

第三届环球杯



第 1 阶段：圣彼得堡

2024 年 6 月 8-9 日

这套试题应包含 15 个问题，共 20 页（有编号）。

主办方





问题 A. 要素比较

9 日

输入文件：	标准输入
输出文件：	标准输出
时间限制	0.4 秒 内存限
制	1024 兆字节

给你一个长度为 n 的排列组合 p ：它按一定顺序包含数字 $1、2、……、n。$ n 按一定顺序排列。你的任务是找出有多少对长度为 m 的子数组，使得左边的数组比右边的数组元素小。

形式上，考虑两个子数组： $p_i, p_{i+1}, \dots, p_{i+m-1}$ 和 $p_j, p_{j+1}, \dots, p_{j+m-1}$ 当 $i < j$ 时，我们称第一个子数组为左子数组，第二个子数组为右子数组；两个子数组可能相交。当 m 个不等式同时成立时，第一个不等式的元素数小于第二个不等式的元素数：

$$p_i < p_j, \quad p_{i+1} < p_{j+1}, \quad \dots, \quad p_{i+m-1} < p_{j+m-1}。$$

输入

第一行包含两个整数 n 和 m ($1 \leq m \leq n \leq 5 \cdot 10^4$)。第二行包含用空格隔开的 n 个整数：排列 p 本身。

输出

输出唯一的整数：问题答案。

实例

标准输入	标准输出
5 3 5 2 1 3 4	0
5 2 3 1 4 2 5	2
4 2 1 2 3 4	3
4 2 4 3 2 1	0

说明

在第二个例子中，答案 2 由两对子数组组成：

- $\{3, 1\}$ 小于 $\{4, 2\}$ 、
- $\{1, 4\}$ 小于 $\{2, 5\}$ 。



在第三个例子中，答案 3 是由三对子数组组成的

- $\{1, 2\}$ 在元素上小于 $\{2, 3\}$ 、
- $\{2, 3\}$ 小于 $\{3, 4\}$ 、
- $\{1, 2\}$ 在元素上小于 $\{3, 4\}$ 。



问题 B. 女学生

9 日

输入文件：	标准输入
输出文件：	标准输出
时间限制	5 秒 内存限
制	512 兆字节

女学生爱丽丝在科技课上学习了铰链装置。她制作了一个工具，可以利用平行四边形的三个顶点（可能重合，也可能不重合）将平行四边形扩展到第四个顶点。

是平行的)。形式上，给定三个点 A 、 B 、 C ，她可以构造一个点 D ，使得向量 \vec{AB} and \vec{DC} 是平等的。

女学生阿琳娜在几何课上学习了正多边形的概念。在这个问题中，我们将使用以下定义：

- 我们会说，如果所有这些点 A_1, A_2, \dots, A_n ($n \geq 3$) 构成退化正多边形，如果所有这些点重合；
- 如果点 A_1, A_2, \dots, A_n ($n \geq 3$) 按逆时针顺序组成非退化正多边形，如果它们都是不同的，位于以某个圆心 O 为圆心的同一个圆上，并且 $\angle A_1 O A_2 = \angle A_2 O A_3 = \dots = \angle A_{n-1} O A_n = \angle A_n O A_1 = 360^\circ / n = \frac{2\pi}{n}$ ，在所有这些角度中，逆时针方向通过 2π 绕 O 旋转，将 OA_i 映射到 $OA_{(i \bmod n)+1}$ ；
- 我们会说，如果存在排列点 A_1, A_2, \dots, A_n ($n \geq 3$) 构成非退化正多边形，如果存在一个按逆时针顺序排列的排列 $A_{(1)}, A_{(2)}, \dots, A_{(n)}$ 这些点按逆时针顺序组成非退化正多边形；
- 我们会说，如果点 A_1, A_2, \dots, A_n ($n \geq 3$) 构成正多边形，如果它们构成退化正多边形或非退化正多边形。

请注意，最后一个定义与点的顺序无关：如果一个点列表构成一个正多边形，那么它们的任何排列也构成一个正多边形。

校长 Arina 决定测试一下女孩们的能力。首先，她给她们布置了一个任务，在平面上构造 $n+m$ 个点。前 n 个点应按逆时针顺序组成一个不退化的正多边形。接下来的 m 个点中的每个点都要在前三个点的基础上使用爱丽丝工具来构建。

