

广西科技大学第七届大学生 程序设计大赛试题册

启迪数字学院 ACM 实验室
2025 年 12 月 14 日



本试题册共 15 题，12 页。

如果您的试题册缺少页面，请立即通知志愿者。

试题列表

题号	题目	时限
A	赛事预测	1s
B	线段王国	1s
C	俗手，本手，妙手都不如举手	1s
D	最优路径	1s
E	小方的能量收集器	1s
F	小方和小谢的“左右脑”	1s
G	相信我，我真是签到题	1s
H	送气球	1s
I	小何的二进制镜子	1s
J	💡 点灯问题	1s
K	城市交通	1s
L	检测箱游戏	1s
M	异或	1s
N	来自小方的平方数	1s
O	最大三角形面积	1s

Problem A. 赛事预测

时间限制: 1000ms, 内存限制: 512MB

第 X 届广西科技大学程序设计竞赛正在进行中, 作为现场技术人员, 你需要根据所有队伍的提交记录, 计算并生成最终的排行榜。这不仅关系到各支队伍的排名, 也是赛事公平性的重要体现。小周作为上届金牌得主, 也在现场关注着比赛进程, 他很想知道最终的排名结果。

给定所有参赛队伍的提交记录, 请你按照 ACM 竞赛规则计算并输出最终的排行榜, 同时计算本次比赛金奖、银奖、铜奖的获奖队伍数量。具体规则如下:

1. 排行榜排序规则:

- 首要依据为“解出题目数量”, 降序排列 (解出越多排名越前)。
- 若解出数量相同, 依据“总罚时”升序排列 (罚时越少排名越前)。
- 若解出数量和罚时均相同, 按队名字典序升序排列 (字母序靠前的排名更前)。

ps: 关于字典序:

字典序是一种基于字符编码的排序方法。在字典序中, 字符串的比较是从左到右逐个字符进行的。

首先比较第一个字符, 如果相同则继续比较第二个字符, 以此类推。

如果某个字符串是另一个字符串的前缀 (前缀是指从串首开始到某个位置结束的一个特殊子串。例如: abc 是 abcdef 的前缀), 则较短的字符串排在前面。

例如, 比较 apple 和 banana 时, 我们看到第一个不同的字符是 a 和 b。在字母表中, a 在 b 之前, 因此 apple 在 banana 之前。又例如, 比较 abaaa 和 aba 时, 因为 aba 是 abaaa 的前缀, 所以 aba 在 abaaa 之前。

2. 总罚时计算:

- 对于每道成功解出 (结果为“AC”) 的题目, 罚时 = 该题首次 AC 的时间 + (AC 前的错误提交次数 × 20 分钟)。
- 未解出的题目 (结果非“AC”), 其错误提交不计算罚时。

3. 奖项设置:

- 有效队伍数: 指至少有一次提交记录的队伍总数。
- 金奖数量 = $\lceil \text{有效队伍数} \times 0.10 \rceil$ (向上取整)。
- 银奖数量 = $\lceil \text{有效队伍数} \times 0.15 \rceil$ 。
- 铜奖数量 = $\lceil \text{有效队伍数} \times 0.20 \rceil$ 。

输入格式

输入一个正整数 n , 表示本次比赛后台一共有 n 条按时间顺序的提交记录。

接下来 n 行, 每行提交记录格式为: 队名 题号 结果 时间, 其中

- 队名: 由大小写字母、数字或下划线组成, 不含空格, 长度 $1 \sim 20$ 。
- 题号: 单个大写字母 ($A - M$), 表示题目编号。
- 结果: 字符串, “AC” 表示通过, 其他字符串 (如 WA、TLE、MLE 等) 均表示未通过。
- 时间: 非负整数, 表示提交时间 (单位: 分钟), 范围 $0 \sim 300$ (竞赛时长为 5 小时)。

输出格式

输出分为两部分:

- 排行榜: 每行表示一个队伍的信息, 格式为 排名 队名 解出数量 总罚时。
 - 若当前队伍与前一名的解出数量或罚时不同, 则排名为“前一名排名 + 1”;
 - 若两者均相同 (并列), 则排名与前一名相同, 队名字典序小的在前。
- 奖项数量: 最后一行输出三个整数, 分别表示金银铜奖的队伍数量, 用空格分隔。

样例

标准输入	标准输出
8 Team3 A WA 50 Team3 B WA 60 Team2 B WA 80 Team1 A WA 90 Team0 A AC 100 Team1 A AC 110 Team2 C WA 110 Team2 C AC 120	1 Team0 1 100 2 Team1 1 130 3 Team2 1 140 4 Team3 0 0 1 1 1

说明

1. 提交记录处理:

- Team0: 解出 A (首次 AC 时间 100, 错误提交 0 次), 总罚时 100。
- Team1: 解出 A (首次 AC 时间 110, AC 前错误提交 1 次), 总罚时 $110 + 1 \times 20 = 130$ 。

- Team2: 解出 C (首次 AC 时间 120, AC 前错误提交 1 次 (110 分钟的 WA)), 总罚时 $120 + 1 \times 20 = 140$ (B 题未 AC, 错误提交不计时)。
- Team3: 未解出任何题目, 总罚时 0。
- 有效队伍数: 4 支 (Team0~Team3 均有提交)。

2. 奖项计算:

- 金奖: $\lceil 4 \times 0.10 \rceil = \lceil 0.4 \rceil = 1$ 。
- 银奖: $\lceil 4 \times 0.15 \rceil = \lceil 0.6 \rceil = 1$ 。
- 铜奖: $\lceil 4 \times 0.20 \rceil = \lceil 0.8 \rceil = 1$ 。

评测数据规模

对于所有评测数据, $1 \leq n \leq 10^4$ 。

Problem B. 线段王国

时间限制: **1000ms**, 内存限制: **512MB**

在神秘的大陆“线段王国”中, 国王拥有一批神奇的线段护符, 这些护符能够形成一道强大的防护屏障, 保护王国免受外敌入侵。每条护符线段覆盖着一段区域, 只有所有护符共同覆盖的区域, 才能确保王国的安全无虞。然而, 最近王国遭遇了魔法风暴, 导致部分护符出现了异常。国王召集了最聪明的魔法师——你, 希望能帮助他解决难题:

你可以选择最多删除一条护符线段(也可以选择不删除任何一条), 以最大化所有剩余护符线段同时覆盖的区域长度。换句话说, 你需要找出删除零条或一条线段后, 所有剩余线段交集的最大长度 (包括端点)。

输入格式

第一行输入一个整数 n , 代表护符线段的数量。

接下来 n 行, 每行两个整数 l 和 r , 用空格分隔, 分别表示每条护符线段的左端点和右端点。(所有端点均为整数)

输出格式

输出一个整数, 代表删除最多一条线段后, 所有剩余线段共同覆盖区域的最大长度。

样例

标准输入	标准输出
1 1 5	5
2 1 5 5 10	6
3 1 5 6 10 11 15	0

说明

- 你可以选择不删除任何线段。

评测数据规模

对于所有测评数据, $1 \leq n \leq 10^5, 1 \leq l \leq r \leq 10^9$ 。

Problem C. 俗手，本手，妙手都不如举手

时间限制：1000ms， 内存限制：512MB

“本手、妙手、俗手，举手”是围棋的四个术语。本手是指合乎棋理的正规下法；妙手是指出人意料的精妙下法；俗手是指貌似合理，而从全局看通常会受损的下法。对于初学者而言，应该从本手开始，本手的功夫扎实了，棋力才会提高。一些初学者热衷于追求妙手，而忽视更为常用的本手。本手是基础，妙手是创造。一般来说，对本手理解深刻，才可能出现妙手；否则，难免下出俗手，水平也不易提升。当然，如果你在主场作战且处于劣势的情况下，可以选择举手来直接取得胜利。



