



10th 中国大学生程序设计竞赛（济南）

2024 年 10 月 27 日



	问题	时间/记忆
A	脚	1 秒/1024MB
B	魔术师	1 秒/1024MB
C	皇后	2 秒/1024MB
D	皇帝	2 秒/1024MB
E	战车	2.077 秒 / 1024MB
F	隐士	2 秒/1024MB
G	命运之轮	7 秒 / 1024MB
H	实力	3 秒 / 1024MB
	被吊死的人	3 秒 / 1024MB
J	节制	2s/ 1024MB
K	魔鬼	7s/ 1024MB
L	塔楼	3s/ 1024MB
M	判决	1s/ 1024MB

共计：14 张、 28 页

比赛开始前不要打开问题集。

本页故意留空。

问题 A. 傻瓜

输入文件：	标准输入
输出文件：	标准输出
时间限制	1 秒 内存限制
	1024 兆字节

"这个地方有很多情绪的味道。兴奋、愤怒、喜悦，还有.....咸？咸是一种感觉吗？"

- 好奇的变色龙尼科

Neeko 来自一个失落已久的浩瀚亚部落，她可以借用他人的外表混入任何人群，甚至可以吸收他人的情绪状态，在瞬间分辨敌我。没有人能够确定妮可可能在哪儿，或者是谁，但那些想要伤害她的人很快就会目睹她的真面目，并感受到她原始精神魔法的全部威力。

Neeko 潜入了一个尺寸为 $n \times m$ 的字符网格，网格中的每个单元格都由长度为 k 的字符串表示。她可以模仿其他单元格；但是，Neeko 的单元格中至少有一个字符与其他单元格中的字符不同，而其他单元格中的字符都相同。您的任务是找到 Neeko。

来源: <https://bilibili.com/BV1ub421e7cd/>

Neeko 潜入了一个尺寸为 $n \times m$ 的字符网格，网格中的每个单元格都由长度为 k 的字符串表示。她可以模仿其他单元格；但是，Neeko 的单元格中至少有一个字符与其他单元格中的字符不同，而其他单元格中的字符都相同。您的任务是找到 Neeko。

输入

输入的第一行包含三个整数 n 、 m 、 k ($2 \leq n, m \leq 200$, $1 \leq k \leq 10$)。

接下来的 n 行每行包含 $m - k$ 个字符，由从 ! (33) 到 ~ (126) 的可见 ASCII 字符组成。第 i 行中的第 $(j - 1) - k + 1$ 个至第 $(j - k)$ 个字符代表单元格 (i, j) 。

保证没有多余的空格或换行，而且答案可以唯一确定。

输出

打印两个整数 r 、 c ，表示 Neeko 在第 r 行和第 c 列的位置。

示例

标准输入	标准输出
3 5 3 QWQQWQQWQQWQQWQ QWQQQWQQQWQQQWQ QWQQWQQWQQWQQWQ QWQQWQQWQWQWQWQ QWQWQ	3 5

本页故意留空。

问题 B. 魔术师

输入文件：标准输入

输出文件：标准输出





















































时间限制1 秒 内存限制1024 兆字节

"现在，大幕拉开"

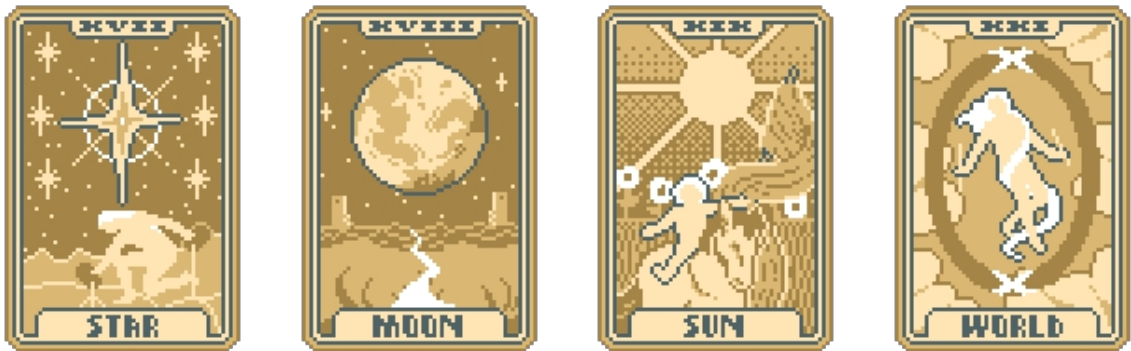
- ????