女孩们完成了这部分任务。然后，Arina 开始说出某些点的名称，并询问它们是否构成正多边形。这对女学生们来说相当困难，所以她们向你求助。请编写一个能完成 Arina 任务的程序。

输入

第一行包含三个整数 n 、 m 、 k ：原始正多边形的顶点数、使用 Alice 工具构建的附加点数以及 Arina 将询问的多边形数 ($3 \leq n \leq 10^4$, $0 \leq m \leq 3 \cdot 10^4$, $1 \leq k \leq 10^4$)。点 K_1, K_2, \dots



K_n 按逆时针顺序组成一个非退化正多边形。

接下来的 m 行描述了如何构建点 K_{n+1}, \dots, K_{n+m} 的构造。第 i 行包含三个整数 a_i, b_i, c_i ($1 \leq$

$a_i, b_i, c_i \leq n+i-1$)：爱丽丝工具所应用的三个点的编号。

点 K_{n+i} 定义为 $K_{a_i} K_{b_i} K_{c_i}$ 。部分或全部数字 a, b, c 可能重合。

$n+i$

$K_{a_i} K_{b_i} K_{c_i}$

$a_i b_i c_i$

接下来的 k 行描述 Arina 的点集。第 i 行描述第 i 组点，格式如下

" $r_i P_1^{(i)} P_2^{(i)} \dots P_{r_i}^{(i)}$ "。这意味着女学生需要检查点 $K_{(i)}, \dots, K_{(i)}$

$r_i P_1^{(i)} P_2^{(i)} \dots P_{r_i}^{(i)}$

形成一个正多边形 ($3 \leq r_i \leq 3 \cdot 10^4, 1 \leq P_j^{(i)} \leq n+m$)。可以保证所有 r_i 的总和在

$\leq P$

大多数 $3 \cdot 10^4$ 。部分或全部数字 $P^{(i)}$ 可能重合。

P_{r_i}



输出

9 日

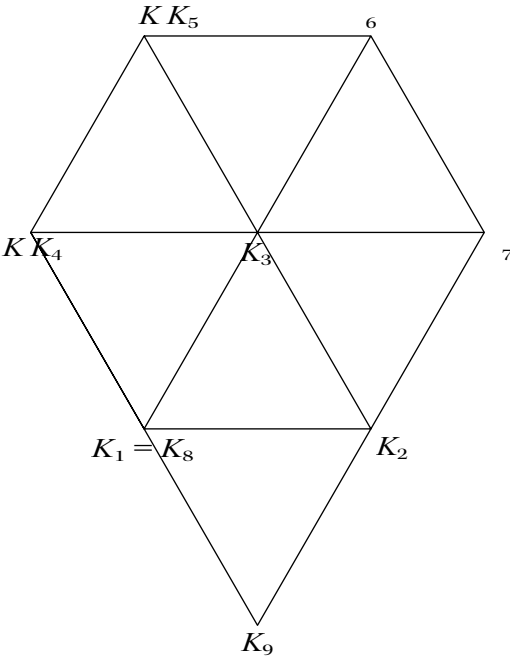
输出 k 行。如果 Arina 的第 i 个集合构成正多边形，则第 i 行应包含 "是"，否则为 "否"。输出的每个字母可以是任何大小写（大写或小写）。

示例

标准输入	标准输出
3 6 8	是
1 2 3	是
3 1 4	是
5 4 3	没有
3 1 2	没有
4 5 3	没有
4 5 2	是
6 4 7 6 5 1 2	没有
3 1 3 2	
3 1 1 8	
4 2 5 6 7	
3 2 1 4	
3 6 5 9	
3 4 7 9	
4 1 3 2 8	

备注

图片为示例：





问题 c. 摘樱桃

9 日

输入文件：	标准输入
输出文件：	标准输出
时间限制	2 秒 内存限
制	512 兆字节

统计专家弗拉基米尔-鲍里索维奇（Vladimir Borisovich）定期分析不同棋手的对局，寻找有趣的系列。如果连续 40 次战胜等级分至少为 2900 的对手，他就会认为这是一个有趣的系列。在这中间，棋手可以与等级分较低的对手进行任意数量的对局：重要的是，与等级分较高的对手进行的 40 局对局都以胜利告终。

