

# 第三届环球杯



## 第 8 阶段：仓前

2024 年 8 月 31 日至 9 月 1 日

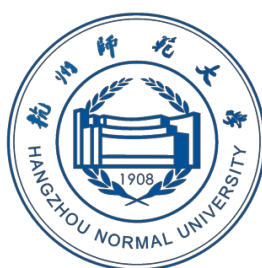
该问题集应包含 13 个问题（A 至 M），共 23 页。

### 根据



国际大学生程序设计竞赛（ICPC）

### 主办方







# 问题 A. 宾果游戏

时间限制                      2 秒 内存限制  
1024 兆字节

你还记得  
小学? 这里有一个更难的版本。

给你两个正整数  $n$  和  $m$ ，求  $x > n$  且  $x$  是好数的最小整数  $x$ 。好数  $x$  满足  $x \equiv 0 \pmod m$  或  $x$  包含  $m$  作为十进制表示的子串。

例如，当  $m = 3, n = 7$  时， $x = 9$  是答案，因为  $9 \equiv 0 \pmod 3$ 。当  $m = 3, n = 12$  时  $x = 13$  是答案，因为 13 包含子串 3。

## 输入

第一行包含一个整数  $T$  ( $1 \leq T \leq 10^5$ )，代表测试用例的数量。

对于每个测试用例，一行包含两个整数  $n$  和  $m$  ( $1 \leq n < 10^{10^6}, 1 \leq m \leq 10^9$ )。  $n$  不包含前导零。

保证  $\sum [\log_{10}(n)] \leq 3 \times 10^6$

## 输出

对于每个测试用例，输出一行，其中包含一个整数  $x$ ，代表答案。

## 示例

标准输入	标准输出
6	9
7 3	13
12 3	10
9 10	251
249 51	1370
1369 37	3
2 1	



## 问题 B. 模拟宇宙

时间限制

1 秒 内存限制

1024 兆字节

"星轨"混合了两种传统的逻辑谜题类型："星际争霸"和"轨道池"。

- 星辰铁路，中巴 14

糖糖正在玩他最喜欢的游戏《鸿海：星轨》。他正在玩一种名为"模拟宇宙"的游戏模式。

游戏玩法围绕着收集被称为"祝福"（Blessings）和"古玩"（Curios）的升级和神器来增强团队的力量，并完成包含难度逐渐增加的敌人的阶段。

在每次闯关开始时，玩家可以选择一条路径，这将赋予路径特定的 BUFF，并增加获得相应路径祝福的几率。在闯关过程中，玩家将完成不同类型的领域（战斗、发生、交易、遭遇、精英、喘息），最终目标是到达最终的 Boss 领域并击败 Boss。

祝福会通过各种途径积累，为团队提供各种被动效果和 BUFF。祝福可以通过随机事件或在"喘息之域"中升级，以增强其效果。玩家还将获得一种名为"宇宙碎片"的货币，可在某些随机事件中使用或用于升级祝福值。

糖糖是在低难度模式下进行游戏的，因此场景已经大大简化。**我们强烈建议您仔细阅读以下内容，尤其是对游戏非常熟悉的玩家。提示中描述的游戏模式场景与原版游戏有较大差异，请仔细辨别。**本闯关共有  $n$  个领域，每个领域的类型都是发生。每个领域都会提供以下两种奖励中的一种：

- 祝福糖糖获得一个祝福，他的攻击值立即增加 1。如果糖糖至少拥有 1 个宇宙碎片，他可以选择用 1 个宇宙碎片升级这个祝福，升级后的祝福可以使他的攻击值增加 2。一个祝福只能升级一次。
- 古玩苏格获得一件古玩。古玩有两种类型，只能选择其中**一种**。在第  $i$  个领域获得的第一种古玩可以升级不超过  $i$  在第  $i$  个领域之前获得的未升级祝福，而在第  $i$  个领域获得的第二种古玩可以立即提供  $b_i$  宇宙碎片。

请注意，宇宙碎片**不能**用于升级苏格已经获得的祝福。它只能用来升级提供祝福奖励的领域中的祝福，而且只能用来升级他在该领域中获得的祝福。而第一种古玩**只能**升级苏格已经获得的祝福



