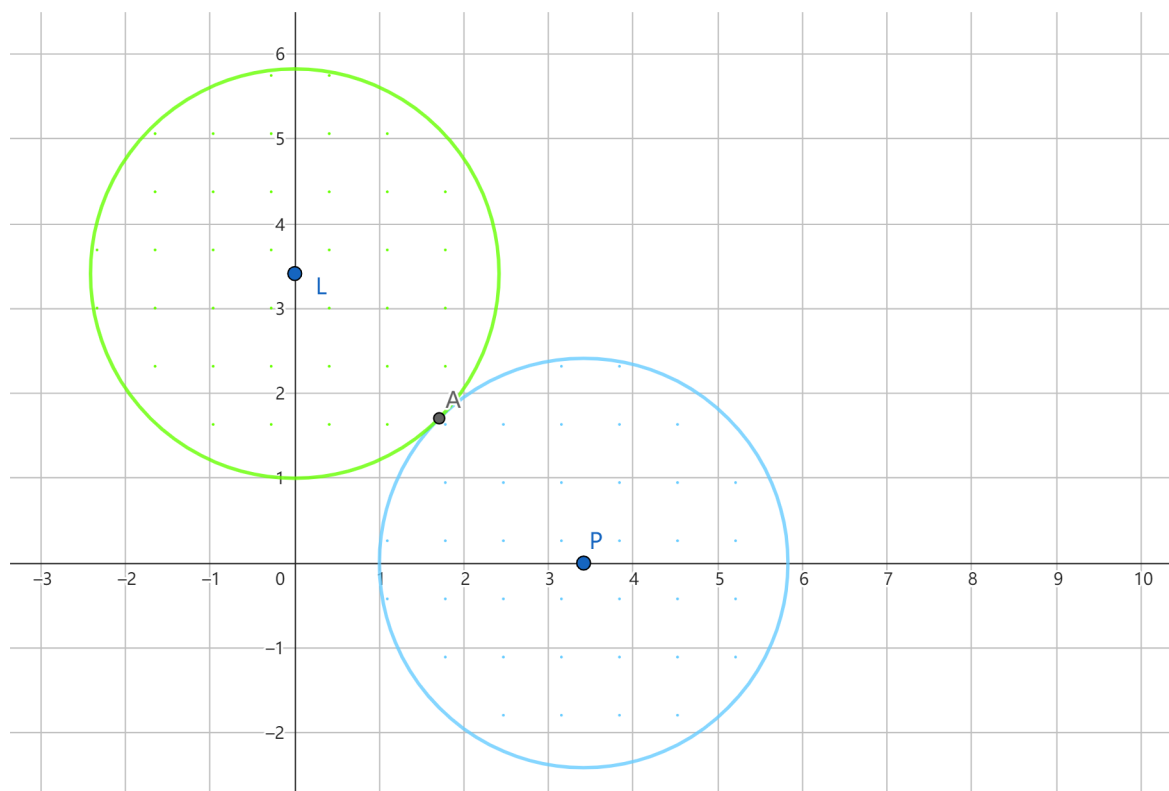
## 【样例 1 输出】

```

1 1.4142135624

```

## 【样例 1 解释】



如图，当  $t = \sqrt{2}$  时，此时 L 点的坐标为  $(0, 2 + \sqrt{2})$ ，P 点的坐标为  $(2 + \sqrt{2}, 0)$ 。此时两点的信号区域的边界正好处于外切的情况，故此时成功建立了信号链接。容易证明  $t < \sqrt{2}$  时两点无法建立信号链接。

## 【样例 2 输入】

```

1 -2 0 2
2 2 0 1
3 -1 0 2 0

```

**【样例 2 输出】**

1 -1

## L. 紧急支援 / Emergency

时间限制：2.0 秒

空间限制：512 MiB

### 【题目描述】

灾难来临后，C 国的  $n$  座城市陷入危机。现在，它们急需进行资源补助。

C 国的  $n$  座城市存在  $n - 1$  条传输路径。他们构成了一个完整的树形结构，也就是说，通过这些路径，C 国的  $n$  座城市可以相互联通。其中，第  $i$  条传输路径可以被描述为  $(u_i, v_i)$ ，表示一条可在  $u_i$  号城市与  $v_i$  号城市双向传输的路径。得益于 C 国先进的科学技术，补助在传输路径上耗费的时间可以忽略不计。