每当弗拉基米尔-鲍里索维奇（Vladimir Borisovich）发现一个有趣的系列时，他都会将其分享到社交网络上。不幸的是，Vladimir Borisovich 的所有操作都是手动进行的，因此他需要花费大量时间来搜索有趣的系列。通过解决一个更普遍的问题，帮助 Vladimir Borisovich 自动完成寻找有趣系列的过程。

你将得到一位棋手连续 n 场对局的结果和对手等级分。您的任务是找出最大可能的等级分 x ，如果将所有对手等级分严格小于 x 的对局从序列中删除，则存在一系列 k 连胜。

输入

第一行包含两个整数 n 和 k ($1 \leq k \leq n \leq 100\,000$)。第二行包含 n 个空格分隔的整数 r_i ：对手的评分 ($1 \leq r_i \leq 100\,000$)。第三行包含 n 个字符，分别为 0 和 1：如果第 i 场比赛获胜，则第 i 个字符等于 "1"，否则等于 "0"。

输出

如果答案存在，则输出 x ，否则输出 0。

实例

标准输入	标准输出
5 2 1 2 3 4 5 01101	2
5 2 3 4 5 2 1 10101	0



问题 D. 小矮人的就寝时间 ^{9 日}

输入文件：标准输入
输出文件：标准输出
时间限制 2 秒 内存限制
制 512 兆字节

这是一个互动问题。

在森林的边缘，有一座小房子。白雪公主和 N 个小矮人住在那里。

众所周知，每个小矮人一天中正好有一半的时间是连续睡觉的，而这一半的时间是在每天的同一时间开始的。在一天的另一半时间里，小矮人保持清醒。

作为女主人，白雪公主想知道每个小矮人上床睡觉的准确时间。在白天的 1440 分钟里，白雪公主可以查看任何一个小矮人的床铺，了解他们是睡着了还是醒着。不过，白雪公主检查每个小矮人的床铺不能超过 50 次：否则，小矮人就会对侵犯他的隐私感到愤怒。

请帮助白雪公主在一天之内学会每个小矮人上床睡觉的准确时间。

在每次测试中，小矮人的时间表都是事先固定的，在互动过程中不会改变。换句话说，这个问题中的互动者不是自适应的。

互动协议

首先，读取包含整数 n 的单独一行：房子里住着多少个小矮人 ($1 \leq n \leq 100$)。

白雪公主使用的是 24 小时显示的时钟。互动开始于 00:00，结束于 23:59。要检查 HH:MM (小时从 00 到 23，分钟从 00 到 59) 时刻第 i 个小矮人是否在睡觉，请单独打印一行，格式为 "at HH:MM check i "。作为回应，您将得到单独一行：如果第 i 个小矮人在这一分钟正在睡觉，则显示 "睡着了"；如果他醒着，则显示 "醒着"。下一次查询的时间不能早于上一次。

要输出答案，请单独打印一行 "答案"，然后再打印 n 行：第一个、第二个、.....、第 n 个小矮人睡觉的时间。第 n 个小矮人进入睡眠状态的时间，格式为 HH:MM。然后，结束程序。

如果您的解决方案对某个矮人执行了过多检查，或打印了错误答案，然后立即终止，则会得到 WA (错误答案)。请记住，在每次检查和给出答案后都要打印换行符并刷新输出缓冲区，否则您的解决方案将得到 IL (超过闲置限制)。

示例

标准输入	标准输出
2	
睡眠	在 01:40 查看 1



觉	在 01:40 查看 2
睡眠	在 07:59 查看 1
觉	在 08:00 查看 1
觉	在 13:41 查看 2
	答话
	20:00
	01:41



说明

9 日

在这个例子中，房子里住着 $n = 2$ 个小矮人。

第一个小矮人从 20:00 睡到 07:59（含）。我们之所以知道这一点，是因为他在 07:59 睡着，但在 08:00 醒了。这意味着小矮人醒来的时间正好是 8:00。因此，他睡觉的时间正好是 12 小时之后。

第二个小矮人从 01:41 睡到 13:40（含 13:40）。我们知道这是因为他在 01:40 和 13:41 都醒着。从 13:41 分到 01:40 分（含 01:40 分）这段时间正好是半天。因此，这是他清醒的第一分钟和最后一分钟。

