

A . 大富翁 / A

时间限制： 1.0 秒

【题目背景】

有一天，小 W 和小 H 在玩大富翁。

【题目描述】

这版大富翁的游戏规则比较独特。它的地图是一棵 n 个节点的有根树，其中 1 号节点为根。树上每个节点都有一个价格，第 x 号节点的价格记为 w_x 。

对于树上两个不同的节点 x, y ，若 x 是 y 的祖先节点（即， x 在 1 号点到 y 号点的简单路径上），则称 x 支配 y 。

游戏过程中，小 W 和小 H 轮流购买树上的一个未被人购买过的节点，直到树上的 n 个节点都被小 W 或小 H 购买。（游戏开始前，树上的所有节点都没有被购买。）

对于一次购买，假设买方购买了 x 号节点，那么他首先要向系统支付 w_x 个游戏币。假设此时 x 支配着 n_1 个已被买方的对手购买了的节点，同时又被 n_2 个已被对手购买了的节点支配。若 $n_1 > n_2$ ，那么对手要向买方支付 $n_1 - n_2$ 个游戏币，若 $n_1 < n_2$ ，那么买方要向对手支付 $n_2 - n_1$ 个游戏币。

小 W 和小 H 都是绝顶聪明的人，他们都会在游戏中采用最优策略，来使自己赚到尽量多的游戏币。现在，小 W 想考考你：如果他先手，他最终能赚到多少个游戏币？（即，在整个游戏过程中，小 W 从小 H 手中获得的游戏币个数减去他支付给系统和小 H 的游戏币个数。你可以认为，游戏开始前，小 H 和小 W 手中都有足够数量的游戏币。注意：答案可能为负数。）

【输入格式】

从标准输入读入数据。

第一行一个正整数 n 。

第二行 n 个非负整数，第 i 个整数为 i 号节点的价格 w_i 。

第三行 $n - 1$ 个正整数，第 i 个正整数表示第 $i + 1$ 号节点的父亲编号。

【输出格式】

输出到标准输出。

一行一个整数表示答案。

【样例 1 输入】

```
1 7
2 0 0 1 0 0 0 0
3 1 1 2 2 3 3
```

【样例 1 输出】

```
1 2
```

【样例 1 解释】

一个可能的游戏过程是：

- 第一次购买：小 W 购买 1 号节点，向系统支付 0 个游戏币。
- 第二次购买：小 H 购买 2 号节点，向系统支付 0 个游戏币，并向小 W 支付 1 个游戏币。
- 第三次购买：小 W 购买 3 号节点，向系统支付 1 个游戏币。
- 第四次购买：小 H 购买 4 号节点，向系统支付 0 个游戏币，并向小 W 支付 1 个游戏币。
- 第五次购买：小 W 购买 6 号节点，向系统支付 0 个游戏币。
- 第六次购买：小 H 购买 5 号节点，向系统支付 0 个游戏币，并向小 W 支付 1 个游戏币。
- 第七次购买：小 W 购买 7 号节点，向系统支付 0 个游戏币。

【样例 2】

见题目目录下的 *2.in* 与 *2.ans*。

【子任务】

对于所有测试数据， $1 \leq n \leq 2 \times 10^5$ ， $0 \leq w_x \leq 2 \times 10^5$ 。保证输入的图为一棵以 1 号节点为根的有根树。

B. 拧螺丝 / B

时间限制：2.0 秒

【题目背景】

张华考上了北京大学；李萍进了中等技术学校；小 E 在工地搬砖：他们都有光明的前途。

由于搬砖太累，小 E 决定进厂拧螺丝。但是小 E 遇到了一个各种刁难人的老板。

【题目描述】

加工一个模块需要拧 n 个螺丝。但是，每隔一段固定的时间，小 E 的老板就会到小 E 的工位上收走一个未完工的模块。在这期间，只够小 E 拧 k 个螺丝。

小 E 的老板刚刚离开。问老板接下来第几次来的时候小 E 才可能有一个完工的模块？假设小 E 的老板极力不想让小 E 达成这件事。

【输入格式】

从标准输入读入数据。

一行两个正整数， n, k 。

【输出格式】

输出到标准输出。

如果小 E 无论如何也不能有一个完工的模块，那么输出一行 `Poor E.S.!`。

否则输出一行一个正整数，表示答案。

【样例输入】

```
1 3 2
```

【样例输出】

```
1 2
```

【样例解释】

小 E 先选择两个模块各拧上一个螺丝，然后老板来了拿走一个。接下来小 E 只需要在剩下的那个零件上面拧两个螺丝就可以完工了。所以老板第二次来的时候小 E 就

有了一个完工的模块。注意小 E 在老板来的时候不能有只拧上了一半的螺丝，不然老板看到了会说小 E 加工不合格。

【子任务】

保证 $1 \leq k \leq n \leq 10^5$ 。

C . 快速 LCM 变换 / C

时间限制： 2.0 秒

【题目描述】

小 I 今天学习了快速最小公倍数变换 (Fast Least-Common-Multiple Transform, FLT), 于是他想考考你。

给定一个长度为 n 的正整数序列 r_1, r_2, \dots, r_n 。你需要做以下操作恰好一次:

- 选择整数 i, j 使得 $1 \leq i < j \leq n$ 。在序列末尾加入 $(r_i + r_j)$, 并将 r_i 和 r_j 从序列中删除。

可以注意到总共有 $\frac{n(n-1)}{2}$ 种可能的操作, 每种操作会得到一个长度为 $n-1$ 的序列。你需要对所有的这 $\frac{n(n-1)}{2}$ 个序列, 求出序列中所有元素的最小公倍数, 并给出它们的和模 998244353 的值。

【输入格式】

从标准输入读入数据。

输入的第一行包含一个正整数 n , 表示序列的长度。接下来一行 n 个正整数 r_1, r_2, \dots, r_n , 描述初始序列。

【输出格式】

输出到标准输出。

输出一行一个整数, 表示所有序列的最小公倍数的和模 998244353 的值。

【样例 1 输入】

```
1 3
2 2 3 4
```

【样例 1 输出】

```
1 40
```

【样例 1 解释】

- $i = 1, j = 2$ 时, 得到的序列为 $\{4, 5\}$, 最小公倍数为 20;
- $i = 1, j = 3$ 时, 得到的序列为 $\{3, 6\}$, 最小公倍数为 6;
- $i = 2, j = 3$ 时, 得到的序列为 $\{2, 7\}$, 最小公倍数为 14。

因此输出为 $20 + 6 + 14 = 40$ 。

【子任务】

对于所有测试数据， $2 \leq n \leq 5 \times 10^5$, $1 \leq r_1, r_2, \dots, r_n \leq 10^6$ 。

D . 种苹果 / D

时间限制：6.0 秒

【题目描述】

农夫种了一棵苹果树，树上结满了大大小小的苹果。夏天正是果树生长发育的大好时节，树上不断抽出新的枝条、结出新的苹果，已有的苹果也在不断长大。

为了观察和记录苹果的生长状况，以便对未来收获的行情有大致的估计，农夫进行了长时间仔细的观察和研究。在整个记录周期的最开始，树上一共结有 n 个苹果，农夫将其编号为 $1 \sim n$ ，有 $n - 1$ 条树枝连接这些苹果，每条树枝的两端都恰好挂有一个苹果，使得整个苹果树成为一个名副其实的树形结构。农夫对每个苹果进行了一番价值估计，第 i 个苹果的初始价值为 a_i ，表示农夫此时摘下它并卖出的净收益，考虑到成本因素， a_i 可能为负。

在整个记录周期中，共发生了 m 件值得记录的事件，所有的事件共分为以下几种类型：

1 $u v w$ ：树上原本连接苹果 u 和苹果 v 的树枝中间结出了一个新的苹果。设原先树上共有 k 个苹果，则此时变为 $k + 1$ 个，农夫将新长出的苹果编号为 $k + 1$ ，其价值为 w 。原先连接苹果 u 和 v 的树枝也因此分裂成两条，一条连接苹果 u 和 $k + 1$ ，另一条连接苹果 $k + 1$ 和 v ；

2 $u w$ ：树上长出了一条新树枝和一个新苹果。设原先树上共有 k 个苹果，则此时变为 $k + 1$ 个，农夫将新长出的苹果编号为 $k + 1$ ，其价值为 w 。新树枝连接苹果 u 和 $k + 1$ 。

3 $u v w$ ：树上一部分苹果的价值发生了变化。树上连接苹果 u 和 v 的一整段枝条（即树形结构上连接 u 和 v 的最短路径，包括 u 和 v 本身）上的所有苹果的价值均增加了 w 。考虑到价值的变化也可能是由于营养不足或病虫害引起的，因此 w 可能为负。

4 $u v w$ ：农夫想在树上进行一次抽样调查来研究自己的可能收益。他定义价值不小于 w 的苹果为“优质苹果”，并选择了树上连接苹果 u 和 v 的一整段枝条（含义同上），想统计一下这段枝条上的苹果中有多少个“优质苹果”。

但由于苹果的数量是在太多了，农夫数不过来，便只好请你来帮忙。注意：由于农夫不能预测未来，因此你帮农夫时必须**强制在线**地回答问题。

【输入格式】

从标准输入读入数据。

第 1 行：2 个正整数 n, m 。

第 2 行： n 个整数 a_i 。

接下来 $n - 1$ 行：每行两个正整数 u_i, v_i ，依次描述初始时树上的所有树枝。

接下来 m 行，每行 3 或 4 个整数，按照时间顺序描述所有的事件，格式如题目描述中所述。

为了体现强制在线性，设上一次 4 事件的答案是 $lastans$ （最开始时 $lastans = 0$ ），则所有事件中输入的 u, v, w 都需要异或上 $lastans$ 才是真正的 u, v, w 。

