



第十届中国大学生程序设计竞赛（济南）

2024 年 10 月 27 日



题目		时空限制
A	愚者	1s / 1024MB
B	魔术师	1s / 1024MB
C	皇后	2s / 1024MB
D	皇帝	2s / 1024MB
E	战车	2.077s / 1024MB
F	隐士	2s / 1024MB
G	命运之轮	7s / 1024MB
H	力量	3s / 1024MB
I	倒吊人	3s / 1024MB
J	节制	2s / 1024MB
K	魔鬼	7s / 1024MB
L	高塔	3s / 1024MB
M	审判	1s / 1024MB

共 14 张 28 页

在比赛开始前，请不要翻阅试题册。

此页留空。

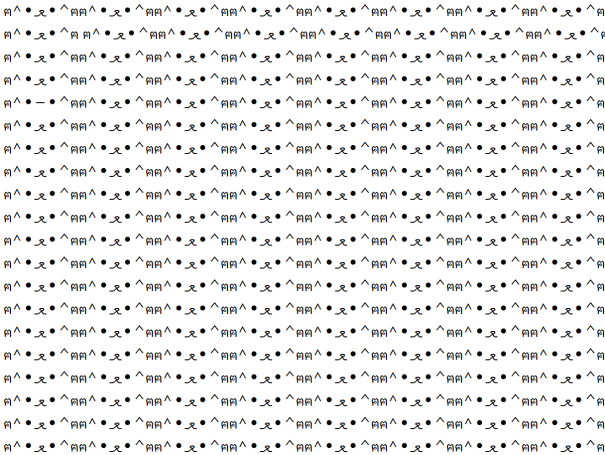
题目 A. 愚者

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 MB

“这个地方尝起来有好多种感情，兴奋，愤怒，快乐，还有...苦味？这也是一种感情吗？”

— 妮蔻，万花通灵

妮蔻来自一个早已迷失的瓦斯塔亚部落。她可以借用别人的外表来伪装自己，融入人群，甚至通过影响别人的情绪状态，一瞬间就能化敌为友。没人知道妮蔻到底在哪儿 — 或者到底是谁，但是想要为难她的人会立刻见识到她的本色，感受原始的精神魔法倾泻在自己身上的痛苦。



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

妮蔻已经潜伏到一个 $n \times m$ 的字符网格中，每格由长度为 k 的字符串表示。她可以模仿其他网格；然而，妮蔻的网格与其他网格至少有一个字符不同，而所有其他网格都相同。你的任务是找到妮蔻。

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 行包含每行 $m \cdot k$ 个字符，由可见的 ASCII 字符组成，范围从 `!(33)` 到 `~(126)`。第 i 行中的第 $((j - 1) \cdot k + 1)$ 到第 $(j \cdot k)$ 个字符表示格子 (i, j) 。
保证没有额外的空格和回车，并且答案可以唯一确定。

输出

输出两个整数 r, c ，表示妮蔻在网格的 r 行 c 列。

样例

标准输入	标准输出
3 5 3 QWQQWQQWQQWQQWQ QWQQWQQWQQWQQWQ QWQQWQQWQQWQQWQ	3 5

此页留空。

题目 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 MB

“此刻，大幕渐起。”

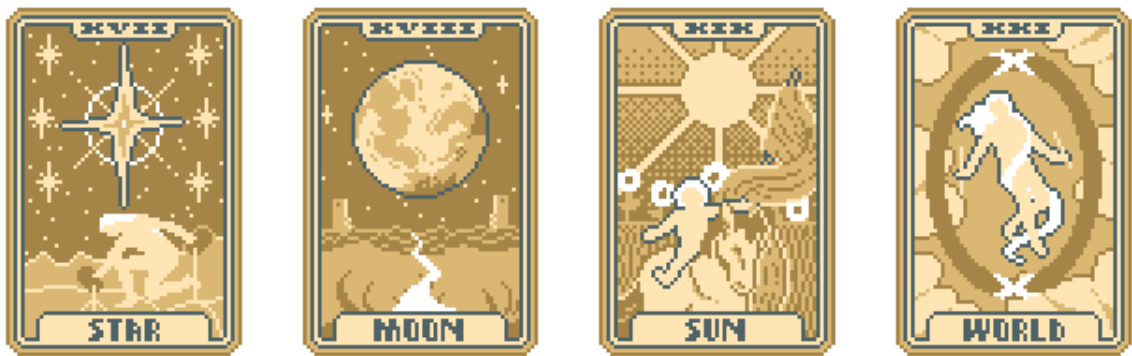
— ????