。

糖糖在游戏开始时只有 0 个祝福值、0 个宇宙碎片和 0 个攻击值。请帮他计算通过  $N$  个领域后的最大攻击值。

## 输入

输入包含多个测试用例，第一行包含一个整数  $t$  ( $1 \leq t \leq 10^3$ )，表示测试用例的数量。

对于每个测试用例，第一行包含一个整数  $n$  ( $1 \leq n \leq 8 \cdot 10^3$ )，表示域的数量。下面  $n$  行中的第  $i$  行以单字符  $t_i$  ( $t_i \in \{B, C\}$ ) 开始，表示奖励类型



的域，是祝福还是古玩。如果  $t_i = C$ ，则后面会出现两个整数  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n$ )，以上述格式表示 Curio。

保证所有测试用例的  $n$  之和不超过  $8 \cdot 10^3$ 。

输出

对于每个测试案例，在一行中输出一个整数，表示糖糖通过  $n$  个域后的最大攻击值。

示例

标准输入	标准输出
2	2
2	8
B	
C 1 1	
6	
B	
B	
C 2 1	
C 1 2	
B	
B	



# 问题 C. 挑战 NPC

时间限制 1 秒 内存限制 256 兆字节

*Sugar 是一个基于 SAT 的约束求解器。约束满足问题 (CSP) 被编码为布尔 CNF 公式，并由外部 SAT 求解器求解。*

- Sugar：基于 SAT 的约束求解器

众所周知，求图的色度数是一个 NPC 问题。然而，小塔尔延声称，他可以用一种简单的贪婪算法来解决这个问题：

为 1 到  $n$  个顶点着色。

$\{col_v \mid v < u, (v, u) \in E\}$  到  $col_u$ ，其中  $E$  是图的边集。例如， $MEX(\{1, 1, 2, 4\}) = 3$ ， $MEX(\emptyset) = 1$ 。

你想证明这种贪婪算法是完全错误的。请构建这样一个图形：你可以用  $c$  种颜色给这个图形着色，但贪心算法至少会用  $c + k$  种颜色给这个图形着色。

## 输入

唯一一行包含一个整数  $k$  ( $1 \leq k \leq 500$ )。

## 输出

第一行包含三个整数  $n$ 、 $m$  和  $c$  ( $1 \leq n \leq 1024$ ， $0 \leq m \leq \frac{n(n-1)}{2}$ ， $1 \leq c \leq n$ )，分别代表顶点数、边数和可用于为该图着色的颜色数。

下面一行包含  $n$  个整数  $col_1, col_2 \dots col_n$  ( $1 \leq col_i \leq c$ )，表示您的着色。

下面的  $m$  行分别包含两个整数  $u, v$  ( $1 \leq u, v \leq n$ ， $u \neq v$ ， $col_u \neq col_v$ )，代表你的图形中的一条边。

如果有多个解决方案，则输出任何一个。

## 示例

标准输入	标准输出
1	4 3 2 1 2 2 1



第三届环球杯

第八阶段：仓前，2024 年 8 月 31 日至 9 月 1 日

	1 2
	2 4
	3 4





# 问题 D. 谜题：像拼字游戏一样简单

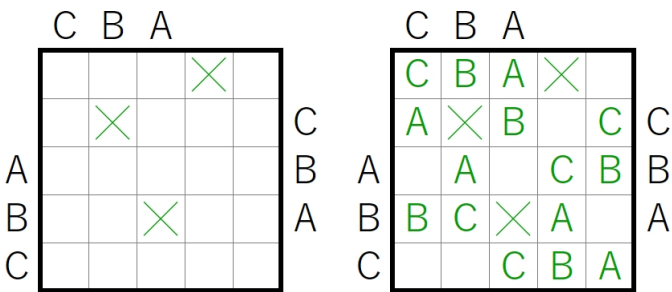
时间限制 1 秒 内存限制 1024 兆字节

很多谜题的变体往往比相应类型的谜题更难。但矛盾的是，就像一个单词加上两个字母后会变得更短一样，有一种变体会让任何类型的谜题变得更容易。

- 弗雷迪-汉德

格莱美是一位谜题大师。今天，她要玩的是 "拼字游戏 "的变体--"像拼字游戏一样简单"。谜题由一个  $n \times m$  的网格组成。每个单元格都要填上一个大写字母，或者留空。