【输出格式】

输出到标准输出。

对于每个 4 事件输出一行，一个非负整数，表示农夫此次调查的“优质苹果”数量。

【样例 1 输入】

```
1 5 6
2 1 3 3 2 2
3 1 2
4 1 3
5 2 4
6 2 5
7 4 3 4 2
8 3 2 6 2
9 4 0 7 1
10 1 5 6 1
11 2 2 7
12 4 0 3 0
```

【样例 1 输出】

```
1 3
2 4
3 2
```

【样例 1 解释】

对于这组样例，去除强制在线后的数据如下：

```
1 5 6
2 1 3 3 2 2
3 1 2
4 1 3
5 2 4
6 2 5
```



```
7 4 3 4 2
8 3 1 5 1
9 4 3 4 2
10 1 1 2 5
11 2 6 3
12 4 4 7 4
```

【数据范围】

对于所有数据， $n, m \leq 2 \times 10^5$ ， $|a_i|, |w| \leq 10^9$ ，保证任意时刻涉及到的苹果编号均有意义，保证初始的 $n - 1$ 条树枝构成树形结构，所有 1 事件保证连接苹果 u 和 v 的树枝在事件发生时存在。

E . 速战速决 / E

时间限制： 1.0 秒

【题目描述】

小 I 与小 J 正在玩一个叫做“开火车”，又称作“拖板车”和“小猫钓鱼”的扑克游戏。游戏规则如下，注意其与一般玩法可能有不同：

- 有 $2n$ 张牌，其中对于整数 $1 \leq i \leq n$ ，牌面为 i 的牌恰好有 2 张。
- 游戏开始时，小 I 和小 J 各拿其中 n 张牌组成双方的初始手牌。
- 维护一个公共牌堆（可以将其看作一个栈），初始没有牌。小 I 与小 J 依次行动，小 I 先手。一次行动时，行动方依次进行以下操作：
 1. 将手牌中的一张牌放在公共牌堆顶；
 2. 若此时公共牌堆中有两张相同的牌，则这两张相同的牌以及在这两张牌之间的所有牌从公共牌堆移到当前行动方手牌中；
 3. 若此时当前行动方没有手牌，则当前行动方失败，另一方胜利。

小 J 是扑克萌新，所以会按照以下策略行动：

- 维护一个队列，初始将 n 张手牌按照一定顺序放入队列中；
- 每次行动时，将队列开头的牌放在公共牌堆顶；
- 若小 J 放入某张牌后公共牌堆中有两张相同的牌，则按照在公共牌堆中自顶到底的顺序将获得的牌放入队列尾。

小 I 通过偷看得到了小 J 的策略以及队列中牌的顺序。现在小 I 不仅想获胜，还想速战速决，用最少的行动次数获胜，但他也是扑克萌新。所以给定小 J 队列中的 n 张牌以及它们的顺序，你需要给出小 I 的策略，使得小 I 能够获胜，同时行动次数最少，或者告诉他这是不可能的。

【输入格式】

从标准输入读入数据。

每组数据的第一行一个整数 n 表示牌面的种数。

第二行 n 个整数 a_1, a_2, \dots, a_n ，从队头到队尾的顺序依次描述小 J 队列中的牌。得到小 J 的 n 张手牌也就得到了小 I 的 n 张手牌，因此不会输入小 I 的手牌。

【输出格式】

输出到标准输出。

如果小 I 不可能获胜，只需要输出一个整数 **-1**；否则第一行输出一个整数 s ，表示你给出的策略中小 I 的行动次数。接下来一行 s 个整数，依次描述每次行动时小 I 从手牌中放入公共牌堆的牌的牌面，两个数之间以一个空格分隔。