请注意，白雪公主可以在同一分钟（示例中为 01:40）检查多个小矮人的床铺。

添加空行只是为了方便读者。在真正的输入和输出中，没有空行。



问题 E. 假硬币和假天平

9 日

输入文件：标准输入
输出文件：标准输出
时间限制 4 秒 内存限
制 512 兆字节

假设你有 C 枚硬币。其中一枚是假币，比真币轻。其他硬币都是真的，重量相同。你也有两盘天平秤。在一次称量中，你可以把一些硬币放在一个盘子里，把一些硬币放在另一个盘子里（剩下的硬币两个盘子都不放），然后看哪个盘子里的硬币轻。你的任务是确定哪枚硬币是假的。

听起来很熟悉吧？不过，有好消息也有坏消息。坏消息是，天平可能会欺骗你多达 k 次，而你却无法知道哪些称量结果是真的。好消息是，你最多可以犯 $3k$ 次错误。也就是说，你最多可以猜 $3k+1$ 次，只要其中至少有一次是正确的，你就赢了。

设 $f(n, k)$ 为最大数 c ，这样，如果你有 c 枚硬币，天平最多可能出现 k 次错误，而你最多可能犯 $3k$ 次错误，那么就存在一种称重策略，无论结果如何，你只要称重不超过 n 次就能获胜。

你应该能找到 $f(n, k)$ 的近似值。特别是，输出 $\ln f(n, k)$ ，其中 \ln 是自然对数。如果您的答案与正确答案的绝对差值不超过 10，您的答案将被视为正确答案。

输入

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

接下来的 t 行中，每一行都包含两个整数 n 和 k ($1 \leq n \leq 10^9$; $0 \leq k \leq 10^9$)。

输出

对于每个测试用例，打印一行，其中包含一个实数：该测试用例的答案 $\ln f(n, k)$ 。如果答案与正确答案最多相差 10，则该答案将被接受。

示例

标准输入	标准输出
2	109.8612289
100 0	106.1174552
100 1	

备注

在第一个测试案例中， $f(100, 0) = 3^{100}$ 。这里，答案是 $\ln(3^{100}) = 100 \cdot \ln(3) = 109.8612289 \dots$ 自然对数（以 e 为底的对数 = 2.71828182845904523536）可使用 C++ 中的 `log`、Python 中的



math.log 和 Java 中的 Math.log 计算。

9 日



问题 F. 整个世界

9 日

输入文件：标准输入
输出文件：标准输出
时间限制 8 秒 内存限制
制 512 兆字节

如果实系数多项式在所有整数点上都取整数值，我们就称它为全系数多项式。

给你 n 对整数 (x_i, y_i) 。你应该找出整个多项式的最小可能度数

f ，使得 $f(x_i) = y_i$ for $i = 1, 2, \dots, n$.

在这个问题中，考虑多项式 $f(x) = 0$ 的阶数为 0。

输入

输入的第一行包含一个整数 T ($1 \leq T \leq 100$)，表示测试用例的数量。

每个测试用例以一行开始，其中包含一个整数 n ($1 \leq n \leq 30$)，表示点的数量。然后是 n 行，每行包含一对整数 (x_i, y_i) ($1 \leq x_i \leq 30, -10^9 \leq y_i \leq 10^9$)。保证所有 x_i 都是成对不同的。

输出

对于每个测试用例，打印一行，其中包含一个整数：该测试用例的答案。

示例

标准输入	标准输出
2	3
2	1
1 0	
4 1	
3	
1 1	
4 4	
6 6	



问题 G. 异常情况

9 日

输入文件：标准输入
输出文件：标准输出
时间限制 3 秒 内存限制
制 512 兆字节

汉密尔顿爵士喜欢长途跋涉...

给您一个随机无向图，该图有 n 个顶点和 m 条边：该图是从所有具有相同顶点和边数的图中随机均匀选择的。你的任务是在给定的图中找出 k 条不相交的哈密顿路径。

如果一条路径恰好经过图中的所有顶点一次，则称为哈密尔顿路径。

在这个问题中，有一个例子和整整 30 次测试。除示例外的所有测试中， $n = 10\,000$ 、 $m = 200\,000$ ， $k = 8$ 。

输入

第一行包含三个整数：顶点数 n 、边数 m 和要找到的路径数 k 。

然后用 m 行来描述图的边。每条边由一对从 1 到 n 的整数描述：即它所连接的顶点的编号。该图没有循环，也没有多条边。

输出

输出 k 行。每行输出一条哈密顿路径：按遍历顺序排列的 n 个顶点序列。图中的每条边最多可用于一条输出路径。如果有多个可能的答案，则输出其中任意一个。保证每个给定的测试都有一个答案。