谜题之外还有一些线索。外部线索表示该方向第一个非空单元格中的字母。此外，一些单元格中可能包含 "x" 标记，表示该单元格应该是空的。



例如，左边的图片展示了一个未解的谜题，右边的图片展示了谜题的解法。

格莱美希望找到谜题的答案。请帮助格莱美找出谜题的答案，或者报告没有答案。

## 输入

第一行包含 2 个整数  $n, m$  ( $1 \leq n, m \leq 1000$ )，分别表示网格的行数和列数。

下一行包含一个点 (".")，接着是  $m$  个字符  $U_i$ ，表示网格上方的线索，然后是一个点 (".")。

接下来的  $n$  行中，每一行都包含一个线索  $L_i$ ，然后是  $m$  个字符  $c_{ij}$ ，接着是一个线索  $R_i$ ，分别表示网格左边的线索、网格单元格和网格右边的线索。



下一行包含一个点（"."），接着是  $m$  个字符  $D_i$ ，表示网格下方的线索，后面是一个点（"."）。

每个线索  $U_i, L_i, R_i, D_i$  要么是一个大写英文字符，要么是一个点（"."）。每个中心单元  $c_{ij}$  要么是一个 "x"，要么是一个点（"."）。

输入中的所有点表示相应位置为空。



输出

如果解决方案不存在，则单行输出 "NO"。

否则，在第一行输出 "是"，然后输出  $n$  行，每行包含  $m$  个字符，表示谜题的答案。每个空单元格都应以圆点表示。如果有多个解法，则输出任意一个。

请注意，您应将中心单元格中的 "x " 字符替换为圆点。还请注意，不应输出线索单元格。

实例

标准输入	标准输出
5 5 .CBA... ....x.. .x...C A.....B B..x..A C..... .....	是 CBA... A.B.C .A.CB BC.A. .CBA
1 2 .... Nx.. . ..O.	没有



## 问题 E. 团队安排

时间限制

1 秒 内存限制

512 兆字节

大多数教师的误解： $w_i$  是一个  
严格的递增序列。

- 鸡大师，  
薄师傅

有  $N$  名学生在上薄老师的算法课。薄老师要求学生们进行团队合作，然后帮助他们分成小组。

每个学生必须正好属于其中一个小组。波师傅非常了解他的学生，知道只有当第  $i$  个学生所在小组的人数不少于  $l_i$  且不多于  $r_i$ （包括他自己）时，他才会对分班感到满意。请注意，一个小组可以正好由一名学生组成。

您将得到  $n$  个整数  $w_1, w_2, \dots, w_n$ 。假设最后有  $m$  个小组，其中第  $i$  个小组由  $c_i$  名学生组成，则这样安排的**权重**为  $w_{c_1} + w_{c_2} + \dots + w_{c_m}$ 。

薄师傅现在想知道，怎样把学生分成小组，才能让每个学生都满意，安排的**权重**最大。请你编写一个程序来帮助薄老师。

### 输入

输入的第一行包含一个整数  $n$  ( $1 \leq n \leq 60$ )，表示学生人数。

在接下来的  $n$  行中，第  $i$  行包含两个整数  $l_i$  和  $r_i$  ( $1 \leq l_i \leq r_i \leq n$ )，描述第  $i$  个学生。

下一行包含  $n$  个整数  $w_1, w_2, \dots, w_n$  ( $|w_i| \leq 10^7$ )。

### 输出

打印一行，其中包含一个整数：重量的最大可能值。如果不可能找到这样的排列，请打印 "不可能"。



实例

标准输入	标准输出
3 2 3 1 2 2 2 4 5 100	9
3 1 3 3 3 2 3 1 1 100	100
2 1 1 2 2 1 1	不可能
3 2 3 1 2 2 2 -100 -200 100000	-300

备注

最后一个例子说明了为什么 ICPC 需要三人小组。



## 问题 F. 阶段Agausscrab

时间限制

1 秒 内存限制

256 兆字节

在?:??或更短的时间内解决这  
个问题，就能得到一个高斯蟹  
!