您会得到一些来自大阿卡纳的特殊塔罗牌，包括 "恋人"、"死亡"、"星星"、"月亮"、"太阳"和 "世界"，以及手中的其他一些扑克牌。每种塔罗牌你最多只能拥有一张，而且每张塔罗牌都有一种独特的能力，可以改变手中其他扑克牌的花色。您的任务是确定在使用每张塔罗牌最多一次后，手中的扑克牌最多能打出多少张同花顺。

扑克牌是标准扑克牌，如下图所示。

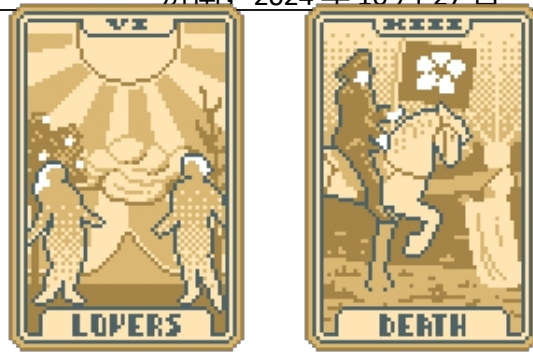
Example set of 52 playing cards; 13 of each suit: clubs, diamonds, hearts, and spades													
	Ace	2	3	4	5	6	7	8	9	10	Jack	Queen	King
Clubs													
Diamonds													
Hearts													
Spades													

以下是塔罗牌能力列表。



- 星星：将最多 3 张选定的扑克牌转换为钻石 (♦)。

-
- ~~月亮~~将最多 3 张选定的扑克牌转换为梅花 (♣)。
 - ~~太阳~~将最多 3 张选定的扑克牌转换为红心 (♥)。
 - *The World*: 将最多 3 张选定的扑克牌转换为黑桃 (♠)。



- 恋人们将 1 张选定的扑克牌转换成任意牌（⊕，可用作任意花色）。
- 死亡：选择正好两张扑克牌，用另一张牌的副本替换其中一张（复制花色、等级和野牌状态）。

您可以按照任何顺序使用塔罗牌。每张塔罗牌最多只能使用一次，永远不能使用。关于“恋人”有一条特殊规则：

- 一旦一张扑克牌被转换成狂野牌（使用“恋人”或“死亡”），即使使用了“星星”、“月亮”、“太阳”和“世界”，它仍然是一张狂野牌；
- 但是，如果用死亡来替换一张非野生卡的复制卡，那么生成的卡将不会是野生卡。

同花顺是由 5 张扑克牌组成的牌组，可以被视为同花色：有一种花色（钻石、梅花、红心和黑桃），使得 5 张扑克牌中的每一张都是这种花色或通配。

打出**同花**意味着从手牌中丢弃组成**同花**的 5 张牌，所有这些牌都不能用于另一个**同花**。与包括**巴拉特罗**在内的许多纸牌游戏相反，出牌后不会从牌组中抽取新牌。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 13$) --测试用例数。测试用例说明如下。

第一行包含一个整数 n ($1 \leq n \leq 52$)，即手中扑克牌的数量。

第二行包含 n 个空格分隔的字符串，每个字符串代表手中的一张扑克牌。每张扑克牌由两个字符表示：一个表示等级，一个表示花色，其中花色为 D（方块）、C（梅花）、H（红心）或 S（黑桃）之一，等级为 2-9、T（10、10）、J（Jack）、Q（Queen）、K（King）或 A（Ace）之一。

第三行包含六个空格分隔的整数 $t_1, t_2, t_3, t_4, t_5, t_6$ ($0 \leq t_i \leq 1$)，其中：

- t_1 代表“星”的编号。

- t_2 代表 ~~月亮~~ 的数字。
- t_3 代表 ~~太阳~~ 的数字。
- t_4 代表 "世界" 的编号。
- t_5 代表 "恋人" 的编号。
- t_6 代表 ~~死亡~~ 人数。

保证 T 个测试用例中的 n 之和不超过 $104 = 52 \times 2$ ，并且每个测试用例中的扑克牌都是成对不同的。

输出

对于每个测试用例，打印单行中最多可打的同花顺次数。

示例

标准输入	标准输出
4	1
5	1
2H 3H 4H 5H 6D	0
1 1 1 1 0 0	2
5	
2S 3S 4D 5C 6D	
0 0 1 0 1 1	
5	
2S 3S 4D 5C 6D	
0 0 1 0 1 0	
13	
AS 2S 3s 4s 5h 6h 7h 8h 9h th jh qh kh	
0 0 0 0 0 1	

备注

在第一种情况下，我们可以用 "太阳" 将 $6\Diamond$ 转换成 $6\heartsuit$ ，然后将 $2\heartsuit 3\heartsuit 4\heartsuit 5\heartsuit 6\heartsuit$ 打成同花顺。这并不是唯一可能的方法；另一种可能的方法是打出同一组牌：