示例

标准输入	标准输出
5 9 2 1 3 1 4 1 5 2 3 2 4 2 5 3 5 4 3 5 4	1 3 5 2 4 5 1 4 3 2



备注

9 日

语句中的示例是 $n \neq 10\,000$ 、 $m \neq 200\,000$ 、 $k \neq 8$ 的唯一测试，用于演示输入和输出格式。
该示例是测试系统中的测试 1。



vdome.com 上的问题 H. Page⁹日

输入文件：	标准输入
输出文件：	标准输出
时间限制	2 秒 内存限
制	512 兆字节

在 vdome.com 网站上，每个新用户都会被分配一个页面，其地址的形式为 `vdome.com/idK`，其中 K 是该用户的顺序号，从创建网站开始计算。例如，网站创建者的地址是 `vdome.com/id1`，他的助手和朋友的地址是 `vdome.com/id2`、`vdome.com/id6` 等。

该网站很快受到欢迎，多年后的今天，它已经拥有约 10^9 个用户。因此，小号网页的所有者会定期收到陌生人的联系，要求购买他们的网页，号码越小，地址越有吸引力。此外，有些人还设法将 `vdome.com/idK` 改为 `vdome.com/namelogin` 形式的字母地址。因此，数字地址行中出现了一些 "空白"。

约翰决定为自己找到一个当前可用地址中数字最小的地址。为此，他从暗网中获得了一个包含所有形式为 `vdome.com/idK` 的活跃用户地址的数据库，并试图从中找出最小的数字。由于收到的数据库中有太多记录，他编写了一个简单的搜索程序。但约翰只是个程序员初学者，因此，当他从每条记录的字符串地址中提取片段时，不小心将字符串反转了，然后才将其读作整数。结果，例如，他从 `vdome.com/id12345` 字符串中得到了 54321，从 `vdome.com/id67500` 中得到了 576，以此类推。也就是说，所有的数字都是倒着写的，前导零消失了，约翰要在得到的数字中寻找最小的数字。

如果他知道，即使用户使用的是字母地址，也可以通过编号访问 `vdome.com` 上的任何页面，因此从第一个到第 N 个的所有编号都在数据库中，不会出现空白，他就可以避免这一切。但约翰对此一无所知。因此，对于给定的 N （数据库中的地址记录数），请回答他的程序将输出的最小缺失数字是多少。

输入

第一行指定一个整数 N ，表示数据库中的记录数 ($1 \leq N \leq 10^9$)。

输出

打印约翰的程序将输出的页码。

实例

标准输入	标准输出
------	------



5	6 9 日
10	10

说明

在第一个例子中，数据库包含 id1、id2、id3、id4、id5 记录。经过处理后，它们变成了由数字 1、2、3、4、5 组成的列表：个位数在反转时保持不变。其中遗漏的最小数字是 6。

在第二个例子中，数据库包含 id1、id2、id3、id4、id5、id6、id7、id8、id9、id10 等记录。经过处理后，它们变成了由数字 1、2、3、4、5、6、7、8、9、1 组成的列表：id10 首先转化为 01，当转化为整数时，它变成了数字 1。其 中 最小的缺失数是 10。



问题 1. 自旋与旋转

输入文件：	标准输入
输出文件：	标准输出
时间限制	2 秒 内存限
制	512 兆字节

有一个大正方形 $ABCD$ （顶点按逆时针顺序排列）。每个顶点上都有一个木钉。每个木钉都将其绳端固定在地面上。绳子可以无限拉伸和收缩，但绝不能离开正方形上方的区域。最初，绳索是 *不缠绕的*：也就是说，如果把它们收缩到足够大，那么其中一条绳索就会沿着正方形的一条边，而另一条绳索就会沿着正方形的对边。

有一个叫 Ka-BAN 的机器人，它可以执行两个指令：

- 旋转，记为 S：绳索的四端都从木桩上分离，绕纵轴逆时针旋转 90° ，保持绳索之间的所有缠结，然后重新连接到新的木桩上。或者，我们也可以将这一操作理解为正方形、木桩和绳索保持不动，但顶点的名称顺时针旋转 90° 。
 - 顶点 A 的新名称是 B ；
 - 顶点 B 的新名称是 C ；
 - 顶点 C 的新名称是 D ；
 - 顶点 D 的新名称是 A ；

