

ECUST Programming Contest 2025

正式赛试题册

2025 年 10 月 21 日 13:00 ~ 17:00

试题列表

题号	试题名称	时空限制	注释
A	金色走廊	1s/1GB	
B	眼见为实	1s/1GB	
C	时间谜题	1s/1GB	
D	图腾花园	1s/1GB	
E	秋色堡垒	1s/1GB	
F	十六号塔	2s/1GB	
G	午夜季风	2s/1GB	
H	化方为圆	3s/1GB	交互题
I	斑斓色彩	6s/1GB	
J	克隆之击	1s/1GB	
K	钟塔	1s/1GB	
L	林间小径	2s/1GB	
M	浮点	1s/1GB	
N	净化行动	2s/1GB	
O	荷塘月色	3s/1GB	

试题册共包含 15 道试题和 22 页。

若您的试题册缺页请立刻联系志愿者。

请务必认真阅读位于比赛文件附件中的赛事规则和技术规则。

赛事开始前请勿翻阅试题册，亦请勿触碰鼠标和键盘。

为保证赛事的平衡性，**A** **B** **C** **D** 四题是更加适合新选手的题目，后面的题目难度将随机排序，即不保证题目难度是递增的。

试题 **A** 金色走廊

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 来到了 ECUST 大学！Maddy 步入了传说中的蓝桥杯实验室，听闻了若干位传奇选手的故事之后，她也想成为其中的一员，不过在此之前她想先研究一点现实的问题。
在这所学校，获得一次 ICPC 金奖的队伍可以申请一次 7500 元的奖金，获得一次 ICPC 银奖的队伍可以申请一次 3000 元的奖金，获得一次 ICPC 铜奖的队伍可以申请一次 1500 元的奖金。
而 ICPC 是一个至多三人组队的比赛（可以单人组队也可以双人组队），奖金将在所有队员之间平分。
在 Maddy 的梦境之中，她的队伍有 n 名选手，获得了 g 次金奖，获得了 s 次银奖，获得了 b 次铜奖，请帮她求出她能获得的奖金金额。

输入格式

一行输入 4 个整数 n, g, s, b ($1 \leq n \leq 3, 0 \leq g, s, b \leq 7$)，分别表示 Maddy 队伍的人数，获得金、银、铜奖的次数。
保证 $g + s + b \leq 7$ 。

输出格式

一行输出一个整数 m ，表示 Maddy 获得的奖金数量（单位：元）。

样例 1

standard input	standard output
3 2 1 2	7000

样例 1 解释

这时 Maddy 队伍获得两个金奖，获得 15000 元奖金，获得一个银奖，获得 3000 元奖金，获得两个铜奖，获得 3000 元奖金，总计 21000 元，均分后 Maddy 得到 7000 元。

样例 2

standard input	standard output
1 0 0 4	6000

提示

在程序设计竞赛中，参赛选手在提交的程序中通常不需要，也不应该在程序的开头输出提示性文字，包括且不限于输出 请输入两个整数：之类的文字，这将导致所提交的程序被判为答案错误或导致其他非预期的评测结果。

试题 **B** 眼见为实

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 和 Baddy 展开了心灵之战!

Baddy 内心中隐藏了一个数字 a , 这个数字 a 满足如下的规则:

- a 是自然数。
- $l \leq a < l + 6$.
- a 除以 2 的余数为 m_2 , a 除以 3 的余数为 m_3 .

现在 Maddy 知道了 l, m_2, m_3 , 请帮她找到一个 Baddy 心中所想的 a 的可能值, 如果不存在可能的答案, 请报告无解。

输入格式

一行输入 3 个整数 l, m_2, m_3 ($0 \leq l \leq 100, 0 \leq m_2 < 2, 0 \leq m_3 < 3$), 分别表示 Baddy 所想数字的范围, 除以 2 的余数, 除以 3 的余数。

输出格式

如果存在合法的答案, 则一行输出一个整数 a , 表示 Baddy 心中所想的数字, 如果有多个合法的答案则你可以输出其中任意一个。

如果不存在可能的答案, 则输出一行一个整数 -1 。

样例 1

standard input	standard output
0 1 2	5

样例 1 解释

依次验证上述性质:

- 5 是自然数。
- $0 \leq 5 < 0 + 6$.
- 5 除以 2 的余数为 1, 5 除以 3 的余数为 2.

因此 5 满足 Baddy 心中所想的条件, 是一个可能值。

样例 2

standard input	standard output
63 1 1	67

提示

0 是自然数。

试题 C 时间谜题

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 的电子表被击碎了。令人惊讶的是上面的数字飞了出来，落在了地上。现在她从地上挑出了两个数字，请帮助她用这两个数字重新组成一个新的 24 小时制的时间 *Time*，或告诉她这是不可能的。

所求的 24 小时制的时间是一个字符串，其格式形如 *HH:MM*，并满足如下要求：

- *HH*, *MM*，是不大于两位的非负整数，若只有一位则应在前面补 0。
- *HH* 的取值范围是 $0 \leq HH \leq 23$, *MM* 的取值范围是 $0 \leq MM \leq 59$ 。
- *HH*, *MM* 和所给出的两个数字应不重不漏对应，注意你可以交换所给出的两个数字的顺序。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^4$), 表示数据组数。