你有一些塔罗牌，每张塔罗牌可能是恋人，死神，星星，月亮，太阳和世界，以及另外一些扑克牌。每种塔罗牌只有至多一张，且都有独特的能力来改变你手中扑克牌的花色。你的任务是，在每张给定的塔罗牌最多使用一次的情况下打出尽可能多的同花。

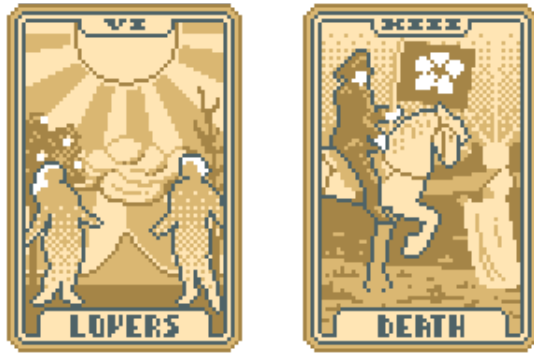
扑克牌是标准的扑克牌，如下所示。

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													

以下是塔罗牌的能力列表。



- 星星：将最多3张选择的扑克牌转换为方块（）。
- 月亮：将最多3张选择的扑克牌转换为梅花（）。
- 太阳：将最多3张选择的扑克牌转换为红桃（）。
- 世界：将最多3张选择的扑克牌转换为黑桃（）。



- 恋人：将1张选择的扑克牌转换为万能牌（ \clubsuit ，可以用作任何花色）。
- 死神：选择2张扑克牌，把一张牌变成另一张牌的副本（获得其花色、点数和万能牌状态）。

你可以以任何顺序使用塔罗牌。每张给定的塔罗牌最多可以使用一次，也可以不使用。关于恋人有一个特殊规则：

- 一旦某张牌（使用恋人或死神）转换成万能牌后，对其使用星星、月亮、太阳和世界后，它仍然是万能牌；
- 然而，如果使用死神将一个万能牌变成另一张非万能牌的副本，该牌将不是万能牌。

同花是一组 5 张可以被视为相同的花色的扑克牌：存在一种花色（方块、梅花、红桃和黑桃）使得这 5 张扑克牌中的每一张要么是这种花色，要么是万能牌。

打出一个**同花**意味着将组成该**同花**的 5 张牌从手中丢弃，所有这些牌都不能用于另一个**同花**。与许多纸牌游戏（包括 *Balatro*）相反，在打出牌后，你不能从牌堆中抽取新的牌。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 13$) — 测试数据的数量。接下来是测试数据的描述。

第一行包含一个整数 n ($1 \leq n \leq 52$)，表示手中的扑克牌数量。

第二行包含 n 个以空格分隔的字符串，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：一个表示点数，一个表示花色，其中花色是 D（方块）、C（梅花）、H（红桃）或 S（黑桃），点数是 2-9、T (10, Ten)、J (Jack)、Q (Queen)、K (King) 或 A (Ace) 中的一个。

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

- t_1 代表星星的数量。
- t_2 代表月亮的数量。
- t_3 代表太阳的数量。
- t_4 代表世界的数量。
- t_5 代表恋人的数量。
- t_6 代表死神的数量。