有两名选手 H 和 C ，他们轮流下棋，每位棋手下有 n 步。棋局可以抽象成一个长度为 $2 \times n$ 的操作序列 p_{ij} ($1 \leq i \leq 2, 1 \leq j \leq n$)。即 $p_{1,j}$ 表示为 H 选手第 j 步的质量， $p_{2,j}$ 表示为 C 选手第 j 步的质量。 每一步的质量由一个正整数表示:

- $p_{ij} = 1$ 表示该步是妙手。
- $p_{ij} = 0$ 表示该步是本手。
- $p_{ij} = -1$ 表示该步是俗手。

注意，此外每一步有可能会发生一事件：对手是否把提子放回棋盖。如果对手没有把提子放回棋盖，可以选择是否举手举报赢棋。

这里 H 国棋手都会遵循一个“举手”策略:

- 一旦对手有犯规，他可以选择立即举报并直接赢棋;
- 但举报行为需要花费时间;
- 如果自己当前处于明显劣势，他才会选择举报。

劣势定义为: 自己下过的妙手数量减去俗手数量严格小于对方的该差值。因为 H 国棋手害怕自己经常下出俗手，所以他会选择在自己落子之前进行举报。

你的任务:

给定整局棋的操作序列 p_{ij} 和 r_i (表示是否犯规, $r_i = 1$ 表示犯规, $r_i = 0$ 表示不犯规), 以及举报花费时间 t 。

假设棋局总时长为 T ，如果举报发生在时刻 x 且 $x + t > T$ ，则比赛直接超时，举报失败。

请判断 H 国棋手是否能够通过举报赢下比赛。

如果可以，输出最早的举报时刻 (即第几步之后举报);

否则输出 “Might as well raise your hand”。

输入格式

第一行三个整数 n, t, T ，分别表示棋手下有的步数、举报所需要时间、棋局的总时长。

接下来的 $2 \times n$ 行:

- 第 $2 \times k - 1$ 行有两个整数 p_{1n} 和 r_n ，分别代表该步数的质量和是否可以举报。
- 第 $2 \times k$ 行有一个整数 p_{2n} ，代表该步数的质量。
- ($k = 1, 2, 3, ..., n$)

输出格式

如果存在合法举报 (满足劣势条件且举报后未超时)，输出举报的步数 (从 1 开始计步)。

否则输出: Might as well raise your hand

样例

标准输入	标准输出
3 2 10 1 0 1 -1 1 0 -1 1 0	5

说明

第 5 步对手犯规且 H 国棋手此时已处于劣势，举报需要 2 单位时间，总时间不超过 10，因此在第 5 步举报即可获胜。

评测数据规模

对于所有评测数据， $1 \leq n \leq 2 \times 10^5$ 。

Problem D. 最优路径

时间限制: 1000ms, 内存限制: 512MB

小何是一个优秀的赛车手！小何的赛车训练场地由若干个检查点（节点）和连接它们的赛道（边）组成。每条赛道都有一个固定的速度限制。有一天 xsy 请求小何同志，让小何同志带他体验一下赛车的乐趣。作为贴心的小何同志，自然想让 xsy 的乘坐过程更加舒适，我们规定乘坐过程中最高速度与最低速度的差值越小，舒适度越高（赛车可瞬间提速或降速，无需考虑加速过程）。小何对赛车时间没有要求，你能帮助小何同志找出两城市间最舒适的航线吗？注：作为最优秀的赛车手，小何在赛道上行驶时，自然是按照赛道允许的最高速度来啦

输入格式

第一行包含两个整数 n 和 m ，表示检查点赛道的数量。检查点编号从 1 到 n 。

接下来 m 行，每行包含三个整数 u, v, w ，表示检查点 u 和 v 之间有一条双向赛道，速度限制为 w 。

其中， $u(1 \leq u \leq n), v(1 \leq v \leq n), w(1 \leq w \leq 10^6)$

接下来一行包含一个整数 q ，表示查询次数。

接下来 q 行，每行包含两个整数 s, t ，表示起点为 s ，终点为 t 。

输出格式

对于每个查询，输出一个整数，表示找到的最小差值。如果无法从终点到达起点，输出 -1 。

样例

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

评测数据规模

对于所有评测数据， $1 \leq n \leq 200, 1 \leq m \leq 10^3, 1 \leq w \leq 10^6, 1 \leq q \leq 11$ 。