- 动物团队

臭名昭著的问题设置小组 "Animal Crew "刚刚为 "独角兽冲天炉 "准备了一个问题集，这将是本次比赛的最新阶段。UniCup 委员会刚刚收到了问题设置者的名单和他们各自设置的问题数量。他们决定用以下规则来命名这个赛段：

- 假设有  $n$  个问题设置者。第  $i$  个问题设置者的名字是由几个小写拉丁字母组成的字符串  $s_i$ ，该人设置的问题数是  $a_i$ 。
- 委员会首先计算每个问题设置者的排名。第  $i$  个问题设置者的排名  $r_i$  的定义是：1 加上**严格**设定比此人问题**更多**的人数。
- 从第 1 个问题设置者到第  $n$  个问题设置者，删除第  $i$  个问题设置者名称中的最后  $r_i$  个字符，并将其连接起来形成字符串  $t$ 。如果第  $i$  个问题设置者名称中的字符数不超过  $r_i$ ，则删除所有字符。
- 最后，将  $t$  的第一个字符大写，这就是舞台的名称。

更多解释请参阅 "备注 "部分。

### 输入

第一行包含一个整数  $n$  ( $1 \leq n \leq 1000$ )，表示测试用例的数量。

第  $i$  行包含一个字符串  $s_i$  ( $2 \leq |s_i| \leq 20$ ) 和一个整数  $a_i$  ( $1 \leq a_i \leq 10$ )，表示第  $i$  个问题设置者的姓名和他/她设置的问题数。

### 输出

输出字符串 "Stage: "作为开头，然后输出空格，最后输出比赛名称。这些字符串应在一行中输出。



实例

标准输入	标准输出
4 弧形 2 gausr 5 废纸 3 bei 3	舞台Agausscrab
4 zhe 1 江 3 sheng 5 西 2	舞台甲申



## 备注

在第一个例子中，有 4 个问题设置者，他们分别设置了一个  $\tau_1 = 2$ 、一个  $\tau_2 = 5$ 、一个  $\tau_3 = 3$ 、一个  $\tau_4 = 3$  个问题，那么它们的等级分别是  $r_1 = 4$ 、 $r_2 = 1$ 、 $r_3 = 2$ 、 $r_4 = 2$ 。

从  $s_1$  中删除最后一个  $r_1 = 4$  个字符后，得到的字符串是 "a"。从  $s_2$  中删除

最后一个  $r_2 = 1$  个字符后，得到的字符串为 "gaus"。从  $s_3$  中删除最后一个

$r_3 = 2$  个字符后，得到的字符串为 "srav"。从  $s_4$  中删除最后一个  $r_4 = 2$  个

字符后，得到的字符串为 "b"。

字符串  $t$  是结果字符串 "agausscrab" 的连接。经过最后一步，我们可以得到舞台的名称-- "Agausscrab"。

您应该输出 "Stage：答案为 "Agausscrab"。

在第二个例子中，删除后，第一个和最后一个字符串变成了空字符串。





## 问题 G. 在树上爬行

时间限制

6 秒 内存限制

1024 兆字节

然而，面对似乎无穷无尽的未来可能性，我发现自己更倾向于卡尔维诺在《树上的男爵》中提出的概念--在树枝中生活似乎比过早地飞翔更可取。

- 生活在树上

有一棵树，它有  $n$  个顶点，分别用  $1, 2, \dots, n$  标示。在第 1 个顶点，有  $m$  只乌龟。每只乌龟都可以沿着树的双向边爬行，到达其他顶点。乌龟非常重，因此对于第  $i$  条边，乌龟在第  $k_i$  次通过后，这条边就会断开，乌龟无法再次通过。请注意，同一时刻可能有多只乌龟通过同一条边。假设在同一时刻有  $cnt$  只乌龟经过同一条边，那么就应该计算这条边的  $cnt$  次。当然， $cnt > k_i$  是不允许的。

你的任务是指挥这  $m$  只乌龟移动，使第  $i$  个 ( $2 \leq i \leq n$ ) 顶点至少被  $c$  只乌龟访问 <sub>$i$</sub> 。注意，如果乌龟多次访问一个顶点，则只计算一次。请找到一种运动方式，使所有乌龟的总距离最小，否则就确定不可能。