保证在 T 组数据中， n 的总和不超过 $104 = 52 \times 2$ ，并且每组数据中的扑克牌两两不同。

输出

对于每个测试数据，在一行输出可以打出的**同花**的最大数量。

样例

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	

注释

在第一个样例中，一种可能的方式是使用太阳将 6◇ 转换为 6♥，并将 2♥3♥4♥5♥6♥ 打出作为同花。除此之外，另一种可能的方式是：

- 使用世界将 4♥5♥6◇ 转换为 4♠5♠6♠；
- 使用太阳将 4♠5♠6♠ 转换为 4♥5♥6♥。

在第二个样例中，打出一个同花的一种可能方式是：

- 使用太阳将 2♠3♠4◇ 转换为 2♥3♥4♥；
- 使用恋人将 5♣ 转换为 5⊗；
- 使用死神将 6◇ 变为 5⊗的副本；
- 打出 2♥3♥4♥5⊗5⊗。

此页留空。

题目 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 MB

注意：本题是“皇帝”的逆版本，数据限制有所不同。

咖波发明了一种有趣的语言，名为 Push-Pop。这种语言是一种解释型语言。解释器初始有一个无限容量的空栈，并执行给定程序的第一条指令。这种语言只有两种指令：

- POP a GOTO x ; PUSH b GOTO y
如果栈非空且栈顶为 a ，则弹出栈顶，并将控制流转移到第 x 条指令（接下来运行第 x 条指令）。
否则，将元素 b 推入栈中，并将控制流转移到第 y 条指令。
- HALT; PUSH b GOTO y
如果栈为空，则在执行此指令后停机。
否则，将元素 b 推入栈中，并将控制流转移到第 y 条指令。

咖波想要构造一个 Push-Pop 程序，在执行完恰好 k 条指令后停止。由于咖波的解释器实现得非常简易，一个程序最多可以包含 64 条指令。

输入

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

输出

第一行包含一个整数 n ($1 \leq n \leq 64$)，表示指令的数量，然后是 n 行，表示 Push-Pop 程序。对于每条指令，应满足 $1 \leq a, b \leq 128$, $1 \leq x, y \leq n$ 。

保证输入存在一个解。

样例

标准输入	标准输出
1	1 HALT; PUSH 1 GOTO 1
5	5 POP 1 GOTO 2; PUSH 1 GOTO 2 HALT; PUSH 1 GOTO 3 POP 1 GOTO 4; PUSH 2 GOTO 4 POP 1 GOTO 2; PUSH 2 GOTO 4 HALT; PUSH 99 GOTO 4

注释

对于第二个例子，指令为：1(PUSH), 2(PUSH), 3(POP), 4(POP), 2(HALT)。

与“皇帝”相比，数据限制的关键差别在于：

- $n \leq 64$;
- 保证输入总是存在一个解;
- 程序应在恰好 k 条指令后停止，而不是在模 998 244 353 意义下。

此页留空。

题目 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 MB

注意：本题是“皇后”的逆版本，数据限制有所不同。

咖波发明了一种有趣的语言，名为 Push-Pop。这种语言是一种解释型语言。解释器初始有一个无限容量的空栈，并执行给定程序的第一条指令。这种语言只有两种指令：

- POP a GOTO x ; PUSH b GOTO y
如果栈非空且栈顶为 a ，则弹出栈顶，并将控制流转移到第 x 条指令（接下来运行第 x 条指令）。
否则，将元素 b 推入栈中，并将控制流转移到第 y 条指令。
- HALT; PUSH b GOTO y
如果栈为空，则在执行此指令后停机。
否则，将元素 b 推入栈中，并将控制流转移到第 y 条指令。

咖波想要升级简易解释器以处理更多指令。给定一个**最多包含 1024 条指令**的程序，计算程序在停机之前执行的步数。

输入

第一行包含一个整数 n ($1 \leq n \leq 1024$)，表示指令的数量。之后 n 行，每行包含一条指令。保证对于每条指令， $1 \leq a, b \leq 1024$, $1 \leq x, y \leq n$ 。

输出

如果程序永远不会停机，则输出 -1 。否则，输出程序执行的指令总数量**模** 998 244 353。

样例

标准输入	标准输出
1 HALT; PUSH 1 GOTO 1	1
5 POP 1 GOTO 2; PUSH 1 GOTO 2 HALT; PUSH 1 GOTO 3 POP 1 GOTO 4; PUSH 2 GOTO 4 POP 1 GOTO 2; PUSH 2 GOTO 4 HALT; PUSH 99 GOTO 4	5
1 POP 1 GOTO 1; PUSH 1 GOTO 1	-1