- 使用 *The World* .NET 将 $4\heartsuit 5\heartsuit 6\Diamond$ 转换为 $4\spadesuit 5\spadesuit 6\spadesuit$ ；
- 使用 *The Sun* 将 $4\spadesuit 5\spadesuit 6\spadesuit$ 转换为 $4\heartsuit 5\heartsuit 6\heartsuit$ 。

在第二个测试案例中，一种可能的同花顺玩法是

- 使用 *The Sun* 将 $2\spadesuit 3\spadesuit 4\Diamond$ 转换为 $2\heartsuit 3\heartsuit 4\heartsuit$ ；
- 使用 *The Lovers* 将 $5\star$ 转换为 $5\otimes$ ；
- 用 *死亡的副本* $5\otimes$ 代替 $6\Diamond$ ；
- 播放 $2\heartsuit 3\heartsuit 4\heartsuit 5\otimes 5\otimes$ 。

本页故意留空。

问题 c. 皇后

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1024 兆字节

注：这是问题 "皇帝 "的逆版本，但限制条件有所不同。

Capoo 发明了一种名为 Push-Pop 的有趣语言。这种语言是一种解释型语言。解释器从无限容量的空堆栈开始，读取自定义程序的第一条指令。这种语言只有两种指令：

- POP a GOTO x; PUSH b GOTO y
如果栈顶元素是 a ，则弹出栈一次，并将控制流转移到第 x 条指令（这意味着下一条指令将是第 x 条指令）。
否则，将元素 b 推入堆栈，并将控制流转移到第 y 条指令。
- HALT; PUSH b GOTO y
如果堆栈为空，则在执行该指令后停止整个程序。
否则，将元素 b 推入堆栈，并将控制流转移到第 y 条指令。

Capoo 想要编写一个 Push-Pop 程序，在执行完 k 条指令后停止运行。由于解释器的天真实现，一个程序最多只能包含 64 条指令。

输入

唯一一行包含一个整数 k ($1 \leq k \leq 2^{31} - 1$, k 为奇数)。

输出

第一行包含一个整数 n ($1 \leq n \leq 64$)，表示指令数，然后是 n 行表示 Push-Pop 程序。对于每条指令， $1 \leq a, b \leq 128$, $1 \leq x, y \leq n$ 应该成立。对于给定的输入，保证存在一个解。

实例

标准输入	标准输出
1	1 停止; 推 1 GOTO 1

5	济南, 5 2024 年 10 月 27 日
	<div>pop 1 推送 2; push 1 goto 2</div> <div>halt; 1 GOTO 3</div> <div>pop 1 4; push 2 goto 4</div> <div>POP 2; push 2 goto 4</div> <div>1 暂停 99 GOTO 4</div> <div>;</div>

备注

第二个示例中的指令是1（push）、2（push）、3（pop）、4（pop）、2（halt）。与"

皇帝"相比，约束条件的主要差异：

- $n \leq 64$;
- 对于给定的输入，解决方案总是存在的；
- 程序应在**整整** k 个指令后停止，**不考虑**模数 998 244 353。

本页故意留空。

问题 D. 皇帝

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1024 兆字节

注：这是问题 "皇后 "的逆版本，但限制条件有所不同。

Capoo 发明了一种名为 Push-Pop 的有趣语言。这种语言是一种解释型语言。解释器从无限容量的空堆栈开始，读取自定义程序的第一条指令。这种语言只有两种指令：

- POP a GOTO x; PUSH b GOTO y
如果栈顶元素是 a ，则弹出栈一次，并将控制流转移到第 x 条指令（这意味着下一条指令将是第 x 条指令）。
否则，将元素 b 推入堆栈，并将控制流转移到第 y 条指令。
- HALT; PUSH b GOTO y
如果堆栈为空，则在执行该指令后停止整个程序。
否则，将元素 b 推入堆栈，并将控制流转移到第 y 条指令。

Capoo 希望升级天真解释器，以处理更多的指令。给定一个最多包含 1024 条指令的程序，计算程序停止前执行的步数。

输入

第一行包含一个整数 n ($1 \leq n \leq 1024$)，然后是 n 行，每行包含一条指令。保证每条指令 $1 \leq a, b \leq 1024, 1 \leq x, y \leq n$ 。

输出

如果程序永远不会停止，或执行的指令数为 998 244 353，则打印-1。

实例

标准输入	标准输出
1 停止; 推 1 GOTO 1	1
5 pop 1 推送 2; PUSH 1 GOTO 2 halt; 1 GOTO 3 pop 1 4; 按 2 GOTO 4 POP GOTO 4 2; PUSH 2 99 GOTO 4	5

济南，2024 年 10 月 27 日	
-1 暂停 ；	
1 民研計 GOTO 1; PUSH 1 GOTO 1 劃 1	-1

备注

与 "皇后 "相比，制约因素的主要差异：

- $n \leq 1024$;
- 答案可能不存在（程序从未停止），在这种情况下，报告-1;
- 对于每个将停止的程序，打印其结果 **modulo** 998 244 353。