每组测试数据输入的唯一一行输入两个整数 a 和 b ($0 \leq a, b \leq 99$), 表示 Maddy 所选的两个数字。

输出格式

对于每组测试数据，若可以用所给出的两个数字组成一个 24 小时制的时间，则输出一行一个字符串 *Time*，表示你所构成的 24 小时制的时间，如果有多个合法的答案则你可以输出其中任意一个。

反之则输出一行一个整数 -1。

样例 1

standard input	standard output
5	09:41
9 41	-1
9 61	09:50
50 9	00:06
0 6	-1
25 35	

样例 1 解释

对于第 1 组样例，可以发现：

- 09, 41 是不大于两位的非负整数，并且在一位数前补了 0。
- 09, 41 分别满足对应位数的大小限制。
- 09, 41 与所给的 9, 41 不重不漏一一对应。

注意 41:09 不是一个合法的答案，因为这不满足对应位数的大小限制。

对于第 2 组样例，容易发现无论如何交换顺序，都无法组成一个合法的 24 小时制的时间。

试题 **D** 图腾花园

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

欢迎来到 ECUSTPC 2025 的赛场！
在这场 4 个小时的比赛中，你将与 Maddy 携手共进，共同面对这些有趣的试题！
Maddy 给你的下一个任务是针对特定的字符串 **ECUST** 进行操作。你需要生成所有在不区分大小写的前提下与 **ECUST** 相等的字符串。
你所生成的字符串必须是两两不同的，并且需要按照 ASCII 码的字典序进行从小到大排列并输出。

输入格式

本题没有输入。
请注意，技术上本题会提供一个随机的一行字符串作为输入，你可以不用读入这个字符串，只需要提供正确的输出即可。

输出格式

输出若干行，每行包含一个和 **ECUST** 在不区分大小写的前提下相等的字符串，你需要让这些字符串按照 ASCII 码的字典序排序。

样例 1

本题没有输入，输出将仅仅展示一个输出的格式示范，而非完整的答案输出，且本题的样例不会在附加文件中出现。

standard input	standard output
NO_INPUT	ECUST
	ECUS ^t
	ECUs ^T
	<lines omitted>
	ecust

提示

字符串 a 的 ASCII 码字典序小于字符串 b ，当且仅当满足以下条件之一：

- a 是 b 的前缀。
- 在 a 和 b 第一个不同的位置，满足该位置上 a 的字符的 ASCII 码值小于 b 的 ASCII 码值。

两个字符串在不区分大小写的前提下相等，当且仅当将两个字符串所有的大写字母都转成对应的小写字母后相等。例如，**ECUST**, **ecust**, **eCuSt** 在不区分大小写的前提下是相等的。
数字 **0** 的 ASCII 码为 48, 大写字母 **A** 的 ASCII 码为 65, 小写字母 **a** 的 ASCII 码为 97.

试题 E 秋色堡垒

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 来到了秋天的堡垒之前，在此她采集了 n 个食材，这些食材的新鲜度分别为 $a_1, a_2, \dots, a_n$ 。Maddy 会用这些食材做若干天的晚饭，特别的，因为她只有两个技艺值分别为 x, y 的食谱，她每天会做且只会做两道菜，少一道都不行哦。
每一天做饭的流程如下：

- Maddy 可以以任意顺序选取两个不同未被选择过的食材，记其美味值分别为 a, b ，随后对其进行烹饪，每个食材对应一个食谱，所制作出的菜肴的美味值分别为 ax 和 by ，则当天晚饭总的美味值为 $ax + by$ ，注意 Maddy 可以以任意顺序选择食材对应的食谱，且每天晚饭必须选择 2 个不同的食材制作 2 道菜肴。

请帮助 Maddy 选取做晚饭的天数并求出在最优的情况下，这几天晚饭的总美味值 D 。（当然如果都太难吃了也可以不做的，换言之可以取天数为 0。）

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$)，表示数据组数。
每组测试数据输入的第一行输入 3 个整数 n, x, y ($2 \leq n \leq 10^5, -10^6 \leq x, y \leq 10^6$)，分别表示食材的个数以及两个食谱的技艺值。
随后一行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($-10^6 \leq a_i \leq 10^6$)，分别表示食材的新鲜度。
保证所有测试数据输入中的 $\sum n \leq 3 \times 10^5$ 。

输出格式

每组测试数据输出一行一个整数 D ，表示在最优的情况下，这几天晚饭的总美味值。

样例 1

standard input	standard output
6	35
4 1 2	56
4 5 6 7	0
5 2 8	23
0 -1 -2 -6 7	23
6 -1 -2	165
1 2 3 4 5 6	
6 3 4	
-1 0 0 1 2 3	
6 -3 -4	
1 0 0 -1 -2 -3	
5 1 -10	
-9 2 -5 3 -7	

样例 1 解释

对于第 1 个样例，我们考虑做 2 天的晚饭，其中：

- 第一天选用新鲜度分别为 7 和 5 的食材分别对应技艺值为 2, 1 的菜谱，可以烹饪出美味值为 $7 \times 2 + 5 \times 1 = 19$ 的晚饭；

- 第二天选用新鲜度分别为 6 和 4 的食材分别对应技艺值为 2, 1 的菜谱，可以烹饪出美味值为 $6 \times 2 + 4 \times 1 = 16$ 的晚饭。

总的美味值为 35.