注释

与“皇后”相比约束条件的关键差别在于：

- $n \leq 1024$;
- 输入数据解可能不存在（程序永远不会停止），在这种情况下输出 -1 ;
- 对于每个会停机的程序，输出指令数量**模** 998 244 353。

此页留空。

题目 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 MB

在夜之城永不停歇的混沌之海中，永恒不变的只有瞬息万变。为了确保旅途安全，边缘行者们依赖于顶尖科技出行。尽管有着公司背景且价格昂贵，你发现德拉曼的人工智能武装出租车仍然是最可靠的交通方式。

今天你需要在夜之城进行一次长途旅行。德拉曼出租车的计费方案如下：

- 起步价为 A 欧元 (EuroDollar)，包含前 X 米的费用。
- 接下来的 Y 米，费用为每米 B 欧元。
- 超过 $X + Y$ 米，费用为每米 C 欧元。

你可以在任何时候进行如下操作：停下来重新呼叫出租车。这样做，你上一部出租车的价格会立刻结算，下一部车的费用将从起步价重新计算。你可以进行这个操作任意次。

现在你想知道使用德拉曼集团的出租车旅行 D 米的最小代价。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 2077$) — 测试数据的数量。接下来是测试数据的描述。

输入包含一行六个整数： A, B, C, X, Y, D ($0 < A, B, C, X, Y, D < 10^{2077}$)，使用没有前导零的十进制整数表示。

保证 T 组数据中 A 的数码数量和不超过 $0x2077$ ；对于 B, C, X, Y, D 同理，亦有相同的限制。

上述约束中使用的值 $0x2077$ 是一个十六进制数，等于十进制的 8311。

输出

对于每个测试数据，输出一行表示使用德拉曼出租车旅行 D 米的最少欧元数。

可以证明，答案是一个正整数。请输出无前导零的十进制形式的整数作为答案。

样例

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	10000000000099799
999 999 1 1 99 9999999999999999	

注释

对于第四个样例，最优解是叫 999 辆出租车，每过一米重新叫一辆。这看起来很奇怪，但你确信这是在夜之城的最佳生活方式。

此页留空。

题目 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 MB

不出户，知天下；不窥牖，见天道。好汉别来无恙？

— 戌狗

知名丹药师戌狗在炼丹时发现精准剔除杂质能够提升丹药的灵韵。在日复一日地炼丹中，他发现了杂质的性质竟然与数学问题有千丝万缕的关系。由于你道行尚浅，戌狗决定以最直白的方式告诉你他需要解决的数学问题，而不是玄之又玄的炼丹问题。

给定两个正整数 $n \leq m$ ，对 $\{1, 2, \dots, m\}$ 的所有大小为 n 的子集，计算以下问题的答案值求和，并对 998 244 353 取模：

- 在 n 个数的集合里，你可以删掉若干个数，使得集合的最小值不等于集合的最大公约数，问满足条件的方案里没有被删的数最多有多少个。如果无解，答案定义为 0。

集合的最大公约数定义为所有元素的共同约数中的最大值。例如，集合 $\{6, 9, 15\}$ 的最大公约数是 3。

输入

输入一行，包含两个整数 m, n ($1 \leq n \leq m \leq 10^5$)。

输出

输出一个整数表示答案，对 998 244 353 取模。

样例

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

注释

对于第一组样例，所有情况列举如下：

- $\{1, 2, 3\}$: $\{2, 3\}$
- $\{1, 2, 4\}$: 无解
- $\{1, 3, 4\}$: $\{3, 4\}$
- $\{2, 3, 4\}$: $\{2, 3, 4\}$