本页故意留空。

问题 E. 战车

输入文件：标准输入
输出文件：标准输出
时间限制 2.077 秒 内存
限制：1024 兆字节 1024 兆字节

在夜城瞬息万变的混乱环境中，不确定性是唯一的确定因素。为了安全出行，掘进者们依赖于最好的技术。尽管价格昂贵且具有公司背景，但德拉门人工智能武装出租车仍是最可靠的交通工具。

今天，您需要在夜城进行一次长途旅行。德拉门出租车的票价结构如下：

- 基本票价为 A 欧元，覆盖前 X 米。
- 接下来的 Y 米，票价为每米 B 欧元。
- 超过 $X + Y$ 米后，票价为每米 C 欧元。

在任何时候，您都可以执行以下操作：停车并重新打车。这样，您将立即结清上一辆出租车的费用，下一辆出租车的车费将从基本车费开始重新计算。此操作可执行任意次数。

现在，您想确定使用德拉门出租车行驶 D 米的最低成本（以欧元计）。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 2077$) --测试用例数。测试用例说明如下。

输入一行包含六个整数： A 、 B 、 C 、 X 、 Y 、 D ($0 < A, B, C, X, Y, D < 10^{2077}$) 以十进制表示，不含前导零。

保证 T 个测试用例中 A 的位数总和不超过 $0x2077$ 。这一限制也分别适用于 B 、 C 、 X 、 Y 、 D 。

约束中使用的值 $0x2077$ 是十六进制数，等于十进制数 8311。

输出

对于每个测试案例，输出一行，其中包含使用德拉门出租车行驶 D 米的最小成本（以欧元计）。

可以证明答案是一个正整数。请打印出不包含前导零的十进制整数。

示例

标准输入	标准输出
5	160
160 27 41 3 12 3	187
160 27 41 3 12 4	3226
160 27 41 3 12 99	999
1 999 999 1 99 999	100000000000099799
999 999 1 1 99 9999999999999999	

备注

对于第四个样本，最佳解决方案是叫 999 辆出租车，每隔一米就叫一辆新的。这看起来很奇怪，但你确信这是在 "夜城 "生活的最佳方式。

本页故意留空。

问题 F. 隐士

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 1024 兆字节

"我的眼睛从我的居所看到了道"你好，勇士

- 徐狗

知名炼丹师徐狗发现，精确去除杂质可以提高他所炼制的灵丹的灵性。通过日复一日的炼丹，他发现这些杂质的性质与数学问题有着千丝万缕的联系。鉴于你的道行尚浅，徐狗决定用最直接的方式告诉你他需要解决的数学问题，而不是通过深奥的炼丹问题。

给定两个正整数 $n \leq m$ ，计算 $\{1, 2, \dots, m\}$ 的所有子集的答案之和，取模 998 244 353：

- 在由 n 数组成的集合中，你可以去掉一些数，使集合的最小值不等于集合的最大公约数。求移除后集合中能保留的最大元素个数。如果没有非空子集满足条件，则答案定义为 0。