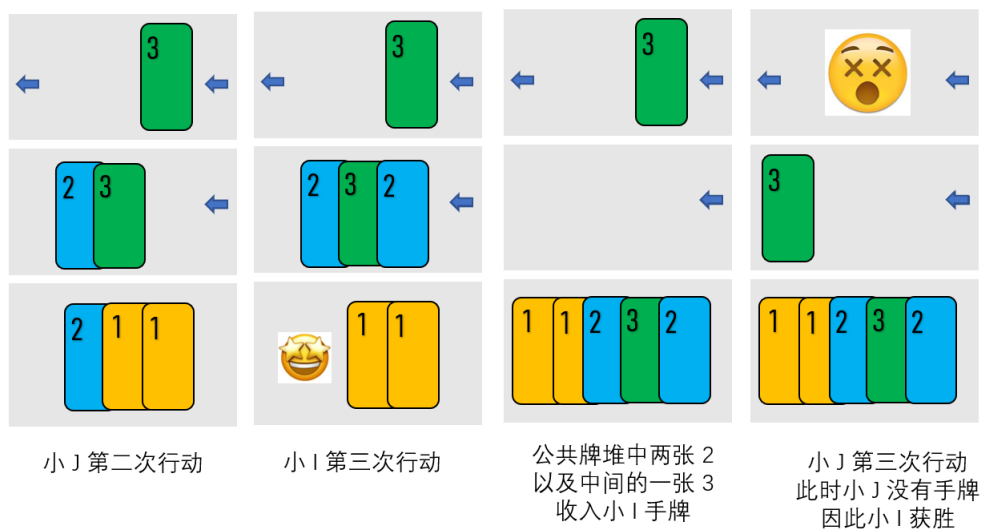
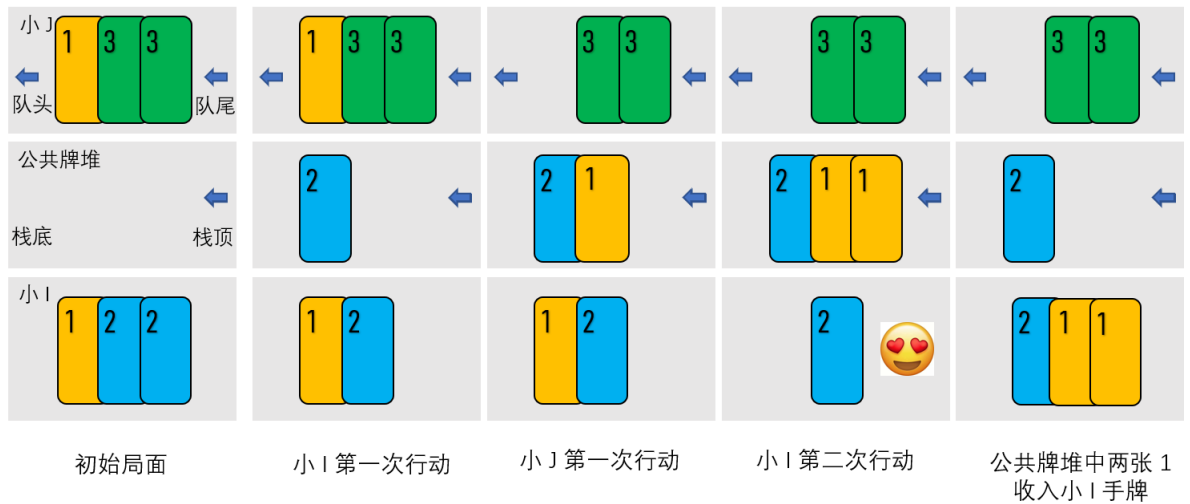
【样例 1 输入】

```
1 3
2 1 3 3
```

【样例 1 输出】

```
1 3
2 2 1 2
```

【样例 1 解释】



【样例 2 输入】

```
1 1
2 1
```

【样例 2 输出】

```
1 -1
```

【子任务】

对于所有测试数据， $1 \leq n \leq 3 \times 10^5$ ， $1 \leq a_1, a_2, \dots, a_n \leq n$ ，且每个整数在序列 a 中至多出现两次。

F . 公平合作 / F

时间限制： 1.0 秒

【题目描述】

在大地的尽头，一座灰白的灯塔矗立在漫长的海岸线上。这一片海域海流复杂、礁石嶙峋，却又是不少航线的必经之路。若没有如此高耸而明亮的灯塔为过路的船只照亮航路，或许会有更多不幸的生命葬身海底。为了看管好这一座海上明灯，一批训练有素的守望人轮流值守于此。日复一日的工作枯燥乏味却又不能有丝毫闪失，紧绷的神经直到下一班守望人到来才得以放松。

在电力普及之前，灯塔通常使用煤油灯为过往的水手指引前行的方向。每次为这座灯塔添加燃油时，需要两位守望人各自搬运一个容积为 L 的油桶；而每次轮到 Y 和 S 所在的班组照料这座灯塔时，总是由 Y 和 S 负责为灯塔加油。将煤油搬运至灯室时，如果不装满油桶，对灯塔的正常运转也没有太大影响，无非是需要多来回搬运几趟。但是，如果两位守望人都想着偷懒，问题恐怕就不只是多几趟那么简单。Y 和 S 想到了一个好办法：互相为对方的油桶装油。

灯塔里有 N 个用于将储存的煤油转移到油桶中的容器，其中第 i 个容器的容积为 a_i 。Y 和 S 先想办法决定由谁先装油。两人先后装油；轮到其中一位守望人装油时，这位守望人每次从所有容器中等概率地随机选出一个容器，将其装满油，并全部倒入对方的油桶中。两位守望人都可以在操作任意多次（可以是 0 次）后结束装油，但后手必须等先手结束后才能开始装油。Y 和 S 先后装完煤油后，两人会比一下谁把对方的油桶装得更满，再各自把自己的油桶搬运到灯室。但是，如果有谁某次选出一个容器后，把对方的油桶装满了，但容器里还有没倒出的煤油，那么这位倒霉的守望人就必须把两个油桶都独自搬到灯室——这也算是为单调的生活平添了几分乐趣。显然，如果先手某次随机选中的容器会使得油桶溢出，那么后手可以往先手的油桶里面装任意量的煤油，然后幸灾乐祸；因此我们约定：当先手溢出时，必定由先手搬两个油桶。