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

此页留空。

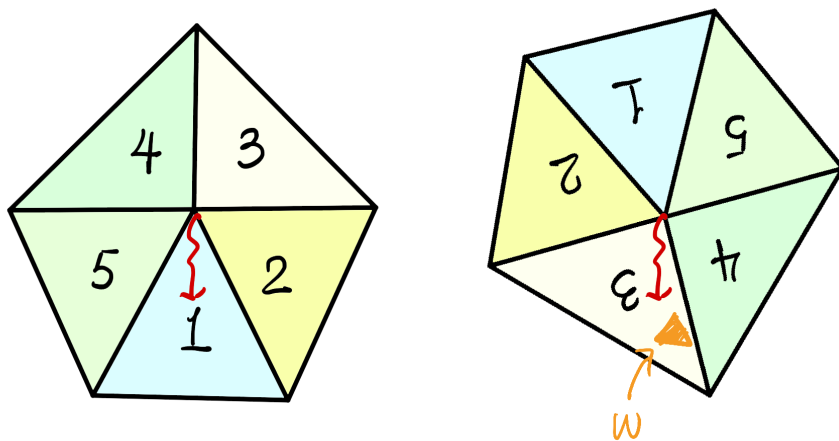
题目 G. 命运之轮

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 MB

教皇和女祭司对宇宙的奥秘充满了好奇。为了探索地球——显然是宇宙的一部分——上的一些奥秘，他们使用了一个奇怪的转盘来研究重力对随机过程的影响。

转盘为凸多边形，凸多边形由旋转中心与每个顶点连线划分为若干三角形区域，每个三角形区域代表一个奖项。转动转盘后，其停留位置的正下方所在的区域为获得的奖项。无偏的转盘旋转中心在其质心，使得每个区域成为最终获奖的位置只和其区域所占角度有关。但不幸的是，这里的转盘可能是有偏的：当旋转中心不在质心时，转盘的结果**永远**都是由旋转中心指向质心。

显然，有偏的转盘很无聊。为了使得各式转盘都有意义，教皇决定用以下方式扰动结果：在轮盘区域内一个**均匀随机**的位置放一枚小磁铁配重（视为放在凸多边形内的一个质点），这么做会改变质心，从而影响结果。



第一个样例：不加磁铁的情况，以及一个可能的磁铁改变结果的情况。

磁铁质量为 w ，转盘每单位面积质量均匀为 1。女祭司想知道，在经过上述扰动后，每个区域成为最终获奖区域的概率。

可以证明，转盘无法唯一确定获奖区域（即旋转中心在质心上或者质心在区域边界上）的概率测度是 0，因此你可以假设不会发生这样的情况。

输入

第一行包含两个整数 n, w ($3 \leq n \leq 100000, 1 \leq w \leq 10^9$)，分别表示转盘的顶点数量和磁铁的质量。

接下来 n 行，每行包含两个整数 x_i, y_i ($|x_i|, |y_i| \leq 30000$)，表示转盘第 i 个顶点的坐标，按边界上逆时针顺序给出。

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，该坐标严格在凸多边形内。

转盘顶点两两不同，但可能存在三点共线：保证当相邻三个点 $u, v = (u \bmod n) + 1, w = (v \bmod n) + 1$ 共线时，点 v 严格在 u 和 w 的连线段上，也即保证了所有内角的大小在 $(0, \pi]$ 里。

保证转盘的面积 S 满足： $\max\{\frac{S}{w}, \frac{w}{S}\} \leq 1000$ 。

输出

输出 n 行，每行一个实数 p_i ，表示由第 i 个顶点和第 $(i \bmod n) + 1$ 个顶点所在直线为边界的区域为最终获奖区域的概率。答案如果具有不超过 10^{-6} 的绝对或相对误差，就会被认为是正确的。

样例

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 36000000 -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

题目 H. 力量

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 MB

人们在日常对话中经常用近似方法来表示数值。无论是讨论时间、金钱还是其他数字，人们通常倾向于四舍五入，以使交流更加简单轻松。例如你和朋友们在一家餐馆里聚餐消费 98 元，此时相比使用准确的数字，许多人会说：“账单总共一百元。”

如果更激进一点，舍入多次，那么最终结果有可能离谱起来。比如，你甚至可以将 145 舍入到 200，因为 145 可以舍入到 150，进而可以舍入到 200；当有人说 2000 时，实际上在舍入之前可能是 2001、1999、1888 甚至 11451。