集合的最大公因子是指集合中所有元素的公因子中最大的值。例如，集合 $\{6, 9, 15\}$ 的最大公约数是 3。

输入

输入由一行组成，包含两个整数 m, n ($1 \leq n \leq m \leq 10^5$)。

输出

输出一个代表答案的整数，模数为 998 244 353。

实例

标准输入	标准输出
4 3	7
11 4	1187
100000 99999	17356471

备注

下面列出了第一个例子中的所有情况：

- $\{1, 2, 3\}:\{2, 3\}$

- $\{1, 2, 4\}$:无解

- $\{1, 3, 4\}:\{3, 4\}$

- $\{2, 3, 4\}:\{2, 3, 4\}$

因此，答案是 $2 + 0 + 2 + 3 \bmod 998\,244\,353 = 7$ 。

本页故意留空。

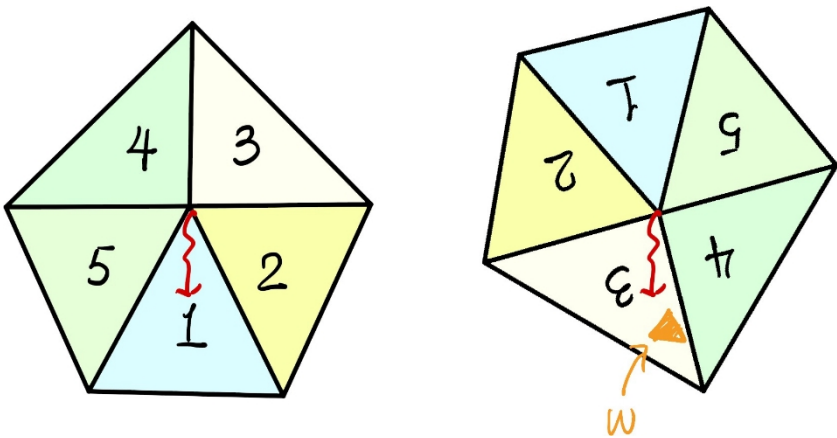
问题 G. 命运之轮

输入文件：	标准输入
输出文件：	标准输出
时间限制	7 秒 内存限制
	1024 兆字节

希洛菲特和大祭司对宇宙的奥秘非常着迷。为了在地球（显然是宇宙的一部分）上探索其中的一些奥秘，他们使用了一个奇怪的轮子来研究重力对随机过程的影响。

转盘是一个凸多边形，由连接旋转中心和每个顶点的线分成几个三角形区域，每个三角形区域代表一个奖品。转盘转动后，奖品由停止位置正下方的区域决定。一个无偏差的转盘的旋转中心位于其中心点，因此最终的中奖位置完全取决于每个区域所占的角度。不幸的是，这里的转盘可能存在偏差：当旋转中心不在中心点时，转盘的结果总是从旋转中心指向中心点。

显然，一个有偏差的轮子是无趣的。为了让各种轮子变得更有意义，“嗜血者”决定用以下方法来扰乱结果：在轮子区域内的一个均匀随机位置（视为位于凸多边形内的一个点质量）放置一个小磁砝码，就会改变中心点，从而影响结果。



第一个样本：没有磁铁的情况，以及磁铁改变结果的可能情况。

大祭司想知道经过上述扰动后，每个区域成为最终获胜区域的概率。

可以证明，转盘唯一确定中奖区域（即旋转中心位于中心点或中心点位于区域边界）的概率度量为 0，因此可以放心地忽略这种情况。

输入

第一行包含两个整数 n, w ($3 \leq n \leq 100000, 1 \leq w \leq 10^9$)，分别代表轮子的顶点数和磁铁的质量。

接下来的 n 行分别包含两个整数 x_i, y_i ($|x_i|, |y_i| \leq 30000$)，它们代表了"....."的坐标。

~~i 轮子的第 1 个顶点，沿边界按逆时针顺序排列。~~

最后一行包含两个整数 O_x, O_y ($|O_x|, |O_y| \leq 30000$)，表示旋转中心的坐标，它严格位于凸多边形内部。

车轮的顶点都是不同的，但有三个点可能是相邻的：可以保证，当三个相邻点 $u, v = (u \bmod n) + 1, w = (v \bmod n) + 1$ 相邻时，点 v 严格位于车轮上。

济南，2024 年 10 月 27 日

连接 u 和 w 的线段，确保所有内角都在 $(0, \pi]$ 范围内。保证车轮的面积 S 满足： $\max_{\substack{u, w}} \frac{S_w}{w S} \leq 1000$ 。

输出

输出 n 行，每行包含一个实数 p_i ，表示连接第 i 个顶点和 $(i \bmod n) + 1$ 个顶点的直线所围成的区域是最终获胜区域的概率。如果答案的绝对或相对误差不超过 10^{-6} ，则视为正确答案。

实例

标准输入	标准输出
5 5 1 0 3 0 4 2 2 4 0 2 2 2	0.313777778 0.235555556 0.107555556 0.107555556 0.235555556
8 8 0 0 1 0 2 0 2 1 2 2 1 2 0 2 0 1 1 1	0.125000000 0.125000000 0.125000000 0.125000000 0.125000000 0.125000000 0.125000000 0.125000000
3 3 -1 -10 1 -10 0 1 0 0	1.000000000 0.000000000 0.000000000
4 3600000 -30000 -30000 30000 -30000 30000 30000 -30000 30000 1 0	0.249998611 0.248327778 0.249998611 0.251675000
4 2500 5 0 5 5 0 5 0 0 1 1	0.402977500 0.402977500 0.097022500 0.097022500

问题 H. 强度

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：
1024 兆字节

在日常对话中，人们经常使用近似方法来表达数值。无论是讨论时间、金钱还是其他数字，人们都倾向于用四舍五入的方法来简化交流。例如，如果您和朋友在一家餐厅用餐，账单上的金额是 98 元，很多人会说：“账单是 100 元”，而不是使用准确的数字。

如果我们采取更激进的方法，进行多次四舍五入，最终结果可能会变得荒谬。例如，你可以把 145 四舍五入为 200，因为 145 可以四舍五入为 150，然后再四舍五入为 200；当有人说 2000 时，在四舍五入之前，实际上可能是 2001、1999、1888 甚至 11451。