经过下级干员的统计，第  $i$  座城市需要的补助点数为  $V_i$ 。为了使得每座城市都能够得到相应数目的补助，救援队计划采取如下策略进行补助发放：

- 首先，救援队会从 1 号城市以每单位时间发放 1 补助点数的速度开始发放补助。
- 每个城市会按照相同的策略处理收到的补助：对于每个接收到补助的城市  $x$ ，若其儿子节点所代表的城市中还有未补助完全的城市，则其会从所有未补助完全的儿子中等概率选取一个，并持续地通过传输路径向其传输补助，直至其补助点数达到要求为止，然后再等概率挑选下一个传输补助的城市，以此类推；若其不存在儿子节点或儿子节点所代表的城市都已补足完全，则将收到的补助置入自己城市的仓库中。以上处理过程都相当迅速，我们可以将这些过程所耗费的时间都忽略不计。
- 当所有城市都已补助完全时，救援队会立即停止向 1 号城市发放补助。

现在，你需要对每个城市  $i$  求出从整个补助行动开始到该城市补助完全的期望时间  $t_i$ 。

### 【输入格式】

从标准输入读入数据。

第一行包含一个整数  $n$  ( $2 \leq n \leq 5 \times 10^5$ )，表示 C 国受灾的城市数目。

第二行包含  $n$  个正整数，其中第  $i$  个正整数  $V_i$  ( $1 \leq V_i \leq 10^3$ ) 表示第  $i$  座城市所需的补助点数的大小。

接下来  $n - 1$  行每行包含两个整数  $u_i, v_i$  ( $1 \leq u_i, v_i \leq n$ )，分别描述每条传输路径  $(u_i, v_i)$ 。

数据保证给出的  $n - 1$  条传输路径可以使得这  $n$  座城市相互联通。

### 【输出格式】

输出到标准输出。



输出一行  $n$  个有理数，其中第  $i$  个数表示从整个补助行动开始到第  $i$  座城市补助完全的期望时间  $t_i$ ，保留 2 位小数。

**【样例 1 输入】**

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

**【样例 1 输出】**

```
1 9.00 7.50 4.50 3.00 3.50 5.00 3.50
```

## M. 流动的城市 / Flow

时间限制：2.0 秒

空间限制：512 MiB

### 【题目背景】

小 P 喜欢雨中的城市。找一个无人的窗台安静地俯瞰整座城市。水流、人流、车流……仿佛整座城市都在流动。

### 【题目描述】

C 市的城市公交地铁布局可以由一个  $n \times m$  的网格图来描述：

- 每个网格图上的格点  $(i, j)$  都是一个办公区、商业区或者住宅，其都有可能是居民每天出行的出发点或者目的地。
- 对于任意  $(i, j) \in [1, n] \times [1, m]$ ，都存在至少一条花费为  $a_i$  的公交线路使得居民能够从  $(i, j)$  到达  $(i + 1, j)$  或从  $(i + 1, j)$  到达  $(i, j)$ 。
- 对于任意  $x \in [1, n]$ ，都存在一条贯穿第  $x$  行的地铁线路，使得乘客能够从第  $x$  行的任意一个格点  $(x, i)$  出发，到达第  $x$  行的另外一个格点  $(x, j)$ ，且花费恒为  $b_x$ 。

现在，你需要回答 C 市居民的  $Q$  个问题，每个问题包含四个数据  $sx_k, sy_k, tx_k, ty_k$ ，表示如果从格点  $(sx_k, sy_k)$  出发，计划到达  $(tx_k, ty_k)$ ，他所需要的最小花费为多少？

### 【输入格式】

从标准输入读入数据。

第一行包含两个整数  $n, m$  ( $2 \leq n, m \leq 2 \times 10^5$ )，分别表示网格图的行数与列数。

第二行包含  $n - 1$  个正整数  $a_1, a_2, \dots, a_{n-1}$  ( $1 \leq a_i \leq 10^9$ )，其中第  $i$  个正整数表示从第  $i$  行到第  $i + 1$  行的公交线路的费用。

第三行包含  $n$  个正整数  $b_1, b_2, \dots, b_n$  ( $1 \leq b_x \leq 10^9$ )，表示每行地铁线路的费用。

第四行包含一个整数  $Q$  ( $1 \leq Q \leq 3 \times 10^5$ )，表示询问的个数。

接下来  $Q$  行每行四个正整数  $sx_k, sy_k, tx_k, ty_k$  ( $1 \leq sx_k, tx_k \leq n, 1 \leq sy_k, ty_k \leq m$ )，表示在本次询问中，你需要回答从  $(sx_k, sy_k)$  出发到达  $(tx_k, ty_k)$  所需要的最小花费。