就本问题而言，这两种操作是等价的。

- 旋转，用 R 表示：Ka-BAN 将绳索从 A 和 D 两端的木桩上拆下，互换，然后重新连接。在交换过程中，最初连接在 A 上的绳端穿过最初连接在 D 上的绳端。如果站在广场外靠近 AD 的一侧观察这个过程，就会发现 A 端和 D 端好像逆时针绕对方旋转了 180° 。

你有一串字母 "S" 和 "R"，这是 Ka-BAN 执行的程序：指令从左到右逐一执行。不幸的是，经过这个过程后，方块和绳索看起来就像一团难以理解的乱麻。为了解决这个问题，您买了一个新的机器人 Iz-BAN。当您阅读说明书时，却大吃一惊：Iz-BAN 和 Ka-BAN 有相同的两条指令！这意味着解开缠结会很困难：尽管任何旋转都可以通过三次旋转来解开，但似乎没有解开旋转的简单方法。

尽管如此，可以证明这两个操作足以解决这个问题！给定 Ka-BAN 执行的程序，为 Iz-BAN 写一个最短的程序，使得在执行该程序后，绳索再次变得不缠绕。



输入

9 日

第一行包含一个整数 T ，即测试用例的数量 ($1 \leq T \leq 300\,000$)。

接下来的每 T 行包含一个由字母 "S" 和 "R" 组成的字符串 s ：Ka-BAN 执行的程序 ($1 \leq |s| \leq 300\,000$)。所有测试用例的字符串长度总和不超过 300 000。

输出

对于每个测试用例，打印一行由字母 "S" 和 "R" 组成的程序：Iz-BAN 解开绳索的最短程序。如果绳子已经解开，则打印一个字母 "S"。可以证明，答案总是存在的，而且是唯一的。

实例

9 日

[illegible]



问题 J. 第一个十亿

9 日

输入文件：	标准输入
输出文件：	标准输出
时间限制	2 秒 内存限
制	512 兆字节

给你 N 个正整数。你的任务是从给定的数字中选择一些，使所选数字的和正好是 10^9 。

为了让您更轻松地解决问题，评委会承诺将按以下方式组织测试。

- 1. 生成两个和 10^9 完全相同的 n 个正整数的随机集合（对所有和 10^9 的集合进行均匀分布）。
- 2. 给你 $N = 2n$ 个按随机顺序生成的元素。 $N = 2n \leq 100$ 。

输入

第一行包含一个正整数 $N = 2n \leq 100$ 。第二行包含 N 个正整数 a_1, \dots, a_N 。问题中正好有 100 次测试。

输出

输出一行，描述总和等于 10^9 的集合⁹：首先是指数 k 的个数，然后是 k 个不同的指数 i_1, \dots, i_k ，满足 $\sum_{j=1}^k a_{i_j} = 10^9$ 。

不一定要找到一个大小正好为 n 的集合。

示例

标准输入
10 386413329 88494216 245947398 316438989 192751270 204627269 65749456 3938400 150458676 345180997
标准输出
5 2 3 4 8 10



问题 K. 任务和错误

输入文件：	标准输入
输出文件：	标准输出
时间限制	2 秒 内存限
制	512 兆字节

尼古拉是开发部门的负责人，他希望尽快完成任务，但不能出现任何错误。为此，他将任务和错误保存在 Jira 跟踪系统中。任务和错误的标识符以 CS-X 格式表示，其中 X 是包含 1 至 5 位小数的数字。为了加快任务的最终完成，尼古拉需要了解每个任务有多少未解决的错误。

他使用跟踪系统中的一个特殊过滤器，编制了一份未解决的错误列表，并为每个错误编制了一份与该错误相关联的非空任务列表。尼古拉运行过滤器后，就会得到一份未解决的错误及其相关任务的列表：

```
CS-20CS-1
CS-100: CS-239
CS-300: CS-239, CS-11111
```

错误 CS-20 与任务 CS-1 相关联，错误 CS-100 与任务 CS-239 相关联，错误 CS-300 与任务 CS-239 和 CS-11111 相关联。错误以及与每个错误相关的任务按标识符升序排列（标识符按数字排序）。错误标识符与任务标识符不同。