现在只剩下了一个问题：当 Y 和 S 都采取最优策略，使得对方搬的煤油尽可能地比自己多的时候，先手搬的煤油不多于后手的概率是多大？

【输入格式】

从标准输入读入数据。

输入的第一行包括两个正整数 N 和 L ，分别表示转移用的容器数量和油桶的容积。保证 $1 \leq N \leq 2000$ ， $1 \leq L \leq 10^9$ 。

输入的第二行包括 N 个正整数 a_1, \dots, a_N ，分别表示每个转移用的容器的容积。保证 $1 \leq a_i \leq 2000$ 。

【输出格式】

输出到标准输出。

输出一个实数，表示先手搬的煤油不多于后手的概率。当你的输出与标准输出的绝对误差不超过 10^{-6} 时，我们认为你的输出是正确的。

【样例 1 输入】

```
1 2 4
2 1 2
```

【样例 1 输出】

```
1 0.687500000000000000
```

【样例 1 解释】

可以证明，此时先手的策略一定是装满对方的油桶，且装满时必胜。经过若干次随机抽取后，能恰好将对方的油桶装满的概率为：

$$\left(\frac{1}{2}\right)^2 + \binom{3}{1} \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 = \frac{11}{16} = 0.6875.$$

【样例 2 输入】

```
1 4 1999
2 8 8 12 22
```

【样例 2 输出】

```
1 0.486400000000000000
```

【样例 3 输入】

```
1 13 31416
2 3 14 15 92 65 35 89 79 32 38 46 26 433
```

【样例 3 输出】

1 0.345160481227743274

【子任务】

对于 100% 的数据，保证 $1 \leq N \leq 2000$ ， $1 \leq L \leq 10^9$ ， $1 \leq a_i \leq 2000$ 。

G . 喵了个喵 II / G

时间限制： 3.0 秒

【题目背景】

本来这题的题面和《喵了个喵》有关的。但是听说有人嫌题面长，就少说点好了。

【题目描述】

给定一个长为 $4n$ 的序列，其中 $1 \sim n$ 各出现 4 次。问是否能够将其划分为两个相等的子序列。

【输入格式】

从标准输入读入数据。

第一行一个正整数 n 。

第二行 $4n$ 个正整数，表示序列。保证 $1 \sim n$ 各出现 4 次。

【输出格式】

输出到标准输出。

如果不能划分为两个相等的子序列，输出一行 **No**。

否则第一行输出 **Yes**。第二行输出一个长为 $4n$ 的 **01** 串。其中第 i 位表示原序列的第 i 个数被划分到第几个子序列。你需要保证你划分出来的两个子序列完全相等。

【样例输入】

```
1 2
2 1 1 2 1 2 2 1 2
```

【样例输出】

```
1 Yes
2 10000111
```

【样例解释】

两个子序列均为 $(1, 2, 1, 2)$ 。

【子任务】

保证 $1 \leq n \leq 5 \times 10^4$ 。

保证序列中 $1 \sim n$ 各出现 4 次。

【评分方式】

你的输出的第一行需要与标准答案一致。若为 Yes，输出任意一种合法的划分均算正确。

H . 背包 / H

时间限制： 1.0 秒

【题目背景】

本题中，你需要解决完全背包问题。

有 n 种物品，第 i 种物品单个体积为 v_i 、价值为 c_i 。

q 次询问，每次给出背包的容积 V ，你需要选择若干个物品，每种物品可以选择任意多个（也可以不选），在选出物品的体积的和恰好为 V 的前提下最大化选出物品的价值的和。你需要给出这个最大的价值和，或报告不存在体积和恰好为 V 的方案。

为了体现你解决 NP-Hard 问题的能力， V 会远大于 v_i ，详见数据范围部分。

【输入格式】

从标准输入读入数据。

第一行两个整数 n, q ，表示物品种数和询问次数。

接下来 n 行每行两个整数 v_i, c_i 描述一种物品。

接下来 q 行每行一个整数 V 描述一次询问中背包的体积。

【输出格式】

输出到标准输出。

对于每组询问输出一行一个整数。若不存在体积和恰好为 V 的方案，输出 **-1**；否则输出最大的选出物品的价值和。

【样例 1 输入】

```
1 2 2
2 6 10
3 8 15
4 100000000001
5 100000000002
```

【样例 1 输出】

```
1 -1
2 187500000000
```

【样例 1 解释】

第二组询问的最优方案为：选择 3 个物品 1 和 12499999998 个物品 2。

【子任务】

对于所有测试数据， $1 \leq n \leq 50, 1 \leq v_i \leq 10^5, 1 \leq c_i \leq 10^6, 1 \leq q \leq 10^5, 10^{11} \leq V \leq 10^{12}$ 。