对于第 2 个样例，我们考虑做 1 天的晚饭，其中选用新鲜度分别为 7 和 0 的食材分别对应技艺值为 8, 2 的菜谱，可以烹饪出美味值为 $7 \times 8 + 0 \times 2 = 56$ 的晚饭。

对于第 3 个样例，我们考虑不做晚饭，因为任何情况下做出的晚饭的美味值均是负的。

试题 **F** 十六号塔

运行时间限制 2 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 来到了一座褪色的高塔面前，高塔下方的石碑上刻着一个数字 n 。
Maddy 发现旁边的石头上有一个机关，她按下了机关，发现石碑上的数字如下变化：

- 将这个数字 n 的十进制表达中的每一位都拆开后取一个平方，再按原先顺序连成一个新的数字 n' 。

Maddy 很好奇，她按了 m 次机关，她记录下了原本的 n 以及这 m 次按下机关之后的得数，共 $m+1$ 个数字 $n, n', n'', \dots, n^{(m)}$ 。

Maddy 希望知道上述 $m+1$ 个得数之和对 9 取模的值 ans 是多少，请帮助她求出。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^3$), 表示数据组数。

每组测试数据输入的唯一一行输入两个整数 n 和 m ($1 \leq n, m \leq 10^9$) 分别表示石碑上所刻的数字和 Maddy 所按机关的次数。

输出格式

每组测试数据输出一行一个整数 ans , 表示上述得数之和对 9 取模的答案。

样例 1

standard input	standard output
4	2
1 100	6
2 4	6
74700 1	5
2279 1	

样例 1 解释

对于第 1 组样例, 1 在按下机关后总还是 1, 每次的得数都为 1, 总和为 $1 \times 101 = 101$, 取模为 $101 \bmod 9 = 2$ 。

对于第 2 组样例, 数字变化如下: $2 \rightarrow 4, 4 \rightarrow 16, 16 \rightarrow 136, 136 \rightarrow 1936$, 这 5 个数字的和对 9 取模为 $(2 + 4 + 16 + 136 + 1936) \bmod 9 = 2094 \bmod 9 = 6$ 。

对于第 3 组样例, 数字变化如下: $74700 \rightarrow 49164900$ 。

对于第 4 组样例, 数字变化如下: $2279 \rightarrow 444981$ 。

提示

一个是对 9 取模表示的是这个数除以 9 之后产生的余数。

试題 G 午夜季风

运行时间限制 2 seconds
内存使用限制 1 gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 面前有 n 条河豚鱼，每条河豚鱼有一个大小值 a_i ，相差甚远的河豚鱼在一起会爆炸！因此 Maddy 选择了让它们排成一排。记排成一排之后从左到右河豚鱼的大小分别为 $a'_1, a'_2, \dots, a'_n$ ，则 Maddy 会称呼这排河豚鱼的爆炸度为

$$B = \sum_{i < j, |i-j| > 1} |a'_i - a'_j|$$

现在河豚鱼在午夜发生了改变！具体而言共有 m 次转变，每次转变会有一条河豚鱼 id 的大小增加 $delt$ ($delt < 0$ 时为河豚鱼大小减小 $|delt|$)，在每次转变之后 Maddy 希望你能找到将河豚鱼排成一排的方法使得其爆炸度最小，你要求出此时的爆炸度 B 。

注意每一次转变是具有后效性的，即前一次的转变效果会保留下来并影响后续的计算和转变。
注意河豚的大小可以是负数。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$)，表示数据组数。

每组测试数据输入的第一行输入两个整数 n 和 m ($2 \leq n \leq 10^5, 1 \leq m \leq 10^5$)，分别表示河豚鱼的个数和转变的次数。

随后一行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($-10^8 \leq a_i \leq 10^8$)，分别表示河豚鱼的大小，注意河豚鱼大小可以是负的。

随后 m 行每行依次输入两个整数 id 和 $delt$ ($1 \leq id \leq n, -10^8 \leq delt \leq 10^8$)，表示每次转变后的河豚鱼编号和河豚鱼大小变化值。

保证单组测试数据中， $\sum |delt| \leq 10^8$ ，所有测试数据输入中所有的 $\sum n, \sum m \leq 3 \times 10^5$ 。

输出格式

每组测试数据对于每次转变输出一行一个整数 B ，表示转变之后使爆炸度最小的河豚鱼排列方法中的爆炸度。

样例 1

standard input

```
1
4 3
1 2 3 4
4 0
2 5
1 -2
```

standard output

```
3
6
8
```

样例 1 解释

在第一次转变后河豚鱼的大小仍然为 $\{1, 2, 3, 4\}$ 此时最优排列方法为 $\{2, 4, 1, 3\}$ ，爆炸度为 $B = |a_1 - a_3| + |a_1 - a_4| + |a_2 - a_4| = |2 - 1| + |2 - 3| + |4 - 3| = 3$ 。

在第二次转变后河豚鱼的大小为 $\{1, 7, 3, 4\}$ 此时最优排列方法为 $\{3, 7, 1, 4\}$ ，爆炸度为 $B = |a_1 - a_3| + |a_1 - a_4| + |a_2 - a_4| = |3 - 1| + |3 - 4| + |7 - 4| = 6$ 。

在第三次转变后河豚鱼的大小为 $\{-1, 7, 3, 4\}$ 此时最优排列方法为 $\{3, 7, -1, 4\}$ ，爆炸度为 $B = |a_1 - a_3| + |a_1 - a_4| + |a_2 - a_4| = |3 - (-1)| + |3 - 4| + |7 - 4| = 8$ 。