### 输入

输入的第一行包含两个整数  $n$  和  $M$  ( $2 \leq n \leq 10^4$ ,  $1 \leq M \leq 10^4$ )，分别表示顶点数和  $m$  的上限。

接下来的每  $(n-1)$  行包含四个整数  $u_i, v_i, l_i$  和  $k_i$  ( $1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq l_i, k_i \leq 10^9$ )，表示  $u_i$ -th 顶点和  $v_i$ -th 顶点之间的双向边，其长度为  $l_i$ ，乌龟第  $k_i$  次经过该边后，该边将断开。可以保证这些道路组成一棵树。

下一行包含  $(n-1)$  个整数  $c_2, c_3, \dots, c_n$  ( $1 \leq c_i \leq M$ )，表示每个顶点将被访问的海龟最小数量。

### 输出

打印  $M$  行，第  $i$  行 ( $1 \leq i \leq M$ ) 包含一个整数，表示在下列情况下的最小总距离  
如果无法找到可行的移动，请打印"-1"。



实例

标准输入	标准输出
4 2 1 2 3 2 2 3 2 1 2 4 5 1 1 1 1	-1 13
4 2 1 2 3 2 2 3 2 1 2 4 5 1 2 2 2	-1 -1

备注

在第一个例子中，当  $m = 1$  时，不可能让一只乌龟同时到达顶点 3 和顶点 4。当  $m = 2$  时，使总距离最小的可能解决方案之一是让两只乌龟从顶点 1 移动到顶点 2，然后让第一只乌龟移动到顶点 3，再让第二只乌龟移动到顶点 4。两只乌龟移动的总距离为  $(3 + 2) + (3 + 5) = 13$ 。



## 问题 H. 置换

时间限制

2 秒 内存限制

1024 兆字节

这个问题的原始版本要求你找出一个区间中第二大数的位置，但没有一个问题设置者知道如何解决这个问题，所以...

- 假动物团队

这是一个互动问题。

有一个长度为  $n$  的未知排列，你想确定数字  $n$  在这个排列中的位置。

为此，您可以提出以下问题：

- 选择区间  $[l, r]$  ( $l < r$ )，求区间中**第二大数的位置**  $[左, 右]$ 。

您希望在不超过约  $\lceil 1.5 \log_2 n \rceil$  的查询中确定数字  $n$  的位置。此外，由于我们不够聪明，我们的交互器只能找到  $\Theta(r - l)$  中第二大的数字，因此您的查询次数中  $r - l + 1$  的总和不应超过  $3n$ 。

在这个问题中，交互器是非适应性的。也就是说，在所有查询之前，排列组合都是固定的。

### 输入

第一行包含一个整数  $T$  ( $1 \leq T \leq 10000$ )，代表测试用例的数量。

对于每个测试用例，第一行包含一个整数  $n$  ( $2 \leq n \leq 10^6$ )，表示置换的长度。保证所有测试用例的  $n$  之和不超过  $10^6$ 。

### 互动协议

要提问，请打印一行 `"? l r"` ( $1 \leq l < r \leq n$ )。然后从标准输入中读取回复。

要报告答案，请打印一行 `"! x"`，表示数字  $n$  的位置是  $x$ 。

打印答案后，程序应处理下一个测试用例，如果没有其他测试用例，则程序终止。

打印完每一行后，不要忘记输出行尾并刷新输出。要执行后者，可以使用 C++ 中的



`fflush(stdout)` 或 `cout.flush()`、Java 中的 `System.out.flush()` 或 Python 中的 `stdout.flush()`。