### 【输出格式】

输出到标准输出。

输出包含  $Q$  行，每行一个整数，依次表示每次询问的答案。

### 【样例 1 输入】

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

**【样例 1 输出】**

```
1 12
2 9
3 10
```

## N. 浸泡！脱水！ / Cube

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

L 正在测算完成本次旅程所需材料的强度。因此，他来到加工站。

加工站中有一个由若干  $1 \times 1 \times 1$  的小正方体零件所组成的  $a \times b \times c$  的长方体。

L 需要对这个长方体进行加工，加工过程中需要挖去恰好  $k$  个  $1 \times 1 \times 1$  的正方体零件。挖去之后，工厂的工作人员将会将其浸泡在红色染料（一种流体）之中，由于正方体零件之间连接相当稳固，只有能够与红色染料接触且未与任何另外的正方体零件接触的面才会被染上红色。

将成品从染料中拿起脱水之后，工厂的工作人员会根据成品的染色的面的情况统计出一个 7 元组  $(c_6, c_5, c_4, \dots, c_0)$ ，其中  $c_i$  表示有  $i$  个面被染上红色染料的正方体零件个数。

L 认为，最终的成品稳固性最佳，当且仅当这个 7 元组的字典序最小。那么，当最终的成品稳固性最佳时，这个七元组是怎样的呢？

### 【输入格式】

从标准输入读入数据。

仅一行，包含 4 个正整数  $a, b, c, k$  ( $2 \leq a, b, c \leq 10^9, 0 \leq k \leq \min\{3 \times 10^5, (a-2) \times (b-2) \times (c-2)\}$ ) 依次表示长方体的长、宽、高，以及挖去的小正方体的个数。

### 【输出格式】

输出到标准输出。

输出一行 7 个非负整数，依次表示  $c_6, c_5, \dots, c_0$ 。

### 【样例 1 输入】

```
1 3 3 3 1
```

### 【样例 1 输出】

```
1 0 0 0 8 12 6 0
```

## O. 最尴尬的一集 / Awkward

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

转眼十余年过去，小 P 已经从当初蹒跚学步的小孩成长为了今天朝气蓬勃的少年，曾经倒在他成长路上的无数挑战也都被他迎刃而解，但有一个坏毛病却始终困扰着他：社交恐惧症。

今天，收到某大厂面试的小 P 一大早匆匆赶到现场。当他走入等候室时他就傻眼了： $n + 2$  个椅子整齐向前排列成一竖列，同样来参加面试的选手们歪歪斜斜地坐在椅子上，几乎每个空位的旁边边都有人！

早已社恐晚期的小 P 瞬间没有了找个座位坐下的念头，便往后一退，凝视着这一令他尴尬至极的场面。

小 P 发现，接下来走进等候室坐下的人可以恰好分成两类，小 P 把他们分别称为“社牛”和“社恐”。

对于“社牛”，他们总是径直走向当前最靠左的一个座位并坐下；而“社恐”则会先认真计算每个空位  $i$  距离其最近的一个有人的位置  $j$  的距离（定义该距离的值为  $|i - j|$ ），在所有距离最大的空位当中，他会选择最靠左的那个位置坐下。

百无聊赖的小 P 正好考考你：假定从左到右每个座位的编号为  $0 \sim n + 1$ ，且一开始只有 0 号座位和  $n + 1$  号座位上有人。现在有  $m$  个人先后进入等候室并落座。你能够在每个人落座时迅速地回答他所坐下的座位的编号吗？

### 【输入格式】

从标准输入读入数据。

第一行包含两个正整数  $n, m$  ( $1 \leq n \leq 10^9, 1 \leq m \leq 10^5$  且  $m \leq n$ )，分别代表最初空置的座位数目和先后进入等候室的人数。

接下来一行一个长度为  $m$  的 01 字符串表示这  $m$  个人的种类信息。设其中第  $i$  个字符为  $c_i$ ，则：

- 若  $c_i = 0$ ，则表示第  $i$  个走进等候室的人是一个“社恐”；
- 若  $c_i = 1$ ，则表示第  $i$  个走进等候室的人是一个“社牛”；