Problem E. 小方的能量收集器

时间限制: 1000ms, 内存限制: 512MB

某能量收集器上分布着 n 个连续的能量节点，每个节点都有对应的能量值(能量值可正可负)。现在需要从这些节点中收集能量，最多可以连续收集 k ($1 \leq k \leq m$)个节点的能量，其中($m \leq n$)。请你找出连续收集的能量总和最大的情况，输出这个最大总和。形式化地，在数列 $\{p_n\}$ 中，找出一个子段 $[l, r](r - l + 1 \leq m)$ ，最大化 $\sum_{i=l}^r p_i$ 。

输入格式

第一行两个整数 n, m 。分别代表共有 n 个能量节点，最多只能连续收集 m 个节点。第二行 n 个整数，第 i 个整数 p_i 代表第 i 个能量节点的能量值。

输出格式

仅一行一个整数，即能收集到的最大能量总和。

样例

标准输入	标准输出
5 2 1 2 3 4 5	9
6 3 1 -2 3 -4 5 -6	5

评测数据规模

对于所有评测数据， $1 \leq n \leq 5 \times 10^5, |p_i| \leq 500$ 。

保证答案的绝对值在 $[0, 2^{31} - 1]$ 之内。

Problem F. 小方和小谢的“左右脑”

时间限制: 1000ms, 内存限制: 512MB

小方和小谢在 ACM 比赛中遇见 “超简单” 题, 总喜欢通过博弈来决定这道 “超简单” 题由谁来 AC。今天小方和小谢在某场 VP 中遇见某道 “超简单” 题, 接下来, 小方和小谢将开始激烈的博弈 ...

博弈内容如下:

首先这里有 n 个石子, 小方和小谢每次每个人可以从中拿走 x 个石子 ($1 \leq x \leq 3$), 小方先手, 小方和小谢都想赢并且想尽可能拿到的石子最多, 最后一次拿走石子的人获胜。假设小方和小谢都遵循最优的策略, 请决定出最后的获胜者以及获胜者所获得的石子 v 。

输入格式

第一行包含一个整数 T , 表示测试用例的数量。

每个测试用例在一行中包含一个整数 n , 表示石子的数量。

输出格式

对于每个测试用例, 如果小方获胜, 则输出 0 v , 否则输出 1 v 。(v 表示获胜者所拿到石子的数量)。

样例

标准输入	标准输出
4	0 1
1	0 2
2	0 3
3	1 1
4	

评测数据规模

对于所有的测评数据: $1 \leq T \leq 10^4$, $1 \leq n \leq 10^{12}$ 。

Problem G. 相信我, 我真是签到题

时间限制: 1000ms, 内存限制: 512MB

小周受邀来参加广西科技大学程序设计竞赛, 但是她不知道在哪签到, 于是她去问了 sumimi 给了她如下的线路:

- 从柳州站出发, 乘坐一条 “柳江绿” 的线路, 在两站之后换乘, 然后再坐 $(5 + 3)$ 站;
 - 换乘一条往 “柳北” 方向的公交, 乘坐 $(6 - 2)$ 站; 换乘, 乘坐 3×2 站;
 - 换乘一条 rgb 值为 $(200, 150, 100)$ 的线路 (模拟柳州工业风配色), 经过 $\frac{120}{30}$ 站后抵达一个换乘站;
 - 换乘一条往 “东环” 方向的线路, 在下一个换乘站换乘后再往西北方 2^2 站;
 - 继续乘坐这条线路, 在第二个换乘站换乘, 然后再在第二个换乘站换乘, 直到终点站的前一站, 该站点与终点站第一个字都表示 “螺”;
 - 换乘一条往北的线路, 经过 $|5 - \frac{7+2}{3}|$ 站;
 - 继续乘坐这条线路, 能换乘就换乘, 换乘两次后, 停止换乘, 坐到终点站后继续换乘, 计算矩阵 $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{pmatrix}$ 的 rank 站后再次抵达终点站;
 - 沿着 “元宝山” 的线路 (柳州知名景点), 在一个第一个字在百家姓前八个 (赵钱孙李周吴郑王) 里的公交站换乘, 乘坐 $\sqrt{169}$ 站, 换乘, 再乘坐 $4!$ 站;
 - 继续乘坐 $\int_1^\infty \frac{1}{x \ln^2 x} dx$ 站, 抵达一个四字且最后一字是方向的站点;
 - 乘坐目前所在站点名字最后一个字方向的线路, 乘坐 $\lim_{x \rightarrow 0} \frac{\sin 2x}{x}$ 站, 该站点的前两个字可以用来形容一个柳州特色动漫 IP (以螺蛳粉为原型, 活跃在本地文创中);
 - 继续乘坐该线路, 经过若干个换乘站后, 在一个前两个字是一种动物的站点;
 - 在该站点换乘, 然后乘坐 $\lfloor e \rfloor$ 站抵达一个没有来过的站点;
 - 换乘一条终点站的前两个字是儿歌角色的线路, 乘坐 C_{12}^2 站;
 - 换乘一条颜色为 0xFFD700 的线路, 乘坐 $\log_4 16$ 站, 抵达一个后两字是房屋组成部分的站点;
 - 继续乘坐 $e^{i\pi}$ 站 (取绝对值), 来到一个后两字在麻将役种中同时出现的站点 (如 “平和” 役);
 - 继续乘坐 $5(\sin^2 x + \cos^2 x)$ 站, 抵达 “科大西站”, 该高校就是本次比赛的签到点了。
- 小周看到以上这么长的路线都晕了, 你能告诉小周目的地的高校是哪个吗?

输入格式

没有输入。

输出格式

输出五个中文字的拼音表示目的地所在高校的名称的拼音（五个字，拼音全部是小写，每个字的拼音之间间隔着一个空格）。例如，原神启动（四个字），即输出“yuan shen qi dong”（不包含双引号）。

Problem H. 送气球

时间限制：1000ms， 内存限制：512MB

广西科技大学程序设计大赛来啦！！

由于比赛过题的人比较多，需要送出很多气球，所以小何需要从基地(0,0)出发，将气球送到坐标系中的各个位置（保证气球的坐标都是整数）。

小何每次送一个气球都从基地出发(0,0)，由于小何是个严谨的人，他每次走的长度只能为整数，且小何每次只能携带一个气球，将气球送到目标点后小何会返回基地，再去送下一个气球。从一个点移动到另一个点(这两个点之间的直线距离必须是正整数)花费的代价为 1。

例如，小何想送气球到(15,9)处，那么他的路线可以为 (0,0) → (12,5) → (15,9) → (12,5) → (0,0)，走了4步，且每次走的距离均为整数满足条件，花费的代价为4。