给定一个数字 x 时，请计算数字 x 在范围 $[0, z]$ 内的不确定性，即在范围 $[0, z]$ 内可以激进舍入到数字 x 的数量。其中，激进舍入被定义为进行如下舍入操作若干次（可以是零次）：

- 设 x 的十进制表示为 $\overline{x_k x_{k-1} \dots x_1}$ ，选择一个下标 $i \in \{1, 2, \dots, k\}$ 。
- 如果 $x_i < 5$ ，令 x 减去 $x_i \cdot 10^{i-1}$ ；
- 否则，令 x 加上 $(10 - x_i) \cdot 10^{i-1}$ 。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 10^5$) — 测试数据的数量。接下来是测试数据的描述。

输入包含一行两个整数: 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，表示范围 $[0, z]$ 内 x 的不确定性。

样例

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

注释

在第二个样例中， $10i + j$ ($i = 0, 1, \dots, 9, j = 5, 6, 7, 8, 9$) 和 10, 11, 12, 13, 14 可以激进舍入到 10。因此，答案是 55。

此页留空。

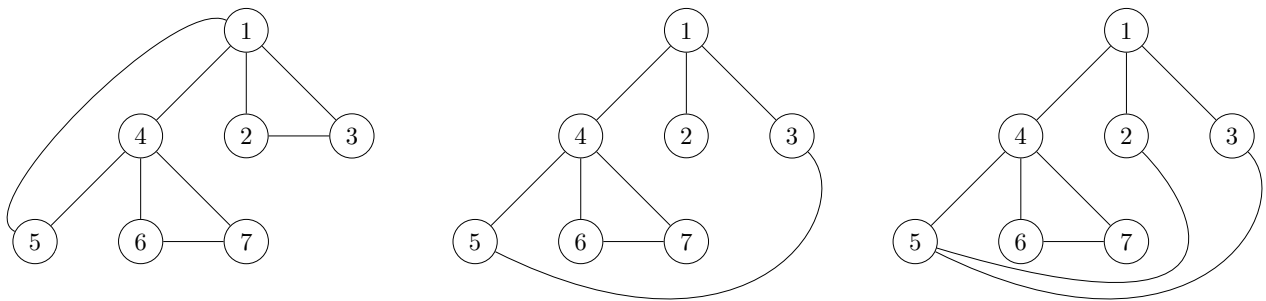
题目 1. 倒吊人

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 MB

克劳黛特·莫雷尔是一位热衷于研究各种植物的植物学家。有一天她在研究玫瑰时，手被玫瑰的刺划伤了。精研百草的她自然知道如何处理伤口，但是更重要的是如何预防这种事情再次发生。为此她想出了一个办法：让玫瑰的刺消失。

玫瑰可以看成是一个 n 个结点的树。为了让玫瑰变得无刺，克劳黛特·莫雷尔可以往图中添加若干条边，只要添加后不形成重边或者自环即可。但注意，她不能往图中添加新的结点。

一个简单图是无刺的，当且仅当每条边恰好出现在一个简单环中。简单环是指不包含任何重复点的环（除了环的开头和结尾是相同的）。下面的图示举例了什么是无刺图，什么不是。



- 左：是无刺图。
- 中：不是无刺图，因为边 $(1,2)$ 不在任何简单环中。
- 右：不是无刺图，因为边 $(1,2)$ 出现在环 $1 \sim 2 \sim 5 \sim 4 \sim 1$ 和 $1 \sim 2 \sim 5 \sim 3 \sim 1$ 中。

现在，克劳黛特·莫雷尔拿出了她的玫瑰，请你分析它们是否能改造成无刺图。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 10^5$) — 测试数据的数量。接下来是测试数据的描述。

第一行包含一个整数 n ($2 \leq n \leq 3 \cdot 10^5$) — 树中结点的数量。

接下来的 $n - 1$ 行中，每行包含两个整数 u_i 和 v_i ($1 \leq u_i, v_i \leq n$)，表示 (u_i, v_i) 是树上的一条边。

保证 T 组数据中 n 的总和不超过 $3 \cdot 10^5$ 。

输出

对于每个测试数据，如果树无法改造为无刺图，则输出 -1 。

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

接下来的 k 行中，每行包含两个整数 x_i 和 y_i ($1 \leq x_i, y_i \leq n$) — 你添加的边。请注意，添加边后，不允许重边或自环。如果有多解，输出任意解。