现在，尼古拉将任务和错误按数字顺序重新分组。重组后的结果如下：

```
CS-1: CS-20
CS-239: CS-100, CS-300
CS-11111: CS-300
```

任务 CS-1 与错误 CS-20 相关联，任务 CS-239 与错误 CS-100 和 CS-300 相关联，任务 CS-11111 与错误 CS-300 相关联。任务以及与每个任务相关的错误按标识符升序排列（标识符以数字排序）。

然而，尼古拉并不想花时间进行重组。那就编写一个程序来帮他完成吧。

输入

输入由一行或多行组成。每行包含一个错误名称和与该错误相关的任务列表。错误与任务之间用冒号和空格隔开。列表中每两个任务之间用逗号和空格隔开。

错误以及与每个错误相关的任务按数字标识符升序排列。错误数量从 1 到 100 不等，每个错误的任务数量从 1 到 10 不等。

输出



打印一行或多行：针对每项任务，列出与之相关的错误。格式和顺序与输入相同。

示例

标准输入	标准输出
CS-20CS-1	CS-1: CS-20
CS-100： CS-239	CS-239： CS-100, CS-300
CS-300： CS-239, CS-11111	CS-11111： CS-300



问题 L.糖果

9 日

输入文件：标准输入
输出文件：标准输出
时间限制 5 秒 内存限制
1024 兆字节

假设有三个数 x_1 、 x_2 和 x_3 。最初，它们都是 0。唯一的条件是，在任何时候， x_1 都应该是 x_1 、 x_2 和 x_3 中最大的（形式上， $x_1 \geq x_2$ 和 $x_1 \geq x_3$ ）。最终有多少种方法可以达到 $x_1 = a$ ， $x_2 = b$ ， $x_3 = c$ ？如果两种方法至少有一步不同，那么它们就是不同的。

由于答案可能非常大，因此将其输出为质数 998 244 353 的模数。

输入

第一行包含三个整数 a 、 b 和 c ，其中 $1 \leq b, c \leq a \leq 10\,000$ 。

输出

输出结果应包含一个数字，等于质数的模乘方式数
998 244 353.

示例

标准输入	标准输出
4 3 2	368



问题 M. 厕所

9 日

输入文件：	标准输入
输出文件：	标准输出
时间限制	5 秒 内存限
制	512 兆字节

有一个长度为 L 的圆形办公室，在办公室的不同位置有几个厕所。每个员工都有一个预定的时间 t_i ，当他们从工作地点起身时，坐标 p_i 在圆圈上，并开始以单位速度沿预定方向绕办公室行走，直到找到一个可用的厕所，之后他们将占用该厕所 d_i 。如果员工遇到一个被占用的厕所，他们将从该厕所经过。如果多名员工同时到达一个可用的厕所，则由较早出发的员工占用（保证 t_i 是单调递增的）。在前一名员工离开厕所的同时，下一名员工也会占用该厕所。

为每一位员工找到他们将在什么时间、什么地点上厕所。

输入

第一行包含三个数字 n 、 m 和 L ($1 \leq n, m, \leq 2 \cdot 10^5, 1 \leq L \leq 10^{12}$)：分别是员工人数、厕所数量和办公室长度。

第二行包含 m 个数字 x_i ($0 \leq x_i < L$)：厕所的坐标。保证所有 x_i 都是不同的。

以下 n 行描述雇员，每行一个。每个雇员由四个值 t_i, p_i, s_i, d_i 描述 ($0 \leq t_i < 10^{18}, 0 \leq p_i < L, 1 \leq d_i \leq 10^{12}, t_i < t_{i+1}$)。字符 s_i 是 "+" 还是 "-"（不带引号），取决于第 i 个雇员是向坐标增加的方向移动还是向坐标减少的方向移动。保证所有输入的数字都是整数。

输出

输出 n 行，每行包含两个数字。第 i 行应包含第 i 位员工将占用的卫生间坐标和发生时间。

示例

标准输入	标准输出
5 3 45	20 20
10 20 30	10 7
0 0 + 200	30 30
2 5 + 10	10 60
20 40 - 100	10 105
21 16 + 10	
50 0 + 22	