小何想要花费最少的代价来把所有气球送到指定位置，请你求出最小代价。

输入格式

第一行一个整数 n ，表示需要送的气球的个数。

接下来 n 行，每行两个整数 x 和 y ，表示气球的坐标。

输出格式

一个整数，表示最小代价。

样例

标准输入	标准输出
1 15 9	4

评测数据规模

对于所有评测数据， $1 \leq n \leq 10^3$ ， $0 \leq x, y \leq 200$ 。

Problem I. 小何的二进制镜子

时间限制: 1000ms, 内存限制: 512MB

小何有一面神奇的镜子，它只能照出二进制数字。如果一个数字在镜子中看到的和自己一模一样（即二进制表示是回文的），那么小明就会认为这个数字是“镜子友好”的。

现在，请你编写一个程序，帮助小明判断一个数字是否是“镜子友好”的。

输入格式

第一行包含一个整数 t ，表示测试用例的数量。

接下来 t 行，每行包含一个整数 n 。

输出格式

对于每个测试用例，如果 n 是“镜子友好”的，输出“Yes”；否则输出“No”。

每个测试用例的输出占一行。

样例

标准输入	标准输出
3	Yes
5	No
4	Yes
1	

说明

5的二进制是101，正反读都一样，所以是镜子友好的

4的二进制是100，反过来是001，不一样，所以不是镜子友好的

1的二进制是1，正反读都一样，所以是镜子友好的

评测数据规模

对于所有评测数据， $1 \leq t \leq 10^3$ ， $0 \leq n \leq 10^{18}$ 。

Problem J. 点灯问题

时间限制: 1000ms, 内存限制: 512MB

小何的新公寓安装了一块智能灯阵，灯阵是 $(N \times N)$ 的方格，每盏灯都能通过遥控器控制。然而这支遥控器似乎出了点毛病：每次按下一个按键时，不仅那一盏灯会亮起，它上下左右相邻的灯也会一同被翻转（亮的变灭，灭的变亮）。

现在，小何发现所有的灯都处于关闭状态（全黑）。他想知道：是否存在一种点击方式，可以让所有灯最终都亮起来？如果可以，请输出任意一种可行的点击方案。

游戏规则：

- 灯阵大小为 $(N \times N)$ 。
- 初始状态：所有灯均为黑（灭）。
- 每次点击一个格子，会翻转该格子及其上下左右相邻格子的亮灭状态。
- 目标：使所有灯最终都变为亮（白）。
- 点击顺序不影响结果（因为翻转是可交换的）。

输入格式

输入一行一个整数 N 。

输出格式

- 若无论如何都无法让所有灯亮起，输出一行：NO
- 否则输出：YES

接着输出 N 行，每行 N 个整数 0 或 1（空格分隔），表示点击方案矩阵：

- 1 表示该格子需要点击一次；
- 0 表示该格子不需要点击。

样例

标准输入	标准输出
3	YES 0 1 0 1 0 1 0 1 0

评测数据规模

对于所有评测数据， $1 \leq N \leq 300$ 。

Problem K. 城市交通

时间限制: 1000ms, 内存限制: 512MB

X 城市的交通十分不便利, 城市可被看作一个 $n \times m$ 大小的矩阵。FXZ 手持该城市的地图, 地图以特定格式描述了城市内的通行规则, 他需要尽快从当前位置赶到机场, 因此需要找到从起点到机场的最短路径。

- X 城市的地图是一个 $2 \times n + 1$ 行、 $2 \times m + 1$ 列的矩阵:
- 地图中 FXZ 所在的格子用 S 表示, 机场所在的格子用 T 表示, 其余可通行格子用空格表示;
 - 左右相邻的两个格子若无法通行, 用|表示; 上下相邻的两个格子若无法通行, 用-表示; +表示格子的四个角;
 - 题目保证城市 X 最外圈无法通行。

请你帮助 FXZ 计算到达机场最少需要走过的格子数 (包括起点 S 和终点 T)。若无法到达机场 T, 则输出“FXZ Got lost...TAT”。

输入格式

第一行输入两个整数 n, m , 表示 X 城市的大小 (n 行 m 列的格子)。接下来 $2 \times n + 1$ 行, 每行包含 $2 \times m + 1$ 个字符, 描述 FXZ 手中的地图。题目保证 S 和 T 都有且仅有一个。

输出格式

若 FXZ 能到达机场, 输出最少需要经过的格子数; 否则输出“FXZ Got lost...TAT” (不包含双引号)。

样例

标准输入	标准输出
4 3 +--+--+ S + +--+ + +--+ T + +-+ + +--+--+	8
3 3 +--+--+ S + + +--+ T + + +--+ +--+--+	FXZ Got lost...TAT

说明

对于第一个样例, 链接: TRDD 所在的位置为(1, 1), 机场的位置为(3, 2)路线为(1, 1) -> (2, 1) -> (3, 1) -> (4, 1) -> (4,2) -> (4,3) -> (3,3) ->(3,2) 共 8 个格子