样例

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	

注释

题目描述中的左图展示了第二个测试数据。

题目 J. 节制

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 MB

这世上啊，唯有花草果木不会负我。

— 辰龙

知名农场主辰龙在种地时发现合理密植能提升植物的产量。
现在农场可以视为一个三维坐标系，植物可以视为坐标系中的一个点。现在有 n 株不同的植物 $A_i = (x_i, y_i, z_i)$ 。对每株植物 A_i 定义密度如下：

- 假设在除了 A_i 之外的植物中还有 a 、 b 和 c 株植物与 A_i 分别有相同的 x 、 y 或 z 坐标。那么， A_i 的密度定义为 $\max\{a, b, c\}$ 。

辰龙种植的植物喜欢内卷，所以他决定移除掉密度较小的植物。请回答，至少要移除多少株植物，才能让剩余植物的密度都大于等于 k 。注意移除植物后，剩余植物的密度可能会变化。特别地，移除掉所有植物始终合法。
请对 $k = 0, 1, \dots, n - 1$ 分别求解。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 2 \times 10^4$) — 测试数据的数量。接下来是测试数据的描述。
第一行包含一个整数 n ($1 \leq n \leq 10^5$) — 植物的数量。
接下来的 n 行中的第 i 行包含三个整数 x_i 、 y_i 和 z_i ($1 \leq x_i, y_i, z_i \leq 10^5$) — 每株植物的坐标。
保证 n 株植物的坐标是不同的。
保证 T 组数据中 n 的总和不超过 2×10^5 。

输出

对于每个测试数据，输出一行 n 个整数，表示 $k = 0, 1, \dots, n - 1$ 的答案。

样例

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	

此页留空。

题目 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 MB

缩写可以让表达变简洁。但是，当两个事物有相同的缩写时，有可能会导致问题。例如，在线搜索“CCPC”时，你会发现不仅有期望的“China Collegiate Programming Contest”（中国大学生程序设计竞赛），还有“China Car Performance Challenge”（中国量产车大赛），“Competition and Consumer Protection Commission”（竞争与消费者保护委员会）等。

然而，没人在写代码的时候开车。为了解决这个问题，你决定给一些经常使用的短语分配不同的缩写。有 n 个由英文字母组成的单词组成的短语，缩写短语的规则是：对短语里的每个单词选择一个非空前缀，然后按顺序拼起来。例如，“ChCoPrCo”和“CCPContest”是“China Collegiate Programming Contest”的缩写，但“CCCP”和“CCPiC”不是。每个短语的缩写互不影响：相同的单词在不同的短语中可以以不同的方式缩写。

给 n 个短语构造一组缩写，使得所有缩写互不相同，且长度总和最小。

输入

输入的第一行包含一个整数 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，用单个空格分隔。每个单词由不超过 128 个小写和大写英文字母组成。保证没有两个短语是完全相同的。

输出

如果没有可能的解决方案，请在一行中输出“no solution”。

否则输出 n 行，第 i 行为第 i 个（按照输入给定的顺序）短语的缩写。如果有多解，输出任意解。

样例

标准输入	标准输出
5 automated teller machine active teller machine active trouble maker always telling misinformation American Teller Machinery	atm atma actm atem ATM
5 Forest Conservation Committee Forum 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	no solution

此页留空。

题目 L. 高塔

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 MB

蓬莱山辉夜是一个喜欢宅在家看视频的 NEET 公主。最近, 她经常在一个名叫 Mikufans 的著名视频分享平台上看视频。

Mikufans 有一个被称为弹幕的功能, 它可以让用户在观看视频的同时留下消息。有时, 同屏的弹幕消息会非常非常多, 让辉夜目不暇接。



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

简单起见, 我们只关注顶部弹幕消息: 顶部弹幕消息会显示在视频屏幕的顶部, 每条恰好占据一行。顶部弹幕同时存在的数量没有上限 (尽管辉夜的屏幕上只会显示前 10^9 行, 但是剩余弹幕仍然会在溢出屏幕的部分被正确维护)。