I . 乱西星上的空战 / I

时间限制： 5.0 秒

【题目背景】

众所周知，在乱西星的现代战争中，制空权是很重要的。为此，人们发展出了无人机技术——遗憾的是，由于乱西星的算法水平和各种意义上的通讯水平实在太落后了，所以这些无人机只能采用各自独立的傻瓜式战斗模式，这些模式不含任何随机因素，因此一场傻瓜式无人机空战总是几乎能在开始前就被预测到结果。

总而言之，乱西星上正在相互战争的 $|A|$ 国和 (A) 国的傻瓜式无人机战斗机编队在其边境交界处的空域遭遇了，现在两国军方希望你能预测这一空战的结果。

【题目描述】

空域与时刻

由于乱西星的神秘物理法则，乱西星的时间和空间并不是连续的；若认为遭遇战开始的时刻是第 1 时刻，那么对于任意的第 $k \in \mathbb{Z}$ 个时刻，在这一时刻开始和结束时，一个物体（无人机或导弹）只能在形如 $(x, y, z) \in \mathbb{Z}^3$ 的位置（即空域内的整点）上。

无人机

由于空域相比无人机要大得多，因此我们可以将无人机视为一个质点（尽管他们实际上长得和地球上的一般意义上的飞机十分相似）。

飞行状态 在每个时刻，一架无人机的飞行状态可以用以下三组参数描述：

1. 当前时刻所在的坐标 $\vec{p} = (x, y, z) \in \mathbb{Z}^3$;
2. 当前时刻的飞行方向向量 $\vec{d}, \|\vec{d}\| = 1$;
 - 其中， $\|\vec{v}\|$ 表示向量 \vec{v} 的长度：设 $\vec{v} = (v_x, v_y, v_z)$ ，则 $\|\vec{v}\| = \sqrt{v_x^2 + v_y^2 + v_z^2}$ 。
 - 你可以简单地将 \vec{d} 理解为机头指向的方向。
3. 当前时刻的无人机升力线方向 $\vec{u}, \|\vec{u}\| = 1, \vec{u} \perp \vec{d}$;
 - 你可以简单的将 \vec{u} 理解为飞机所在平面的、从机腹指向机背的单位法向量。
 - 此时， \vec{d} 和 \vec{u} 可以唯一确定一个“左手向” $\vec{l} = \vec{u} \times \vec{d}$ 。

飞行性能 不严格地讲，一般而言，一架飞机通常有三个操作轴，即俯仰、滚转和偏航：俯（负杆）和仰（正杆）分别对应飞机机头向下和向上（即保持 \vec{l} 不变）；滚转即飞机以飞行方向为中轴线旋转（即保持 \vec{d} 不变）；偏航则为飞机机头向左或者向右（即保持 \vec{u} 不变）。由于无人机的特殊设计，其没有偏航轴，只能进行俯仰和滚转——容易看出，即使仅进行俯仰和滚转，一架无人机也能随意地改变 \vec{d} 和 \vec{u} （在保持 $\|\vec{u}\| = \|\vec{d}\| = 1, \vec{u} \perp \vec{d}$ 的前提下）。

以上的俯仰（正杆或负杆）、滚转操作，以及直线飞行，及其复合统称“机动”。

由于无人机型号差异，一架无人机的飞行性能可以用以下三组参数描述（为方便起见，在本节中，对进行一次机动前的飞行状态对应参数为 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ ，进行一次机动后的飞行状态对应的参数为 $\vec{p}' = (x', y', z'), \vec{d}', \vec{u}', \vec{l}'$ ）：

1. 正杆率 $\theta_u \in (\frac{\pi}{4}, \frac{\pi}{2})$ 和负杆率 $\theta_d \in (\frac{\pi}{4}, \frac{\pi}{2})$ ；
 - 若无人机**仅进行正杆**机动，则必须有 $\vec{p} = \vec{p}', \vec{l} = \vec{l}', \vec{u} \cdot \vec{d}' \geq 0$ ；此时，进行一次这样的机动花费的时间是 $\frac{\angle(\vec{d}, \vec{d}')}{\theta_u}$ 。
 - 若无人机**仅进行负杆**机动，则必须有 $\vec{p} = \vec{p}', \vec{l} = \vec{l}', \vec{u} \cdot \vec{d}' \leq 0$ ；此时，进行一次这样的机动花费的时间是 $\frac{\angle(\vec{d}, \vec{d}')}{\theta_d}$ 。
2. 滚转率 $\gamma \in (\frac{\pi}{4}, \frac{\pi}{2})$ ；
 - 若无人机**仅进行滚转**机动，则必须有 $\vec{p} = \vec{p}', \vec{d} = \vec{d}'$ ；此时，进行一次这样的机动花费的时间是 $\frac{\angle(\vec{u}, \vec{u}')}{\gamma}$ 。
3. 飞行极速 $v_m > 0$ ；
 - 若无人机**仅进行直线飞行**，则必须有 $\vec{d} = \vec{d}', \vec{u} = \vec{u}'$ ；此时，花费的时间是 $\frac{\|\vec{p}' - \vec{p}\|}{v_m}$ 。