给定一个数 x ，计算 x 在 $[0, z]$ 范围内的不确定性，即在 $[0, z]$ 范围内，经过进取舍入后能成为 x 的数的计数。这里，进取舍入的定义是任意（可能是零）次执行以下舍入操作：

- 设 x 的十进制表示为 $x_{kk-1} \dots x_1$ ，并选择一个索引 $i \in \{1, 2, \dots, k\}$ 。
- 如果 $x_i < 5$ ，则从 x 中减去 $x_i - 10^{i-1}$ ；
- 否则，在 x 上添加 $(10 - x_i) - 10^{i-1}$ 。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 10^5$) -- 测试用例数。测试用例说明如下。

输入一行包含两个整数： x 和 z ($0 \leq x, z \leq 10^{18}$)。其含义参见上文。

输出

每行包含一个数字，代表 x 在 $[0, z]$ 范围内的不确定性。

示例

标准输入	标准输出
5	2147483647
0 2147483646	55
10 100	0
671232353 1232363	1919810
123001006660996 3122507962333010	114514
100019990010301090 44519984489341188	

备注

在第二个测试案例中， $10i + j$ ($i = 0, 1, \dots, 9, j = 5, 6, 7, 8, 9$) 和 10, 11, 12, 13, 14 可以四舍五入为 10。因此，答案是 55。

本页故意留空。

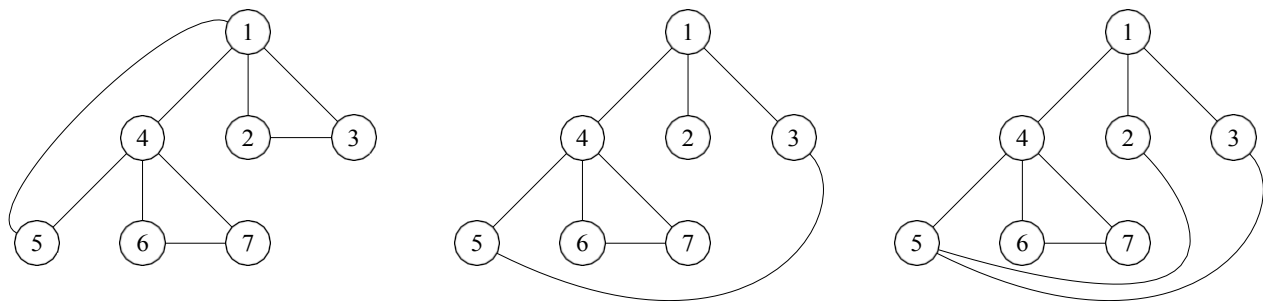
问题 I. 被绞死的人

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
	1024 兆字节

克劳德特-莫雷尔是一名植物学家，热衷于研究各种植物。一天，她在研究玫瑰时，不小心被刺扎到了手。凭借植物学知识，她知道如何处理伤口，但更重要的是，她想防止此类事件再次发生。为此，她想出了一个办法：让玫瑰刺消失。

玫瑰花可以看作一棵有 n 个节点的树。为了让玫瑰无刺，克劳德特-莫雷尔可以在图中添加几条边，只要添加的边不产生多条边或自循环。但是，她不能在图中添加新的节点。

当且仅当每条边都恰好出现在一个简单循环中时，简单图才是**无刺图**。简单循环的定义是不包含任何重复节点（起点和终点节点相同除外）的循环。下面的插图解释了什么是无刺图，什么不是。



- 左图：这是一张无刺图。
- 中间：这不是一个无刺图，因为边 $(1, 2)$ 没有出现在任何简单循环中。
- 对这不是无刺图，因为边 $(1, 2)$ 同时出现在两个循环中 $1 \sim 2 \sim 5 \sim 4 \sim 1$ 和 $1 \sim 2 \sim 5 \sim 3 \sim 1$ 。

现在，克劳德特-莫雷尔拿出了她的玫瑰，而你的任务是分析它们能否变成无刺图形。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 10^5$) --测试用例数。测试用例说明如下。

第一行包含一个整数 n ($2 \leq n \leq 3 \cdot 10^5$) -- 树中的节点数。

下面 $n - 1$ 行中的每一行都包含两个整数 u_i 和 v_i ($1 \leq u_i, v_i \leq n$) , 表示 $(u_i, v_i)_i$ 是树上的一条边。

保证 T 个测试用例中 n 的总和不超过 $3 \cdot 10^5$ 。

输出

对于每个测试用例, 如果树无法转化为无刺图, 则输出-1。

否则，在第一行输出 k ($0 \leq k \leq n$) --您添加的边的数量。

在接下来的 k 行中，每行应包含两个整数 x_i 和 y_i ($1 \leq x_i, y_i \leq n$) --即您添加的边。请注意，添加边缘后，不允许有多条边缘和自循环。如果有多个解，请打印任意一个。