示例

标准输入	标准输出
2	
5	
	? 1 5
3	
	? 1 3
3	
	? 2 3
3	
	! 2
2	
	? 1 2
2	
	! 1

备注

请注意，这些示例仅用于展示交互过程的正确格式，但并不保证您在这些交互之后一定能获得确定的结果。



## 问题 I. 小猪分类

时间限制

1 秒 内存限制

1024 兆字节

猪肉排序？我以为是猪肉订单

。

- 将军西诺  
Mahamatra

Putata 和 Budada 提出了一种新算法--小猪排序。这种算法可以通过以下过程轻松地对  $n$  个实数进行排序：

- 假设需要排序的序列为  $v_1, v_2, \dots, v_n$ ，它们是  $n$  个非负实数。
- 普塔塔和布达达小心翼翼地从小猪城里挑选出  $n$  只小猪，这  $n$  只小猪的速度正好是  $v_1, v_2, \dots, v_n$ 。小猪镇可以用坐标轴来描述。第  $i$  只小猪最初位于坐标  $x_i$ 。小猪的初始坐标是**成对不同的**。
- 所有小猪同时开始奔跑。 $t$  秒后，第  $i$  只小猪的坐标为  $x_i + v_i \cdot t$ 。请注意，速度可能为零，这意味着小猪根本没有移动。
- 经过相当长的时间后，坐标轴上小猪的顺序就是  $v_1, v_2, \dots$  序列的（排序）顺序。 $v_n$ 。

普塔塔和布达达进行了一次实验，以验证算法的正确性。然而，时间就是金钱。很长的等待时间是不切实际的。作为替代方案，他们拍摄了  $m$  张小猪的照片。为了确保有足够的实验数据，他们确保照片的数量多于数组中元素的数量，即  $m$  **大于**  $n$ 。

遗憾的是，照片中的许多信息都已损坏。他们可以从照片中获得以下信息：

- 第一张照片拍摄于时间 0，而其他照片的拍摄时间无法区分。这些照片是在**不同的**时间拍摄的。
- 第  $i$  张图片中小猪的坐标为  $x_{i,1}, x_{i,2}, \dots, x_{i,n}$ ，而小猪则无法区分。

请帮助 Putata 和 Budada 弄清实验结果。你们应该找到一个序列  $r_1, r_2, \dots, r_n$ ，它是第一幅图中坐标最小的第  $i$  个小猪的速度秩。只有当且仅当  $(v_i < v_j) \vee ((v_i = v_j) \wedge (x_{1,i} < x_{1,j}))$  时，您才能保证  $r_i < r_j$ ，并且  $r_1, r_2, \dots, r_n$  是  $1, 2, \dots, n$ 。



## 输入

输入包含多个测试用例。第一行包含一个整数  $t$  ( $1 \leq t \leq 250$ )，表示测试用例的数量。

对于每个测试用例，第一行包含两个整数  $n$ 、 $m$  ( $1 \leq n < m \leq 500$ )，分别表示数组的长度和图片的数量。

以下  $m$  行中的第  $i$  行包含  $n$  个整数，其中第  $j$  行表示  $x_{i,j}$  ( $-10^7 \leq x_{i,j} \leq 10^7, x_{i,j} \leq x_{i,j+1}$ )，即小猪在第  $i$  行图片中的位置。如果  $u \neq v$ ，则保证  $x_{1,u} \neq x_{1,v}$ 。

保证  $m$  的总和不超过 500。



输出

对于每个测试用例，在一行中输出  $n$  个整数，分别表示  $r_1, r_2, \dots, r_n$ 。

您应保证  $r_1, r_2, \dots, r_n$  是  $1, 2, \dots$ 。如果有多个答案，请输出任意一个。

示例

标准输入	标准输出
3	1 2
2 4	1
1 2	3 1 2
3 4	
5 6	
7 8	
1 2	
1	
1	
3 4	
1 2 3	
6 9 9	
10 15 17	
12 18 21	





## 问题 J. 偶数或奇数生成树

时间限制 2 秒 内存限制 512 兆字节

奇怪的是，连我自己都觉得奇怪，尽管这似乎是一个理所当然的规则，但我甚至找不到一个以前存在过的例子。这里或那里的奇怪问题甚至与它有点相似，但即便如此，它们都不太一样。真奇怪。甚至

- 萨姆-卡普尔曼-莱恩斯

给你一个有  $n$  个顶点和  $m$  条边的无向图。假设  $T$  是该图的生成树，并将  $\text{Cost}(T)$  表示为  $T$  中所有边的总权重。请找出  $T_1$  和  $T_2$ ，使得：