试题 **H** 化方为圆

运行时间限制 3 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

这是一道交互题。

题目描述

Maddy 和 Baddy 在玩猜机关的游戏！

机关是一个二维图形，Maddy 现在在二维坐标平面内隐藏了一个机关，这个机关可能是下面两种中的一种：

- 弹球，这是一个格点圆，其圆心位于 $(0, 0)$ ，圆上有至少一个正格点（至少一个横纵坐标为正整数的点），且其横纵坐标绝对值小于 10^5 。
- Kevin，这是一个格点正方形，其中心位于 $(0, 0)$ ，每个顶点都在格点上（横纵坐标为整数），且其横纵坐标绝对值小于 10^5 ，保证正方形不是退化的，也即其边长不小于 1。

但 Baddy 不知道这个机关是什么，她可以每次询问 Maddy 一个问题：

- 格点 (x, y) 是否在这个机关所表示的图形的之外（边界算在图形内部）？
- Maddy 会告诉她这个点在图形之外或是在图形之内，如果点在边界上，Maddy 会告诉 Baddy 点在图形内部。

Baddy 可以至多询问 35 次问题，她需要告诉 Maddy：

- 这个机关是 Kevin，弹球，还是存在两种可能的答案？

注意，回答你的答案不算在 35 次询问之内。

交互格式

第一行输入一个整数 T ($1 \leq T \leq 10^3$)，表示测试数据的数量。

每组数据会直接开始交互过程，每一次交互你可以以下面的格式向交互器询问问题：

- 输出一行，首先输出一个字符 `?` 表示你询问一次，随后输出两个整数 x 和 y 表示你需要询问点 (x, y) 是否在机关外，你需要保证 $0 \leq |x|, |y| \leq 2 \times 10^5$ 。
- 若你的输入合法并且没有超出询问限制，交互器会给出一个字符串 Q ，若 $Q = \text{IN}$ ，则表示 (x, y) 在机关所表示的图形之内或在边上，若 $Q = \text{OUT}$ 则表示 (x, y) 在机关所表示的图形外部。
- 若你询问的问题超出了对应的询问次数限制，交互器会输出一个整数 -1 并结束你的交互进程，此时你会获得答案错误的评测结果。

若你已经得到答案的话，请按照以下的格式输出答案：

- 输出一行，首先输出一个字符 `!`。
- 随后输出一个字符串 S ， S 需为 `Bumper`，`Kevin` 或 `NotConfirm`，分别表示机关是弹球，机关是 Kevin，或存在两种可能的答案。

若你的答案正确则交互器会输出一行一个字符串 `Correct!`，反之则会输出一行一个整数 -2 并立即结束，此时你会获得答案错误的评测结果。

在完成一组交互之后你应该立刻开始下一组交互的询问，若已是最后一组数据你可以安全退出。

交互器不是自适应的，这意味着在每组数据开始之前机关就已经确定，在交互过程中不会改变。

每次输出都需要输出换行并且刷新缓冲区，否则你可能会得到意料之外的结果。

如果接收到了 -1 或 -2 两种交互器的错误信息，请尽快结束你的交互进程，否则你可能会得到意料之外的结果（如超时）。

为了刷新缓冲区，你可以：

- 对于 C 或者 C++, 使用 `fflush(stdout)` 或 `cout.flush()`。
- 对于 Java 或 Kotlin, 使用 `System.out.flush()`。
- 对于 Python, 使用 `stdout.flush()`。

样例 1

interactor input	contestant output
1	? 1 0
IN	? 0 1
IN	? 100 100
OUT	? 0 2
IN	! Bumper
Correct!	

提示

请注意样例中的交互过程仅供参考，实际交互过程并不唯一，且这一示例的交互过程不一定是可行的或是最优的，本题的样例不会在附加文件中出现。

存在两种可能的答案当且仅当：

- 存在一个 Kevin 形状 K 覆盖的格点集合 S_K 和一个弹球形状 B 覆盖的格点集合 S_B 和当前我们隐藏机关的格点集合 S_{hidden} 完全相同。

这时候你不可能通过你的询问区分两种形状，你应该输出 **NotConfirm**。

但是如果不存在这样的一对机关使得覆盖集合相同，那你应该可以用你的询问唯一确定机关种类，这时输出 **NotConfirm** 会被视为答案错误。

试题 I 斑斓色彩

运行时间限制 6 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 找到了一把崭新的剑，她决定用一些宝石装饰她的剑。在神秘的 Celeste 山地区，有 C 块各不相同的宝石散落在各处。

Maddy 希望她的剑看起来非常五彩斑斓，因此她至少需要用 k 块宝石来装饰她的剑。

在出发之前，她拿到了一张地图，地图上记载了 Celeste 山地区的详细地貌，其上如下记载：

- Celeste 山地区被分为了一个 $n \times m$ 的网格，分别用坐标 $(1, 1)$ 到 (n, m) 标示每一子地块的情况。
- 我们称一块坐标为 (i, j) 的子地块的相邻子地块为 $(i + 1, j)$ 、 $(i - 1, j)$ 、 $(i, j + 1)$ 和 $(i, j - 1)$ ；注意相邻子地块亦需要在 Celeste 山地区之中，即相邻子地块的坐标 (i, j) 亦应满足 $1 \leq i \leq n, 1 \leq j \leq m$ 。
- 每一块子地块的地形为陆地、水或者岩浆。
- 其中 C 个子地块中有宝石，且宝石只可能在地形为陆地或者水的子地块中。