### 【输出格式】

输出到标准输出。

输出  $m$  行，每行一个正整数  $x_i$ 。其中第  $i$  行代表第  $i$  个人坐下的座位的编号。

**【样例 1 输入】**

```
1 5 5
2 11000
```

**【样例 1 输出】**

```
1 1
2 2
3 4
4 3
5 5
```

## P. 轮回的房间 / Loop

时间限制：1.0 秒

空间限制：512 MiB

### 【题目描述】

这是一个长为  $n$  宽为  $m$  的长方形房间，周围的墙壁全部被光滑平整的玻璃包围，除此之外，房间中没有任何摆放的物件。因此，我们可以将整个房间抽象成一个二维平面，以长方形的一角为原点，长与宽所在直线分别为  $x$  轴与  $y$  轴建立直角坐标系，使得整个房间内部均落在第一象限内。

小 P 知道，他永远走不出这间房间——至少他是这样认为。因此，他打算在这个房间之中进行一项有趣的试验。

首先，他在这个房间内选择了一个整点  $A(x_0, y_0)$ 。随后，他朝着向量  $\vec{n} = (a, b)$  的方向射出一束水平光线。假设光的能量在传播中不会衰减且忽略小 P 对光线的阻挡，可以预料到的是，整个房间内的某些位置将会出现明亮的光束。现在小 P 想知道，房间内可见光束的总长度是多少？

需要注意的是，根据光的反射定律，光线在径直射向房间的某个角落时，将会原路返回。

### 【输入格式】

从标准输入读入数据。

输入一行六个整数  $n, m, x_0, y_0, a, b$ ，分别表示房间的长和宽、点  $A$  的横坐标及纵坐标、向量  $\vec{n}$  的水平方向大小及竖直方向大小。

数据保证， $2 \leq n, m \leq 10^8, 0 \leq |a|, |b| \leq 10^2, a^2 + b^2 \neq 0, 0 < x_0 < n, 0 < y_0 < m$ 。

### 【输出格式】

输出到标准输出。

输出一行一个实数，表示答案。

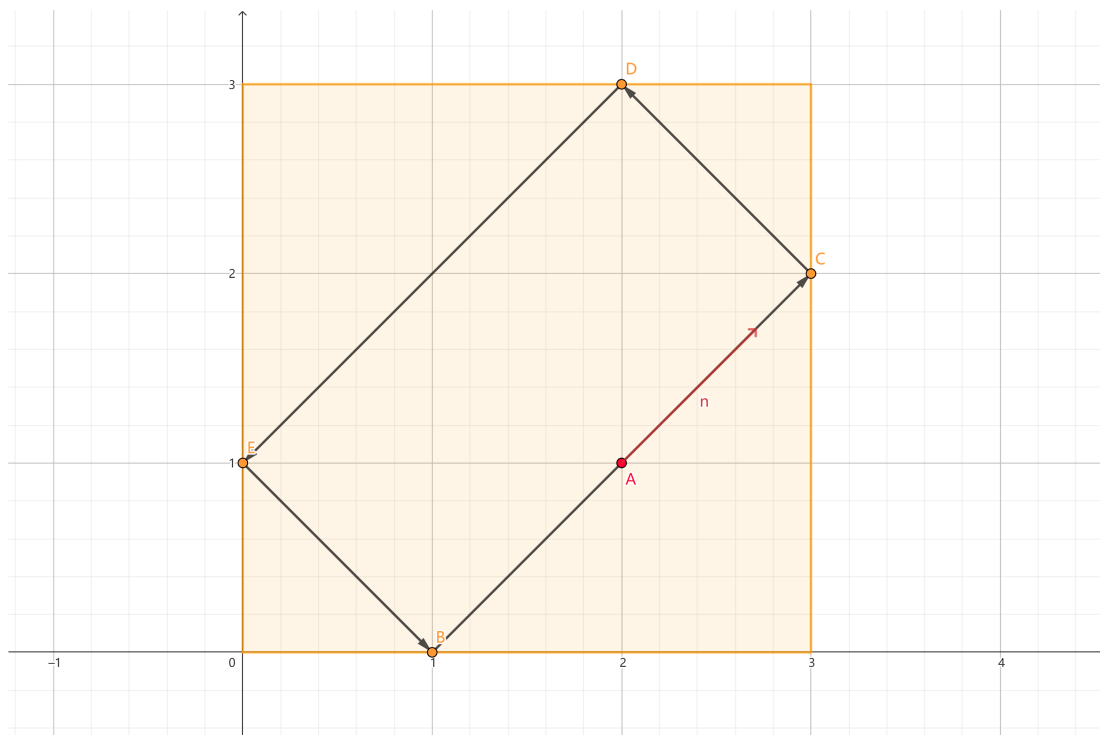
在本题中，当你的答案与标准答案的绝对误差或者相对误差在  $10^{-4}$  以内时即被视为正确。即，假设你得到的答案为  $a$  而标准答案为  $b$ ，你的答案被视作正确当且仅当  $\frac{|a-b|}{\max\{1, |b|\}} \leq 10^{-4}$ 。