问题 N. (未) 标注的图形 ^{9 日}

输入文件：	标准输入
输出文件：	标准输出
时间限制	3 秒 内存限
制	512 兆字节

这是一个运行两次的问题。

帕夫努蒂老人想把 n 个顶点上的标注图 G 发送给无限老人。当且仅当有两个不同的整数 $i, j \in [n]$ 时，顶点集合 $[n] = \{1, 2, \dots, n\}$ 上的两个标签图是不同的，当且仅当有两个不同的整数 $i, j \in [n]$ 使得顶点 i 和 j 在其中一个图中相邻，而在另一个图中不相邻。

不幸的是，帕夫努蒂老人的旧电脑只能存储无标记图。在顶点集合 $[m] = \{1, 2, \dots, m\}$ 是不同的，当且仅当不存在这样的排列 $\pi \in S_m$ ，即如果根据该排列对其中一个图的顶点重新编号，那么这两个图就会成为相等的带标记图。因此，如果试图将一个带标签的图输入帕夫努蒂老人的老式计算机，后者会以某种方式将其所有顶点洗牌。

Pafnutiy 老人决定向 Infinitiy 老人发送一个更大的图：也就是说，如果 G 有 n 个顶点，那么 Pafnutiy 老人将发送 $m = f(n) = n + \lceil \log_2 n \rceil + 3$ 个顶点上的某个图 H 。然后，鉴于 H 和 H' 作为无标记图是相等的，老人 Infinitiy 只需以某种方式将收到的图 H' 解码回 G 。但他们如何组织这种传输呢？

输入

第一行包含两个整数 v_1 和 v_2 ：输入图和输出图的顶点数 ($3 \leq \min\{v_1, v_2\} \leq 2024$)。接下来的 v_1 行包含长度为 v_1 的二进制字符串：输入图的邻接矩阵。

图是无向的，因此邻接矩阵是对称的。此外，保证不存在自循环。

输出

打印 v_2 长度为 v_2 的二进制字符串₂：输出图的邻接矩阵。

如果 $v_1 < v_2$ ，则 $v_1 = n$ ， $v_2 = m = n + \lceil \log_2 n \rceil + 3$ 。在这种情况下，你应该像帕夫努蒂老人一样：接收图 G 并输出图 H 。

否则， $v_2 = n$ ， $v_1 = m = n + \lceil \log_2 n \rceil + 3$ 。在这种情况下，你应该扮演 Infinitiy 老头的角色：接收图 H' （这是对 H 的顶点进行某种排列后得到的），并输出初始图 G 。

图应该是不定向的，因此邻接矩阵应该是对称的。此外，图中不能有自循环。



实例

9 日

标准输入	标准输出
5 11	00010000000
01110	00000000000
10110	00000000000
11001	10000000000
11001	00000000000
00110	00000000000
	00000000000
	00000000000
	00000000000
	00000000000
	00000000000
	00000000000
11 5	01110
01000000000	10110
10000000000	11001
00000000000	11001
00000000000	00110
00000000000	
00000000000	
00000000000	
00000000000	
00000000000	
00000000000	
00000000000	
00000000000	



问题 o. 神秘序列

9 日

输入文件：标准输入
输出文件：标准输出
时间限制：2 秒 内存限制：512 兆字节

在古老的数字王国 X 中，有一个神秘的公式将每个数字与之前的数字联系在一起：

$$x_{i+2} = a - x_{i+1} + b - x_i$$

这个公式非常强大，即使是最聪明的魔术师，如果不知道前两个数字，也无法破译序列中的所有数字。

然而，随着时间的推移，除了第一个和最后一个数字外，序列的所有成员都丢失了。

很久很久以前，一位最高级别的巫师被要求找出这个神秘序列中的所有数字 x_i 。他踏上了漫长的旅程，深深地思索着咒语和数学公式。他用自己的知识和魔法，深入研究数字的秘密，最终揭示了这个神秘公式中包含的所有隐藏的 x_i 。

那你呢？只知道第一个和最后一个数字，你能重建序列的所有成员吗？

输入

第一行包含实数 A 和 B ($0.25 \leq A, B \leq 1$ ，小数点后最多两位) 以及整数 n, X_1, X_n ($2 \leq n \leq 10, 1 \leq X_1, X_n \leq 100$)。

输出

输出 n 行：数字 X_1, X_2, \dots, X_n ，每行一个。如果绝对或相对误差不超过 10^{-6} ，答案将被视为正确。

示例

标准输入	标准输出
1.0 1.0 10 1 10	1 -0.3235294118 0.6764705882 0.3529411765 1.029411765 1.382352941 2.411764706 3.794117647 6.205882353 10