示例

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

备注

左图显示的是第二个测试案例。

问题 J. 腾博会登录

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
	1024 兆字节

在这个世界上，只有绿色从未让我失望。
。

- 陈龙

知名农民陈龙发现，合理的种植密度可以提高产量。
现在，我们可以把农场看作一个三维坐标系，把一棵植物看作坐标系中的一个点。有 n 种不同的植物 $A_i = (x_i, y_i, z_i)$ 。对于每种植物 A_i ，其密度定义如下。

- 假设除 A_i 之外，还有 a 、 b 和 c 株植物的 x 、 y 或 z 坐标分别与 A_i 相同。那么， A_i 的密度为 $\max\{a, b, c\}$ 。

由于陈龙的植物喜欢内卷，他决定移除一些密度较小的植物。请回答至少需要移除多少棵植物，才能使剩下的每棵植物的密度都大于或等于 k 。请注意，移除一个点后，其他植物的密度可能会发生变化。特别是，移除所有植株总是有效的。
您需要分别求解 $k = 0, 1, \dots, n - 1$ 分别求解。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 2 \times 10^4$) --测试用例数。测试用例说明如下。
第一行包含一个整数 n ($1 \leq n \leq 10^5$) --植物的数量。
在接下来的 n 行中，第 i 行包含三个整数 x_i, y_i 和 z_i ($1 \leq x_i, y_i, z_i \leq 10^5$) --即每株植物的坐标。
可以保证 n 个植物的坐标是不同的。
保证 T 个测试用例中 n 的总和不超过 2×10^5 。

输出

对于每个测试用例，在一行中输出 n 个整数，代表 $k = 0, 1, \dots, n - 1$ 。

示例

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

本页故意留空。

问题 K. 魔鬼

输入文件：标准输入

输出文件：标准输出

时间限制7 秒 内存限制1024 兆字节

人们使用缩写来简洁地表达。当两个短语使用相同的缩写时，就会带来麻烦。例如，在网上搜索 "CCPC "一词，你不仅能找到想要的 "中国大学生程序设计竞赛"，还能找到 "中国汽车性能挑战赛"、"竞争和消费者保护委员会 "等。

但是，没有人会一边写代码一边开车。为了解决这个问题，你决定为你经常使用的每一个短语指定一个类似首字母缩写的独特缩写。一共有 n 个短语，每个短语由小写和大写英文字母组成。要为一个短语创建缩写，需要为短语中的每个单词选择一个非空前缀，然后按顺序连接起来。例如，"ChCoPrCo "和 "CCPContest "是 "中国大学生程序设计竞赛 "的有效缩写，但 "CCCP "和 "CCPiC "则不是。每个短语的缩写都独立于其他短语：同一个词在不同的短语中可以有不同的缩写。

构建缩写方案，使所有缩写的总长度最小，同时确保 n 缩写是不同的。

输入

第一行输入包含一个整数 n ($1 \leq n \leq 128$) ，表示短语的数量。

接下来的 n 行每行包含一个非空短语，由不超过 128 个非空单词组成，单词之间用空格隔开。每个单词仅由不超过 128 个小写和大写英文字母组成。保证没有两个短语是相同的。

输出

如果没有可能的解决方案，则单行打印 "无解决方案"。

否则，打印 n 行，其中第 i 行包含解决方案中第 i 个短语的缩写，按输入给出的顺序排列。如果有多个解决方案，则打印任意一个。

实例

标准输入	标准输出
5	大气
自动出纳机 主动出纳机	AT
主动麻烦制造机	MA
美国泄密机械	ACT
	M
	ATE
	M

济南, 2024 年 10 月 27 日	
5 森林保护委员会 论坛 Fuming Corruption Collusion Federation Fulsome Cash Concealment Foundation Funky Crony Capitalism Facilitator Funny Cocky Cocky Funny	FCCF FCoCF FuCCF FCCFa FCCFu
3 A AA AA A A A A	无解

本页故意留空。

问题 L. 塔楼

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 1024 兆字节

Houraisan Kaguya 是一位喜欢在家看视频的 NEET 公主。最近，她经常在一个名为 Mikufans 的著名视频共享网站上观看视频。

Mikufans 上有一个实用功能，允许用户在视频播放过程中留言，这就是所谓的 "danmaku"。有时，会同时出现很多 "danmaku" 留言，以至于辉耀无法全部接收。



来源: <https://bilibili.com/video/BV1xx411c79H>

为简单起见，我们只关注顶部的弹幕信息：顶部的弹幕信息显示在视频屏幕的顶部，每条信息占一行。同时显示的弹幕信息数量没有限制（虽然 Kaguya 的屏幕只会显示前 10^9 行，但其余信息仍会正确地保留在屏幕的溢出区域）。