### 【样例 1 输入】

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

**【样例 1 输出】**

```
1 8.4852813742
```

**【样例 1 解释】**

房间内可见光束的分布如上图所示，容易求得此时光束的总长度为  $L = 6\sqrt{2}$ 。

**【样例 2 输入】**

```
1 8 4 3 2 1 -2
```

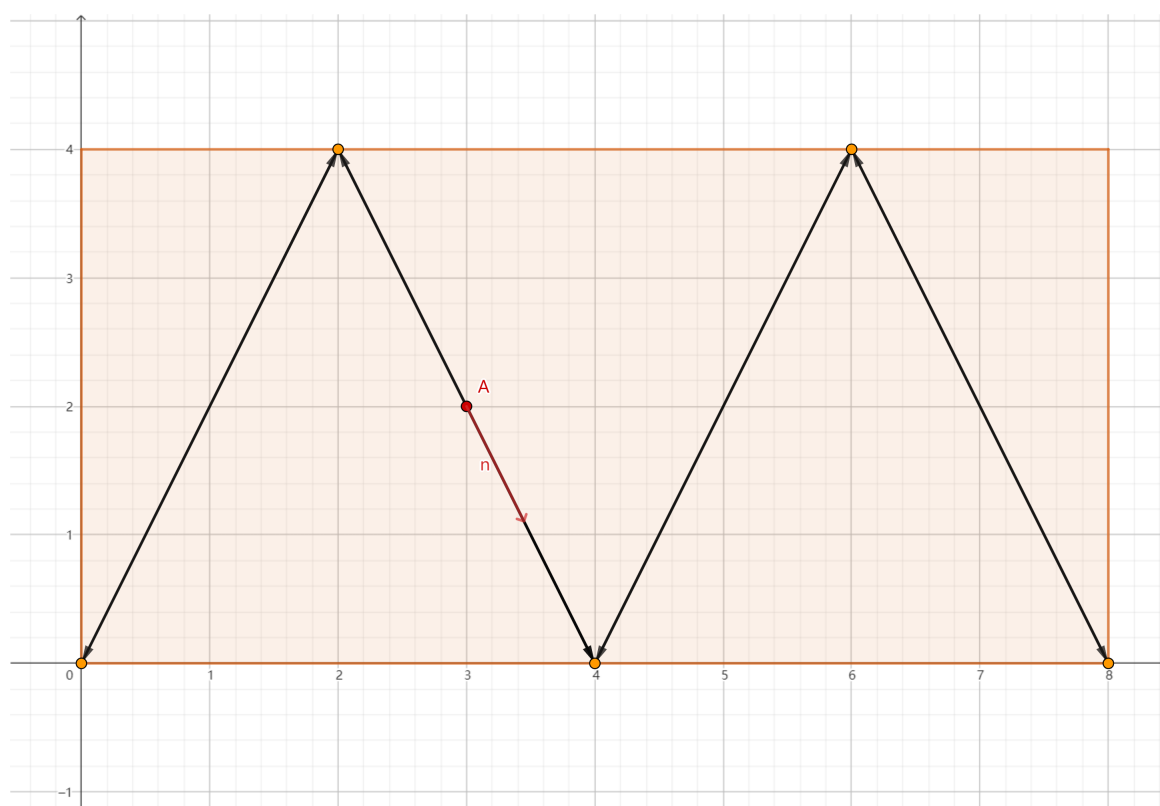
**【样例 2 输出】**

```
1 17.8885438200
```

**【样例 2 解释】**

此时整个光束恰好落在一条折线段上，容易求得该折线段长度  $L = 8\sqrt{5}$ 。





## Q. 彗星坠落的那一天 / Comet

时间限制：2.0 秒

空间限制：512 MiB

### 【题目背景】

我在沉睡中重生/窗外黎明/变黄昏/梦在将醒未醒时/成真

### 【题目描述】

2013 年 6 月 24 日凌晨 1 点 32 分，C 国的科学家们突然发现原本预计于明天白天掠过地球的  $n$  颗元素互不相同的魔法石突然改变了轨迹，正以难以想象的速度径直撞向地球。