合法位移 在每个时刻，若一架无人机可以从 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 这一飞行状态，严格按照**滚转**、**俯仰**（正杆或负杆）和**直线飞行**的顺序进行机动，使飞行状态变为 $\vec{p}' = (x', y', z') \neq \vec{p}, \vec{d}', \vec{u}', \vec{l}'$ ，满足 $\vec{d}' / (\vec{p}' - \vec{p})$ ，并且各机动花费的时间之和不超过 1，则称这是一次（无人机的）合法的综合机动。

如果一架无人机可以从 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 通过一次合法的综合机动，使飞行状态变为 $\vec{p}' = (x', y', z') \neq \vec{p}, \vec{d}', \vec{u}', \vec{l}'$ ，并且在所有使飞行状态变为 $\vec{p}'' = \vec{p}', \vec{d}'', \vec{u}'', \vec{l}''$ 的合法综合机动中，总用时是最短的，则称之为一次（无人机的）合法位移（或称该位移合法）。此时，无人机会沿直线从 \vec{p} 移动到 \vec{p}' 。如无特殊指明，下文中“位移”均默认（应当为）合法位移。

眼镜蛇机动 在每个时刻，无论无人机飞行性能如何，无人机总是可以通过眼镜蛇机动，从 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 这一飞行状态，变为 $\vec{p}' = \vec{p}, \vec{d}' = \vec{u}, \vec{u}' = -\vec{d}, \vec{l}' = \vec{l}$ 这一飞行状态。注意，这种机动不被视为合法机动。

其它参数 除此之外，一架无人机还具有以下参数：

1. 无人机编号（简称“编号”）；
 - 保证任意两架无人机编号不同。
2. 所在阵营（简称“阵营”）；
 - 所在阵营必须是 $|\backslash|$ 国或者是 $()$ 国中之一，并且双方互称敌方阵营。

坠毁 一架无人机坠毁，当且仅当其符合下列条件之一：

1. 在某激活的导弹位移过程中，与该导弹的距离不大于导弹的空爆距离（详见下文）。
2. 在某导弹位移结束后，与该导弹位置重合（从而导弹直接命中无人机导致坠毁，下同）。
3. 在无人机位移过程中，存在至少一枚激活的导弹，与其距离不大于该导弹的空爆距离。
4. 在无人机位移结束后，存在至少一枚导弹所在位置与其位置重合。
5. 在无人机位移结束后，存在另一架无人机与其坐标相同（从而发生碰撞导致双双坠毁）。

无人机坠毁后将立即消失，此后不会发射导弹，也不会导致其它无人机坠毁。但无人机已经发射的导弹不会立刻消失或爆炸。

此时也称无人机被摧毁。

空空红外制导导弹

类似的，一枚空空红外制导导弹（下文简称“导弹”）也可视为一质点，并且同样可以描述其飞行状态和飞行性能。

飞行状态 由于导弹无所谓上下左右，因此仅需要以下两组参数以描述一个导弹的飞行状态：

1. 当前时刻所在的坐标 $\vec{p} = (x, y, z) \in \mathbb{R}^3$ ；
2. 当前时刻的飞行方向向量 $\vec{d}, \|\vec{d}\| = 1$ ；
 - 你可以简单地将 \vec{d} 理解为导弹弹头指向的方向。

飞行性能 同样由于一枚导弹无所谓上下左右，因此其不存在俯仰、滚转和偏航轴，其向各个方向改变 \vec{d} 的性能是相同的，此时统称仅改变 \vec{d} 的操作为”偏航”。其与直线飞行及复合统称“机动”。

因此一枚导弹的飞行性能可以用以下两组参数描述（为方便起见，在本节中，对进行一次机动前的飞行状态对应参数为 $\vec{p} = (x, y, z), \vec{d}$ ，进行一次机动后的飞行状态对应的参数为 $\vec{p}' = (x', y', z'), \vec{d}'$ ）：

1. 偏航率 $\theta_r \in (\frac{\pi}{4}, \frac{\pi}{2})$ ；
 - 若导弹**仅进行偏航**机动，则必须有 $\vec{p} = \vec{p}'$ ；此时，进行一次这样的机动花费的时间是 $\frac{\angle(\vec{d}, \vec{d}')}{\theta_r}$ ；
2. 飞行极速 $v_m > 0$ ；
 - 若导弹**仅进行直线**飞行，则必须有 $\vec{d} = \vec{d}'$ ；此时，花费的时间是 $\frac{\|\vec{p}' - \vec{p}\|}{v_m}$ 。