在视频播放过程中，可能会出现三种事件：

1. 新用户发送一些顶级 danmaku 消息。每条信息将按顺序排列在最上面的空行。
2. 来自特定用户的 danmaku 消息会消失，其所在的行也会变成空行。其他信息则不受影响，仍然保留在各自的位置上。
3. 辉耀对弹幕信息很感兴趣，因此她想知道顶级弹幕的发件人是谁

——特定行的信息。

辉耀每天都有很多视频要看，她太忙了，没时间从头开始重新看视频，所以她向您求助。请帮她找到丹幕信息的发送者。

输入

第一行输入包含一个整数 n ($1 \leq n \leq 5 \times 10^5$)，表示事件的数量。

以下 n 行中的每一行都按顺序包含一个事件。每个事件以下列格式之一描述：

- 1 k : 新用户发送 k 条 ($1 \leq k \leq 10^9$) 顶级 *danmaku* 消息。用户 ID 是以前未使用过的最小正整数。
- 2 u : 用户 u 的 *danmaku* 消息消失。保证 ID 是有效的，而 *danmaku* 用户 u 的信息以前从未消失过。
- 3 l : 辉夜想知道最上面第 l 行 ($1 \leq l \leq 10^9$) 的 *弹幕* 信息发送者的 ID。如果该行为空，则答案定义为 0。

输出

对于类型 3 的每个查询，用一行输出答案。

实例

标准输入	标准输出
7 1 2 1 4 3 3 2 1 3 2 1 4 3 7	2 0 3
5 3 6 3 8 1 2 1 5 3 2	0 0 1

问题 M. 判决

输入文件：	标准输入
输出文件：	标准输出
时间限制	1 秒 内存限制
	1024 兆字节

周末，青山和她的朋友丹尼尔一起制作了一款名为 "*Generalissimos*" 的单人游戏，玩家可以在游戏中绘制伟大的画作。

游戏在一张 $n \times m$ 的桌子上进行。每个单元格都有自己的颜色，即红色、蓝色或白色。最初，单元格 (a, b) 是红色的，单元格 (c, d) 是蓝色的（这两个单元格不重合），其他单元格是白色的。我们称 (a, b) 和 (c, d) 为特殊单元格，其他单元格为非特殊单元格。在游戏过程中，玩家可以执行一定的操作，包括三个步骤：

1. 玩家选择一个非特殊单元格 (x, y) 。
2. 玩家选择另一个单元格 (x', y') 。它必须是非白色的，而且必须与 (x, y) 相邻（即 (x', y') 和 (x, y) 必须有一条公共边）。
3. 单元格 (x, y) 的颜色为 (x', y') 的颜色。

换句话说，在一次操作中，玩家可以给一个非特殊单元格涂上与其相邻的非白色单元格相同的颜色。需要注意的是，一个单元格可能会被着色不止一次，而最新的颜色会覆盖之前的颜色。

玩家可以执行任意次数的操作，然后停止游戏。之后，地图表的最终配置将被打印出来。

不幸的是，"将军"中充满了作弊者，而作弊者可以随时在任何位置上着色。为了维护正义，你决定编写一个判断程序，以确定给定配置在正常对局中是否可能是合法配置，或者其中一定有作弊者。

输入

输入由多个测试用例组成。第一行包含一个整数 T ($1 \leq T \leq 10^4$) --测试用例数。测试用例说明如下。

第一行包含两个整数 n, m ($1 \leq n, m \leq 500, 2 \leq n \cdot m$) --行数和列数。

第二行包含四个整数 a, b, c 和 d ($1 \leq a, c \leq n, 1 \leq b, d \leq m$)。

接下来的 n 行中，每行包含 m 个字符。每个字符为 "R"、"B" 或 "."，分别代表一个红色单元格、一个蓝色单元格和一个白色单元格。

济南, 2024 年 10 月 27 日

保证单元格 (a, b) 和 (c, d) 不重合, 且 a 行 b 列和 c 行 d 列上的字符分别为 "R" 和 "B"。

保证 T 个测试用例中 $n - m$ 的总和不超过 250 000。

输出

对于每个测试用例, 如果是合法配置, 则打印 "YES" (不带引号), 否则打印 "NO" (不带引号)。

您可以打印任何大小写的字母 (大写或小写)。

示例

标准输入	标准输出
4 3 3 1 1 1 2 RBB RRR BBR 6 6 1 1 6 6 RRRRRR BBBBBR BRRRBR BRBBBB BRRRRR BBBBBB 5 5 3 3 4 4 BBR.B BBR.B RRR.B ...BB BBBB. 1 5 1 1 1 3 RBBBR	是 是 没有 没有

备注

下图显示了第一个测试案例，以及玩家如何在不作弊的情况下达到配置。每顶皇冠代表一个特殊单元。