在视频播放期间, 可能会发生三种类型的事件:

1. 有一个新的用户发送了若干条顶部弹幕消息。这些消息将按顺序放置在最上面的空行中。
2. 某个用户发送的所有弹幕消息消失了, 它们所在的行变为空行。其他的消息不会受到影响, 仍然会保持在原来的位置。
3. 辉夜对一条弹幕消息很感兴趣, 因此她想知道位于某一行的顶部弹幕消息来自于哪个用户。

辉夜每天都有许多视频要看, 她没有精力从头重新看一遍视频, 因此她找到了你。请你帮她找到这些弹幕的发送用户。

输入

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

接下来的 n 行, 每行按顺序描述一个事件, 事件的格式如下:

- 1 k : 有一个新的用户发送了 k ($1 \leq k \leq 10^9$) 条顶部弹幕消息。这个新用户的 ID 是之前未使用过的最小正整数。

- 2 u : 用户 u 发送的所有弹幕消息消失了。保证该 ID 是有效的，并且 ID 为 u 的用户发送的弹幕消息还没有消失。
- 3 1: 辉夜想知道位于从上往下数第 l ($1 \leq l \leq 10^9$) 行的弹幕消息的发送者的 ID。如果该行是空的，则答案定义为 0。

输出

对于每个类型为 3 的事件，在一行中输出答案。

样例

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

题目 M. 审判

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 MB

在一个周末, 青山和她的朋友丹尼尔创造了一个名为大将军的单人游戏, 在这个游戏中, 玩家可以绘制出巨大的画作。

游戏在一个 $n \times m$ 的表格上进行。一个单元格有它的颜色, 可以是红色、蓝色或白色中的一种。最初, 单元格 (a, b) 是红色, 单元格 (c, d) 是蓝色 (这两个单元格不重合), 其他单元格都是白色。我们把 (a, b) 和 (c, d) 称作特殊单元格, 其他的称作非特殊单元格。游戏中, 玩家可以执行一种操作, 其分为三步:

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, 并且必须与 (x, y) 相邻 (即, (x', y') 和 (x, y) 必须有公共边)。
3. 单元格 (x, y) 被涂上单元格 (x', y') 的颜色。

换句话说, 一次操作中, 玩家可以把某个非特殊单元格涂成其某一相邻的非白色单元格的顏色。注意一个单元格可能被涂色多次, 最新的颜色将覆盖早期的颜色。

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

不幸的是, 大将军里外挂肆虐, 作弊者在任何时间任何位置都可以涂色。为了伸张正义, 你决定编写一个裁判程序, 判断给出的状态是否可能是正常游戏中的合法状态, 或者一定有作弊者。

输入

输入包含多个测试数据。第一行包含一个整数 T ($1 \leq T \leq 10^4$) — 测试数据的数量。接下来是测试数据的描述。

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

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

接下来的 n 行, 每行包含 m 个字符。每个字符是 'R', 'B' 或 '.', 分别表示红色单元格, 蓝色单元格和白色单元格。

保证单元格 (a, b) 和 (c, d) 不重合, 并且第 a 行 b 列和第 c 行 d 列的字符分别是 'R' 和 'B'。

保证 T 组数据中 $n \cdot m$ 的总和不超过 250 000。

输出

对于每个测试数据, 如果它是一个合法状态, 则输出 "YES" (不带引号), 否则输出 "NO" (不带引号)。

你可以以任何大小写形式输出字母。

样例

标准输入	标准输出
4	YES
3 3	YES
1 1 1 2	NO
RBB	NO
RRR	
BBR	
6 6	
1 1 6 6	
RRRRRR	
BBBBBR	
BBBBRR	
BRBBBB	
BBBBRR	
BBBBBB	
5 5	
3 3 4 4	
BBR.B	
BBR.B	
RRR.B	
...BB	
BBBB.	
1 5	
1 1 1 3	
RBBBB	

注释

以下图表显示了第一组样例，以及玩家如何在不作弊的情况下达到该状态。皇冠标记特殊单元格。