对于第二个样例, 无法从 S 到达 T。

评测数据规模

对于所有评测数据, $1 \leq n, m \leq 3000$ 。

Problem L. 检测箱游戏

时间限制: 1000ms, 内存限制: 512MB

朱哥带来了 n 个外形相同的密封箱子, 其中恰有 1 个箱子含有有毒气体, 其余 $n - 1$ 个箱子安全。小方与小谢决定玩一个检测游戏来决定奖品归属。

两人轮流行动, 小方先手。在每一回合中, 当前玩家必须从场上尚未被移除的箱子中选择 1 个 进行检测。检测仪器的准确率为 p , 满足 $(0.5 < p < 1)$ 。每次检测结果独立生成, 且公开给双方。

检测规则如下 (单次检测):

- 若被测箱子真实为安全: 以概率 p 返回 “SAFE”, 以概率 $1 - p$ 返回 “TOXIC”;
- 若被测箱子真实为有毒: 以概率 p 返回 “TOXIC”, 以概率 $1 - p$ 返回 “SAFE”。

检测结果揭晓后, 按下述规则执行:

1. 若结果为 “TOXIC”: 该箱子立即移除(不揭示其真实属性), 回合结束, 轮到对方行动;
2. 若结果为 “SAFE”: 当前玩家必须立刻二选一:

- 打开该箱子: 若该箱真实安全, 则当前玩家立即获胜; 若真实有毒, 则当前玩家立即失败 (对方获胜);
- 不打开: 该箱子立即移除 (不揭示其真实属性), 回合结束, 轮到对方行动。

游戏在以下任意情况发生时立即结束。

- 打开箱子将立刻揭示真实状态并结束游戏: 开安全者胜; 开有毒者负。
- 当场上只剩 1 个箱子时: 轮到行动的玩家必须直接打开该箱子 (不再进行检测); 若安全则该玩家获胜, 若有毒则该玩家失败。

朱哥给了小方一次 “二次检测权”, 可在她的任意一次回合中使用一次 (也可选择不用)。当小方在自己的回合选择某个箱子并声明使用特权时:

- 对该箱子进行两次独立检测 (两次均满足上述准确率 p , 结果独立且公开);
- 若两次检测中至少有一次为 “TOXIC”: 则该箱子立即移除, 回合结束, 轮到对方行动;
- 若两次检测结果均为 “SAFE”: 小方可立即选择打开该箱或不打开:
 - 打开: 若真实安全则小方立即获胜; 若真实有毒则小方立即失败;
 - 不打开: 该箱子立即移除, 回合结束, 轮到对方行动。

该特权只能使用一次。

假设两名玩家都采取最优策略 (均尽力最大化自身胜率), 请计算: 小方在游戏开始时的最大获胜概率。

输入格式

- 第一行包含一个整数 T , 表示测试用例的数量。
- 接下来 T 行, 每行包含两个数 $n_i \ p_i$ 。
- n_i 为整数, 代表箱子数量;
- p_i 为实数, 代表检测准确率, 满足 $(0.5 < p_i < 1)$ 。

输出格式

对每组数据, 输出一个实数, 表示小方在最优策略下的最大获胜概率。当你的答案与标准答案的绝对或相对误差不超过 10^{-12} 时, 视为正确。

样例

标准输入	标准输出
2 3 0.6 5 0.7	0.400000000000 0.659433333333

评测数据规模

对于所有评测数据, $1 \leq T \leq 10^5$, $2 \leq n \leq 10^6$, $0.5 < p < 1$ 。

所有测试的总规模不超过 10^6 。

Problem M. 异或

时间限制: 1000ms, 内存限制: 512MB

小方最近新学了一些有趣的数学知识——异或，于是就来考考他的朋友小谢，你能帮小谢解答一下吗？

- 十进制 5 的二进制为 101;
- 十进制 0 的二进制为 0;
- 所以 $5 \oplus 0 = 5$ ，异或运算过程如下:



现有两个整数 a 和 b ，你可以进行任意次数（包括 0 次）的如下操作：挑选一个整数 x ，要求 $(0 \leq x \leq \text{当前的 } a \text{ 值})$ （注意这里的 a 是操作时的当前值，不是最初给定的初始值）；随后将 a 更新为 a 和 x 的按位异或结果（即 $a := a \oplus x$ ），其中 (\oplus) 表示按位异或操作）。你的目标是通过一系列这样的操作，让 a 最终恰好等于 b 。请找出一个操作方案，使得操作次数不超过 100 次；如果不存在这样的方案，就输出 -1 。

输入格式

- 首先输入一个整数 t ，表示测试用例的数量。
- 每个测试用例包含两个整数 a 和 b 。

输出格式

对于每个测试用例:

如果无法通过允许的操作将 a 变为 b ，输出 -1 ；否则，第一行先输出一个整数 k （ $0 \leq k \leq 100$ ）表示操作次数，然后接下来第二行再输出 k 个整数 x_1, x_2, \dots, x_k ，表示每次操作选择的 x 的值（顺序为操作执行顺序）。若有多个可行方案，输出任意一个即可。

样例

标准输入	标准输出
6	2
9 6	7 8
13 13	0
292 929	-1
405 400	1
998 244	5
244 353	2
	25 779
	-1

评测数据规模

对于所有评测数据, $1 \leq t \leq 10^3$, $1 \leq a, b \leq 10^9$ 。

Problem N. 来自小方的平方数

时间限制: 1000ms, 内存限制: 512MB

给定任意正整数 n , 请你在不超过 100 次操作内, 将其转化为一个完全平方数。每次操作需遵循以下规则:

- 选择当前数 n 的一个正约数 x (即 n 能被 x 整除, 满足 $(n \bmod x = 0)$);
- 每次选择的 x 必须与之前所有操作选的 x 都不同; 选择后, 将 n 加上 x 。

注意: 整个操作过程中, n 的值不超过 (10^{18}) 。

输入格式

本题包含多组测试用例。第一行输入一个整数 T , 表示测试用例的组数。接下来 T 行, 每行输入一个正整数 n 。

输出格式

对于每组测试用例: 第一行输出一个整数 m ($0 \leq m \leq 100$), 表示操作次数。第二行输出 m 个用空格分隔的不同正整数, 表示每次操作选择的 x 。

样例

标准输入	标准输出
2 2025 182	0 3 7 3 4

评测数据规模

对于所有评测数据, $1 \leq T \leq 10^3$, $1 \leq n \leq 10^{12}$ 。

Problem O. 最大三角形面积

时间限制: 1000ms, 内存限制: 512MB

在二维平面中, 给定三角形的三个顶点:

点 A 位于 x 轴, 坐标为 $A(x_0, 0)$ 点 B 位于 x 轴, 坐标为 $B(x_1, 0)$ 点 C 位于 y 轴, 坐标为 $C(0, y_0)$, 这三个点构成了一个三角形 ABC。

现在需要画一条 平行于 x 轴 的直线 k (即水平线), 它与边 AC 相交于点 a, 与边 BC 相交于点 b。在边 AB 上也存在一个点 c。由这三个点 a, b, c 可以构成一个新的三角形 abc。

直线 k 的具体位置未知, 点 c 的位置也未知。请你求出三角形 abc 的 最大可能面积。

输入格式

第一行输入一个整数 T, 表示测试数据组数。

接下来 T 行, 每行包含三个整数: $x_0x_1y_0$

输入保证 x_0, x_1, y_0 均为整数。

输出格式

对于每组数据, 输出一个实数, 表示最大三角形面积。答案与标准答案的误差需不超过 10^{-6} 。

样例

标准输入	标准输出
2 -3 -2 -5 -7 11 -10 -7 24 0	0.6250000000 22.5000000000 4.5000000000

评测数据规模

对于所有评测数据, $1 \leq T \leq 10^5$, $-10^7 \leq x_0, x_1, y_0 \leq 10^7$ 。