据现有数据可知，撞击地球的第  $i$  个魔法石的危险系数为  $a_i$ ，其恰好由第  $i$  种元素制成。

研究发现，第  $i$  种元素拥有一个映射对象  $f_i$  和映射模式  $x_i$  两个重要参数，在魔法石坠落地球之前，每种元素都会依据这两个重要参数发生一次吸引关系，从而使得某些魔法石发生合并。

具体而言，对于任意一个元素  $i$ ，它的吸引关系会在某个未知的时间点生效。接下来发生的事件将会是如下情形之一：

- 如果元素  $i$  与元素  $f_i$  恰好在包含在一个魔法石中，该吸引关系不会对任何魔法石的危险系数和所包含的元素种类产生实质性变化。
- 如果元素  $i$  与元素  $f_i$  分别包含在不同的魔法石  $x$  与  $y$  中，巨大的吸引力会将两个魔法石拉扯在一起，使其产生猛烈的撞击，最终形成一个新的魔法石  $z$ 。新的魔法石  $z$  会继承撞击前的两颗魔法石的所有元素；同时，新魔法石的危险系数将会由如下规则决定：
  - 若元素  $i$  的映射模式  $x_i = 0$ ，假设撞击前的两个魔法石的危险系数分别为  $p$  和  $q$ ，那么撞击产生的新魔法石的危险系数就是  $p$  or  $q$ ；
  - 反之，若元素  $i$  的映射模式  $x_i = 1$ ，那么撞击产生的新魔法石的危险系数则是  $p$  and  $q$ 。
  - 其中，or 是按位或运算，and 是按位与运算。

现在，想要避免撞击的发生已经不可能了。科学家们唯一能做的便是估计最终撞击地球的魔法石的危险系数，并以此制定最为合理的抗灾措施，保证更多人的生还。

然而，科学家至今无法预测每种元素对应的吸引关系生效的准确时间，只能保证  $n$  种吸引关系两两之间不会同时发生。因此，科学家希望你在所有可能  $n!$  种吸引关系生效的时间顺序之中，找到吸引关系全部生效后所形成的魔法石危险系数之和的最大值。时间紧急，他们希望你能尽快给出答案。

**【输入格式】**

从标准输入读入数据。

第一行包含一个正整数  $n$  ( $2 \leq n \leq 5 \times 10^5$ ), 表示撞击地球的魔法石的数目。

第二行包含  $n$  个整数, 代表最初每个魔法石的危险系数  $a_i$  ( $0 \leq a_i < 2^{30}$ )。

接下来  $n$  行, 每行包含两个整数  $f_i, x_i$  ( $1 \leq f_i \leq n$ ,  $f_i \neq i$  且  $x_i \in \{0, 1\}$ ), 分别代表每种元素的映射对象和映射模式。

**【输出格式】**

输出到标准输出。

输出仅一行一个正整数, 表示可能的吸引关系全部生效后魔法石危险系数之和的最大值。

**【样例 1 输入】**

```
1 5
2 3 1 4 6 8
3 2 1
4 3 0
5 4 1
6 1 1
7 3 0
```

**【样例 1 输出】**

```
1 13
```

**【样例 1 解释】**

下面给出当魔法石危险系数之和取到最大值时, 一种可能的吸引关系发生顺序:

- 元素 3 与其映射对象 4 首先生效, 对应魔法石撞击后产生了一颗魔法值为  $4 \text{ and } 6 = 4$  的新魔法石。
- 随后, 元素 1 与其映射对象 2 生效, 对应魔法石撞击后产生了一颗魔法值为  $3 \text{ and } 1 = 1$  的新魔法石。
- 元素 5 与其映射对象 3 生效, 对应魔法石撞击后产生了一颗魔法值为  $8 \text{ or } 4 = 12$  的新魔法石。

- 元素 4 与其映射对象 1 生效，由于此时两元素已在同一魔法石内，故对应魔法石无实质性变化。
- 最后，元素 2 与其映射对象 3 生效，对应魔法石撞击后产生了一颗魔法值为  $1 \text{ or } 12 = 13$  的新魔法石。