Maddy 从一个给定的子地块 S 出发，她需要在这区域收集至少 k 块宝石。她的行动需要消耗精力值，行动和精力值的计算规则如下：

- Maddy 初始位于所给定的子地块 S ，保证这一子地块的地形为陆地或者水且没有宝石。
- Maddy 每次可以移动到一个相邻的、地形为陆地或者水的子地块中；她不能进入岩浆子地块。
- 若本次移动的起点与终点均为“陆地”，则这次移动不消耗精力值。
- 否则（即本次移动涉及“水”，无论是从水出发，或移动到水，或水 $\leftrightarrow$ 水），这次移动将会消耗 1 点精力值。

请告诉 Maddy 她至少需要消耗多少精力值 stm 才能收集至少 k 块宝石，或告诉她不可能收集到 k 块宝石。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 100$)，表示测试数据的数量。

每组测试数据的第一行输入 4 个整数 n, m, C, k ($1 \leq n, m \leq 2 \times 10^3, 1 \leq k \leq C \leq 15$)，分别表示网格的长与宽、地图上共有的宝石数量以及 Maddy 需要的宝石数量。

随后一行输入两个整数 S_x 和 S_y ($1 \leq S_x \leq n, 1 \leq S_y \leq m$)，表示 Maddy 的起始位置的横纵坐标。

随后输入 n 行，每行一个长度为 m 的字符串 $S_1, S_2, \dots, S_n$ ，其中 $S_i = \overline{s_{i,1}s_{i,2}\dots s_{i,m}}$ 描述第 i 行各子地块的地形。对任意 $1 \leq i \leq n, 1 \leq j \leq m$ ：

- 若 $s_{i,j} = 0$ ，表示该子地块为陆地；
- 若 $s_{i,j} = 1$ ，表示该子地块为水；
- 若 $s_{i,j} = 2$ ，表示该子地块为岩浆。

随后 C 行，每行输入两个整数 x, y ($1 \leq x \leq n, 1 \leq y \leq m$)，表示其中一块宝石所位于的子地块坐标。

保证：单组测试数据中所有宝石位置与 Maddy 起始位置均不在岩浆上，且这些坐标两两不同。保证所有测试数据中的 $n \cdot m$ 之和不大于 4×10^6 ，且至多有 5 组测试数据满足 $C > 10$ 。

输出格式

对于每组测试数据：

- 若 Maddy 可以收集到至少 k 块宝石，输出一行一个整数 stm ，表示她至少需要消耗的精力值；
- 否则输出一行一个整数 -1 。

样例 1

standard input

```
2
3 5 3 2
1 1
00000
11211
11011
1 4
3 1
3 5
1 5 1 1
1 1
01121
1 5
```

standard output

```
2
-1
```

样例 1 解释

对于第 1 个样例，其示意图如下：

	1	2	3	4	5
1	起点			宝石 1	
2					
3	宝石 2				宝石 3

其中答案的路径是 $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (1,4) \rightarrow (1,3) \rightarrow (1,2) \rightarrow (1,1) \xrightarrow{1 \text{ 精力值}} (2,1) \xrightarrow{1 \text{ 精力值}} (3,1)$.

试题 J 克隆之击

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 和 Baddy 决定通过智慧的方法一较高下！

Maddy 和 Baddy 初始有一个含有 n 个正整数的多重集合 $S = \{a_1, a_1 + d, a_1 + 2d, \dots, a_1 + (n-1)d\}$, 她们决定玩一个游戏, 游戏规则如下:

- 由 Maddy 开始操作, Maddy 和 Baddy 轮流操作。
- 每位玩家操作时需要 S 中加入一个正整数。
- 当每位玩家都操作了 m 次后, 游戏结束,
- Maddy 的目标是让最终 S 的 k -连通数尽可能大, Baddy 的目标是让最终 S 的 k -连通数尽可能小。

其中的 k -连通数定义如下:

- 先将 S 排序, 对于每组相邻的数, 若其差大于 k , 则将连通数增加 1, 连通数初始为 1.
- 一个等价的定义是在图上把 S 上的元素作为顶点, 差不大于 k 的顶点之间连边, k -连通数即为连通块的数量。

假设 Maddy 和 Baddy 都是无比智慧的, 在进行游戏时总会选择最优的策略, 请帮助她们求出游戏结束时 k -连通数的值。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 100$), 表示数据组数.

每组测试数据输入的唯一一行输入 5 个整数 n, a_1, d, m 和 k ($1 \leq n, a_1, m \leq 100, 0 \leq d, k \leq 100$), 分别表示初始 S 的大小, 等差数列的首项和公差, 游戏的轮数, 参量 k 的值。

输出格式

每组测试数据输出一行一个整数 ans , 表示在 Maddy 和 Baddy 选择最优的策略的情况下, 游戏结束时 S 的 k -连通数的值。

样例 1

standard input	standard output
4	3
3 2 3 2 2	6
4 3 2 5 1	6
4 1 1 5 2	12
9 7 8 3 0	

样例 1 解释

对于第 1 个样例, 初始的 $S = \{(2+0 \times 3), (2+1 \times 3), (2+2 \times 3)\} = \{2, 5, 8\}$, 下给出一个可能的双方策略:

- Maddy 加入一个 13, $S = \{2, 5, 8, 13\}$;
- Baddy 加入一个 3, $S = \{2, 3, 5, 8, 13\}$;

- Maddy 加入一个 18, $S = \{2, 3, 5, 8, 13, 18\}$;
- Baddy 加入一个 7, $S = \{2, 3, 5, 7, 8, 13, 18\}$.