- 成本( $T_1$ ) 为偶数，成本( $T_1$ ) 最小。
- 成本 ( $T_2$ ) 为奇数，成本 ( $T_2$ ) 最小。

### 输入

第一行包含一个整数  $T$  ( $1 \leq T \leq 10^4$ )，即测试用例数。对于每个测试用例  
第一行包含两个整数  $n$  和  $m$  ( $2 \leq n \leq 2 \cdot 10^5$ ,  $1 \leq m \leq 5 \cdot 10^5$ )，分别表示顶点数和边数。  
以下  $m$  行中的每一行都包含三个整数  $u_i, v_i$  和  $w_i$  ( $1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq w_i \leq 10^9$ )，描述了一条无向边。  
保证所有  $n$  的总和最多为  $2 \cdot 10^5$ ，所有  $m$  的总和最多为  $5 \cdot 10^5$ 。

### 输出

针对每个测试用例，输出包含两个整数的一行： $\text{Cost}(T_1)$  和  $\text{Cost}(T_2)$ 。请注意，如果找不到这样的生成树，请打印"-1"作为代价。

### 示例

标准输入	标准输出
3	-1 5
2 1	-1 -1



第三届环球杯

第八阶段：仓前，2024 年 8 月 31 日至 9 月 1 日

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



## 问题 K. 糖甜 3

时间限制

3 秒 内存限制

1024 兆字节

摩尔的选票问题又是哪个问题  
?

- 匿名 Apiad, 问题

校验器

毕比、彭教授和毕皇帝正在玩游戏。三位玩家持有三种不同类型的牌。皮皮持有第一种类型的  $A$  牌，彭教授持有第二种类型的  $B$  牌，别皇帝持有第三种类型的  $C$  牌。三位玩家玩这个游戏时，牌堆最初是空的。

游戏持续  $A + B + C$  轮，每轮都是如此：

- 至少还剩 1 张牌的玩家出牌。
- 如果牌堆是空的，或者牌堆中所有牌的类型都与他的牌相同，则玩家将这张牌放入牌堆。
- 否则，他会从牌堆中抽出一张牌，连同打出的牌一起扔掉。

例如，假设皮皮持有 1 张类型 1 的牌，彭教授持有 3 张类型 2 的牌，别皇帝持有 2 张类型 3 的牌。玩家出牌的顺序为[皮皮、彭教授、彭教授、别帝、别帝、彭教授]。每轮结束后，牌堆中的牌是：

第一轮结束后：[1]（皮皮将自己的牌放入牌堆）

第二轮结束后：[]（彭教授打出自己的牌，并从牌堆中扔掉一张牌，同时扔掉自己打出的牌）。

第三轮后：[2]（彭教授将自己的牌放入牌堆）

第四轮结束后：[]（别帝打出自己的牌，从牌堆中扔掉一张牌，同时扔掉自己打出的牌）

第五轮之后：[3]（别帝将牌放入牌堆）

第六轮结束后：[]（彭教授打出自己的牌，从牌堆中扔掉一张牌，同时扔掉自己打出的牌）

现在，假设有  $m$  个时刻牌堆是空的，最后时刻牌堆是空的。形式上，存在一个列表  $t_1, t_2 \dots t_m$  ( $1 \leq t_1 < t_2 < \dots < t_m = A + B + C$ )，其中包含所有整数  $t_i$ ，使得在第  $t_i$ -轮 ( $1 \leq i \leq m$ ) 之后，牌堆是空的。三位玩家将获得  $m^x$  袋糖。这里， $x$  是游戏前决定的常数。如果游戏结



束后牌堆不是空的，则三位玩家不会获得任何糖。

现在三个玩家想知道，在所有可能的游戏中，他们能得到的糖袋总和是多少。当且仅当存在  $i$  时，两局游戏被认为是不同的，这样第  $i$  轮的玩家是不同的。输出模型  $10^9 + 7$ 。

## 输入

唯一的一行包含四个整数  $A$ 、 $B$ 、 $C$ 、 $x$  ( $1 \leq A, B, C \leq 1000$ ,  $1 \leq A + B + C \leq 1000$ ,  $1 \leq x \leq 10^9$ )，分别表示貔貅手中的牌数、彭教授手中的牌数和帝誉手中的牌数。