排序后相邻差依次为 $D_1 = 1, D_2 = 2, D_3 = 2, D_4 = 1, D_5 = 5, D_6 = 5$, 比 $k = 2$ 大的共有 2 个, k -连通数为 3.

试题 **K** 钟塔

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 准备重建一座钟塔...

Maddy 有一个长度为 n 的序列 $\{a_i\}$, 现在她希望通过最少的操作来让一个长度为 n 初始全为 0 的序列 $\{h_i\}$ 变为 $\{a_i\}$, 其中每次操作的规则如下:

- Maddy 选择一个坐标 $k(1 \leq k \leq n)$ 和一个方向 $d \in \{L, R\}$.
- 若 $d = L$, 对于所有 $1 \leq i \leq k$, 令 h_i 变为 $|i - k| + 1$.
- 若 $d = R$, 对于所有 $k \leq i \leq n$, 令 h_i 变为 $|i - k| + 1$.
- 注意每次操作都会完全覆盖所对应的区间。

请帮助她求出最少所需的操作次数, 若不可能通过上述操作达成目的, 请报告无解。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$), 表示数据组数。

每组测试数据的第一行输入一个整数 n ($1 \leq n \leq 10^5$), 表示序列长度。

随后一行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$), 表示 Maddy 希望构造得到的序列 $\{a_i\}$ 。

保证所有测试数据输入中的 $\sum n \leq 3 \times 10^5$ 。

输出格式

对于每组测试数据, 若可以通过上述操作达成目的, 则输出一行一个整数, 表示 Maddy 所需要进行最少的操作次数。

反之则输出一行一个整数 -1 。

样例 1

standard input	standard output
4	3
6	-1
1 2 1 1 2 3	6
3	6
3 3 3	
6	
1 1 1 1 1 1	
11	
1 2 1 3 4 1 2 1 1 2 3	

样例 1 解释

对于第 1 个样例, 其操作过程可表示为:

1. 选择 $k = 4, d = R$, 序列 $\{h_i\} = \{0, 0, 0, 1, 2, 3\}$;
2. 选择 $k = 3, d = L$, 序列 $\{h_i\} = \{3, 2, 1, 1, 2, 3\}$;
3. 选择 $k = 1, d = L$, 序列 $\{h_i\} = \{1, 2, 1, 1, 2, 3\}$ 。

容易发现这即是最少步数的操作序列了。

对于第 2 个样例, 容易发现无论 Maddy 进行任何操作, h_2 都不可能变为 3。

试题 L 林间小径

运行时间限制 2 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 在树林里迷路了..., 她看到了一棵奇特的树。
在她面前是一个大小为 n 的树, 这是一个由 n 个顶点 $n-1$ 条边组成的无向无环连通图 T 。
但是 Maddy 一不留神, 树的形态发生了改变, 有两个不相邻的不同顶点之间通过了一条新生的枝丫连了起来! 也就是说, 树 T 变成了一个由 n 个顶点 n 条边组成的无向连通图 G' 。
Maddy 为了考验你, 她给出了原本树的形态以及所有顶点的 $\sum D_i$, 其中 $\sum D_i$ 为顶点 i 在这个新的图 G' 上到每一个其他顶点的距离之和, 也即 $\sum D_i = \sum_{j \in V} D_{G'}(i, j)$ 。
你需要告诉 Maddy 新生枝丫所连接的两点。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$), 表示数据组数。
每组测试数据输入的第一行输入一个整数 n ($3 \leq n \leq 10^5$), 表示图的顶点数量。
随后 $n-1$ 行每行输入两个整数 u 和 v ($1 \leq u, v \leq n, u \neq v$), 表示树 T 上存在一条 u 到 v 的边。
随后一行输入 n 个整数 $\sum D_1, \sum D_2, \dots, \sum D_n$ 分别表示在 G' 上每个顶点到其他顶点的距离之和。
保证所有测试数据输入中的 $\sum n \leq 3 \times 10^5$, 且每组数据输入的图构成一棵树, 所隐藏的边的两点在原图不同且上不相邻。

输出格式

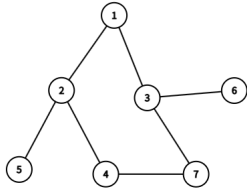
每组测试数据输出一行两个整数 x 和 y , 表示新生枝丫所连的两个顶点。
如果有多个合法的答案则你可以输出其中任意一个。(例如 1 2 合法时 2 1 也合法)

样例 1

standard input	standard output
1	4 7
7	
1 2	
1 3	
2 4	
2 5	
3 6	
3 7	
10 10 10 11 15 15 11	

样例 1 解释

下面的图展示了 G' 的形态。



提示

图上两点的距离 $D_{i,j}$ 定义为所有从 i 开始到 j 结束的路径中经过边数的最小值。

试题 M 浮点

运行时间限制 1 second
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 碰到了一台机器，在机器面前她需要报出一长串数，这样她才可以获得水晶之心！
机器首先告诉了 Maddy 一个整数 n ，具体而言，这串数的要求如下：

1. 这串数由 n 个在 $[0, 2^{30})$ 之内的整数组成，记其为 $a_1, a_2, \dots, a_n$.
2. 对于所有的 $1 \leq i < n$ ，满足有 $a_i < a_{i+1}$ 且 $a_i \text{ xor } a_{i+1} < a_i \text{ and } a_{i+1}$.

请帮助 Maddy 求出一串这样的数，或告诉 Maddy 答案并不存在。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^3$), 表示测试数据的数量。
每组测试数据的唯一一行输入一个整数 n ($2 \leq n \leq 10^6$), 表示机器告诉 Maddy 的数。
保证所有测试数据输入的 $\sum n \leq 10^6$.

输出格式

对于每组测试数据中若存在这样的一串数则输出一行 n 个整数，表示一串满足机器要求的数，若不存在则输出一行一个整数 -1 。
如果有多个合法的答案则你可以输出其中任意一个。

样例 1

standard input	standard output
4	9 15
2	4 5 6
3	9 10 12 15
4	8 9 10 11 12
5	

样例 1 解释

对于第 1 个样例， $a_1 \text{ xor } a_2 = 9 \text{ xor } 15 = 6$, $a_1 \text{ and } a_2 = 9 \text{ and } 15 = 9$, 因此有 $a_1 \text{ xor } a_2 < a_1 \text{ and } a_2$, 且容易发现 $a_1 < a_2$ 。
对于第 2 个样例，

- $a_1 \text{ xor } a_2 = 4 \text{ xor } 5 = 1$, $a_1 \text{ and } a_2 = 4 \text{ and } 5 = 4$, 因此有 $a_1 \text{ xor } a_2 < a_1 \text{ and } a_2$, 且容易发现 $a_1 < a_2$ 。
- $a_2 \text{ xor } a_3 = 5 \text{ xor } 6 = 3$, $a_2 \text{ and } a_3 = 5 \text{ and } 6 = 4$, 因此有 $a_2 \text{ xor } a_3 < a_2 \text{ and } a_3$, 且容易发现 $a_2 < a_3$ 。

提示

对于两个整数 a 和 b , $a \text{ xor } b$ 表示按位取异或，即结果的二进制表示的一位为 1 当且仅当原数的二进制表示在该位上有且仅有一个 1。
对于两个整数 a 和 b , $a \text{ and } b$ 表示按位取与，即结果的二进制表示的一位为 1 当且仅当原数的二进制表示在该位皆为 1。

试题 **N** 净化行动

运行时间限制 2 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 将操控一个国王来涤清 Baddy 的邪恶战车军团...

具体而言，她们现在在进行一个游戏，游戏在无限大的国际象棋棋盘上进行。

Baddy 操控了 n 个黑色车，这些车在游戏过程中固定在它们的初始位置，不会移动。它们分别部署在 (x_1, y_1) , $(x_2, y_2), \dots, (x_n, y_n)$ 格子上。

Maddy 选择了一个位置 (X, Y) 来部署她的白色王，王的走法是可以走到与之相邻的八个格子里，例如若王在 (p, q) 格子，则 $(p-1, q-1)$ 、 $(p-1, q)$ 、 $(p-1, q+1)$ 、 $(p, q-1)$ 、 $(p, q+1)$ 、 $(p+1, q-1)$ 、 $(p+1, q)$ 、 $(p+1, q+1)$ 格子都可以到达。

一个黑色车棋子的攻击范围是与其在同一行或是同一列，且他们中间没有其他棋子阻挡的白色棋子，自身所在格不算在攻击范围内。

Maddy 的目标是操控白色王吃掉 Baddy 的所有黑色车。具体而言：

- 白色王可以遵循上面的规则任意移动，但在移动的过程中不能进入黑色车的攻击范围，否则白色王会被黑色车吃掉。
- 若其移动到了一个黑色车所占据的格子，那这个黑色车将会被吃掉，这个黑色车后续将不再能攻击白色王。

Maddy 的目标是吃掉 Baddy 的所有 n 辆车。请你判断：是否存在一种吃子顺序和王的移动路径，使得王可以最终吃掉所有的车？

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$)，表示数据组数。

每组测试数据的第一行输入一个整数 n ($1 \leq n \leq 10^5$)，表示黑色车的数量。

随后一行输入两个整数 X 和 Y ($1 \leq X, Y \leq 10^9$)，表示 Maddy 操控的白色王的起始位置。

随后 n 行，每行输入两个整数 x_i 和 y_i ($1 \leq x_i, y_i \leq 10^9$)，分别表示 Baddy 部署的第 i 辆黑色车的位置的坐标。

保证所有测试数据输入中的 $\sum n \leq 3 \times 10^5$ ，且每组测试数据中所有黑色车的坐标 (x_i, y_i) 互不相同，且在游戏开始时，白色王不在任何一个黑色车的攻击范围内。