输出

输出一个整数，代表所有可能的游戏中糖袋的总和，模数为  $10^9 + 7$ 。

实例

标准输入	标准输出
1 2 3 1	110
4 5 7 12	881078346

备注

对于第一个例子， $m=1$  时有 6 个可能的有效棋局， $m=2$  时有 16 个可能的棋局，而在  $m=3$  的情况下，有 24 种可能的游戏，所以答案是  $6 \times 1^1 + 16 \times 2^1 + 24 \times 3^1 = 110$ 。



## 问题 L.挑战矩阵乘法

时间限制

9 秒 内存限制

256 兆字节

取一个  $(\omega - 2)$ ，每天切去  
0.001 个，到一万代结束时，还  
会剩下一些。

- 小青鱼

众所周知，计算有向无环图的可达性很难高效解决。但小塔尔延声称，他有一种算法可以在几乎线性的时间内解决这个问题。从形式上看，他解决了下面的问题，并提供了许多测试用例来证明他的算法是正确的：

- 给定一个有  $n$  个顶点和  $m$  条边的有向无环图 (DAG)，求每个节点  $u$  的数目  $r_u$ ，即从顶点  $u$ （包括  $u$  本身）出发可以到达的顶点数。

不过，聪明的你已经发现，小塔尔杰恩生成的所有图形都很特别。更具体地说，让  $in_i$  是顶点  $i$  的入度， $out_i$  是顶点  $i$  的出度，那么所有的图形满足  $\sum_{i=1}^n |in_i - out_i| \leq 120$ 。

你想证明在此约束条件下，解题非常容易。请编写一个解题程序。

### 输入

第一行包含两个整数  $n$  和  $m$  ( $2 \leq n \leq 10^6$ ,  $1 \leq m \leq 10^6$ )。

下面的  $m$  行，每行包含两个整数  $u, v$  ( $1 \leq u < v \leq n$ )，代表图中的一条有向边。  
可以保证  $\sum_{i=1}^n |in_i - out_i| \leq 120$ 。

### 输出

一行包含  $n$  个整数，分别代表  $r_1, r_2, \dots, r_n$ 。



实例

标准输入	标准输出
4 6 1 3 2 3 2 4 1 2 1 3 1 3	4 3 1 1
5 7 1 4 1 5 1 2 2 4 3 4 2 5 1 4	4 3 2 1 1



## 问题 M. 三角形

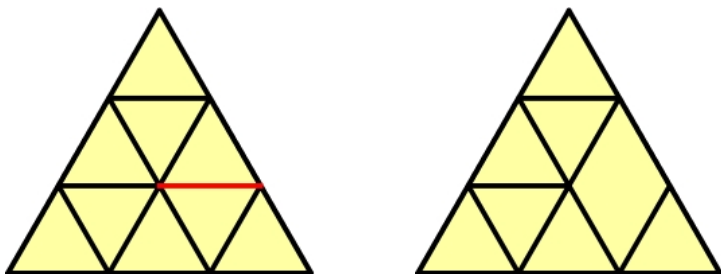
时间限制 1 秒 内存限制 256 兆字节

你知道埃及不是拥有金字塔最多的国家吗？这个称号实际上属于苏丹，苏丹的努比亚金字塔数量超过 200 座，而埃及只有大约 118 座。

- 弗雷迪-汉德

格莱美有一张三角形方格纸，上面有一个大三角形。边长为  $n$  的大三角形被分成了  $n^2$  个边长为 1 的小三角形。三角形方格纸的原始形状如第一幅图所示。

现在，格莱美想用这张纸出一道题，所以她选择了一条水平边并将其删除，然后要求你找出纸上剩余三角形的个数。第二幅图是网格结果的一个示例。



### 输入

唯一一行包含 3 个整数  $n$ 、 $a$ 、 $b$  ( $1 \leq b \leq a \leq n \leq 10^6$ )，分别表示网格边长、所选行和所选删除边的索引。

### 输出

输出一个整数，表示网格中三角形的数量。

### 实例

标准输入	标准输出
3 2 2	10





849586 233333 123456	153307446989958297
----------------------	--------------------

备注

在第一个例子中，初始三角形网格的边长为 3。删除第二横行的第二条边后，网格中还剩下 10 个三角形。