输出格式

对于每组测试数据，如果白色王可以最终吃掉所有的黑色车，则输出一行一个字符串 YES，否则输出一行一个字符串 NO。

注意评测时不会区分 YES 和 NO 的大小写，换言之当答案是肯定的时候输出 yes, YES, Yes, YeS 等都会被认为是正确的。

样例 1

standard input

```
4
1
4 6
5 7
2
1 1
8 2
8 9
2
4 1
2 3
7 8
4
7 10
2 11
5 3
9 20
14 6
```

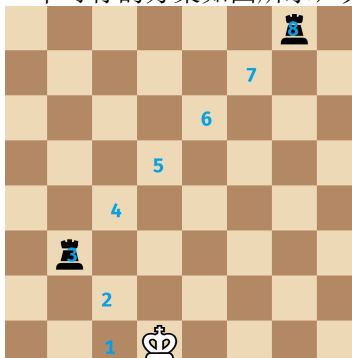
standard output

```
YES
NO
YES
NO
```

样例 1 解释

对于第 1 组样例，王可以一步吃掉黑车。

对于第 3 组样例，棋盘的示意图如下，第一维是从左到右递增，第二维是从下到上递增，左下角格子是 (1,1):
一个可行的方案如图所示，其中蓝色数字表示王在对应步数所在的位置。



试题 0 荷塘月色

运行时间限制 3 seconds
内存使用限制 1 Gigabyte
读入文件 标准输入
输出文件 标准输出

题目描述

Maddy 碰到了朵朵绽放的荷花，但荷花之中又透露了一丝诡异...
在这个荷花池中，有一棵具有 n 个结点的树，这是一个由 $n - 1$ 条边连成的无向无环连通图。
每个结点 i ($1 \leq i \leq n$) 都带有 a_i 的权值，而一棵树的 k - 点对 (x, y) 定义如下：

- $1 \leq x < y \leq n$, x 和 y 是整数，表示树上的结点。
- 记树上的结点 x 到结点 y 之间的路径上的节点上的权值构成一个多重集合 S ，注意 a_x 和 a_y 也算在路径上。
- 此时需要有 $\gcd S = \min S = k$ ，也即 S 中元素的最大公因数等于 S 中元素的最小值且为 k 。

Maddy 现在在树上随机选取两个不同的点 (x, y) ，如果存在 k 满足 (x, y) 是 k - 点对，则她会得到 k 个灯笼。
请帮助 Maddy 求出其期望获得的灯笼数 q 对 998 244 353 取模的值 ans ，具体的输出要求请参照提示。

输入格式

第一行输入一个整数 T ($1 \leq T \leq 10^5$)，表示数据组数。
每组测试数据输入的第一行输入一个整数 n ($2 \leq n \leq 10^5$)，表示图的顶点数量。
随后 $n - 1$ 行每行输入两个整数 u 和 v ($1 \leq u, v \leq n, u \neq v$)，表示树 T 上存在一条 u 到 v 的边。
随后一行输入 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$) 分别表示树上点的权值。
保证所有测试数据输入中的 $\sum n \leq 3 \times 10^5$ ，且每组数据输入的图构成一棵树。

输出格式

对于每组测试数据，输出一行一个整数 ans 表示 Maddy 期望获得的灯笼数对 998 244 353 取模的值。

样例 1

standard input	standard output
1	332748119
6	
1 2	
1 3	
2 4	
2 5	
3 6	
6 2 3 4 2 1	

样例 1 解释

给定的树共有 6 个点，边为

1-2, 1-3, 2-4, 2-5, 3-6,

权值分别为

$a_1 = 6, a_2 = 2, a_3 = 3, a_4 = 4, a_5 = 2, a_6 = 1$.

我们枚举所有 $\binom{6}{2} = 15$ 对点，检查路径上的权值集合 S 的 $\gcd(S)$ 与 $\min(S)$ 是否相等。
例如：

- 点对 $(1, 2)$: 路径为 $[6, 2]$, $\min = 2$, $\gcd = 2$, 因此属于 $k = 2$ 。
- 点对 $(1, 3)$: 路径为 $[6, 3]$, $\min = 3$, $\gcd = 3$, 因此属于 $k = 3$ 。
- 点对 $(1, 6)$: 路径为 $[6, 3, 1]$, $\min = 1$, $\gcd = 1$, 因此属于 $k = 1$ 。
- 点对 $(3, 4)$: 路径为 $[3, 6, 2, 4]$, $\min = 2$, $\gcd = 1$, 不符合条件。

最终统计结果如下:

$$\begin{aligned} k = 1 : 5 \text{ 对 } & (1, 6), (2, 6), (3, 6), (4, 6), (5, 6), \\ k = 2 : 6 \text{ 对 } & (1, 2), (1, 4), (1, 5), (2, 4), (2, 5), (4, 5), \\ k = 3 : 1 \text{ 对 } & (1, 3), \\ k = 4, 5, 6 : 0 \text{ 对.} \end{aligned}$$

可以求得对应的期望灯笼数为

$$\frac{5 \times 1 + 6 \times 2 + 1 \times 3 + 0 \times 4 + 0 \times 5 + 0 \times 6}{15} = \frac{4}{3}.$$

对应取模得到答案 332748119.

提示

可以证明本题的答案会是一个有理数, 记这个答案是 $\frac{p}{q}$ 其中 p, q 互质, 你需要输出的数字 ans 须满足在 $0 \leq ans < 998\,244\,353$ 之内且

$$q \cdot ans = p \pmod{998\,244\,353}.$$

可以证明这样的 ans 在题目意义下一定存在。