合法位移 在每个时刻，若一枚导弹可以从 $\vec{p} = (x, y, z), \vec{d}$ 这一飞行状态，严格按照**偏航**和**直线飞行**的顺序进行机动，使飞行状态变为 $\vec{p}' = (x', y', z') \neq \vec{p}, \vec{d}'$ ，满足 $\vec{d}' // (\vec{p}' - \vec{p})$ ，并且各机动花费的时间之和不超过 1，则称这是一次（导弹的）合法位移（或称该位移合法）。此时，导弹会沿直线从 \vec{p} 移动到 \vec{p}' 。如无特殊指明，下文中“位移”均（应当）默认为合法位移。

其它参数 除此之外，一枚导弹还具有以下参数：

1. 保险距离 $d_s > 0$ 和激活状态；
 - 导弹被发射后立即处于未激活状态。
 - 每个时刻结束时，若导弹处于未激活状态，并且发射该导弹的无人机已坠毁，或者与发射该导弹的无人机的距离大于保险距离 d_s 时，进入激活状态。此后将保持激活状态，并称该导弹被激活，或称其为一枚激活的导弹。
2. 空爆距离 $d_p > 0$ ；
 - 每次导弹位移过程中，当一枚激活的导弹与任一无人机（包括发射该导弹的无人机）距离不大于 d_p 时，该导弹会进入可空爆状态（详见下文“可空爆”）。
 - 每次无人机位移过程中，若存在一无人机与一枚激活的导弹距离不大于 d_p 时，该导弹也会进入可空爆状态。
3. 最大锁定角 $\beta_s \in (\frac{\pi}{4}, \frac{\pi}{2})$ ；
 - 任意时刻，一枚飞行状态为 $\vec{p} = (x, y, z), \vec{d}$ 、最大锁定角的导弹能锁定到 \vec{p}' 处的无人机，当且仅当 $\vec{d} \cdot (\vec{p}' - \vec{p}) > 0$ ，并且 $\angle(\vec{d}, \vec{p}' - \vec{p}) \leq \beta_s$ 。
 - 此时称该无人机能被该导弹锁定，或称其在导弹的锁定范围内。
 - 称 $\angle(\vec{d}, \vec{p}' - \vec{p})$ 为锁定角。
4. 制导时长 $t_z > 0$ ；
 - 若导弹在第 k 个时刻被发射，则到第 $k + t_z$ 个时刻结束时，若导弹仍未爆炸，则导弹会立刻爆炸。此时称导弹超过制导时长。

爆炸、消失与可空爆 一枚导弹在符合下列全部条件时，会立刻爆炸并消失：

1. 在导弹位移开始前，导弹处于激活状态；
2. 符合以下条件之一：
 1. 在该导弹位移过程中，存在一架位于 \vec{q} 的无人机，使 $\min_{\lambda \in [0,1]} \|\lambda \vec{p}' + (1 - \lambda) \vec{p} - \vec{q}\| \leq d_p$ ，其中 \vec{p}, \vec{p}' 为导弹本次位移的起点和终点。
 - 此时，所有这样的无人机都会被该导弹摧毁。同时，一架无人机可能同时被若干枚导弹摧毁。
 2. 在无人机位移过程中，存在一架无人机，记其从位置 \vec{q} 位移到 \vec{q}' ，满足 $\min_{\lambda \in [0,1]} \|\lambda \vec{q}' + (1 - \lambda) \vec{q} - \vec{p}\| \leq d_p$ ，其中 \vec{p} 为导弹此时的位置。

- 此时，所有这样的无人机都会被该导弹摧毁。同时，该导弹也可能同时摧毁若干这样的无人机。

此时，称该导弹可空爆，或该导弹进入可空爆状态。

一枚导弹在符合下列条件之一时，不会发生爆炸，但是会在当前时刻结束时消失：

1. 导弹脱锁（见下文“导弹脱锁”），并且在当前时刻开始时已被激活；
2. 导弹超过制导时长；
3. 导弹未激活，并且导弹位移结束后与一无人机位置重合；
 - 此时，该无人机会被这枚导弹摧毁。同时，一架无人机可能同时被若干枚导弹摧毁。
4. 导弹未激活，无人机位移结束后，与一无人机位置重合。
 - 此时，该无人机会被这枚导弹摧毁。同时，一枚导弹可能同时摧毁若干这样的无人机。

无人机视野、雷达搜索与导弹锁定

无人机视野 任意时刻，一架飞行状态为 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 的无人机能够发现一架位于 $\vec{p}' = (x', y', z')$ 的无人机，当且仅当 $\vec{d} \cdot (\vec{p}' - \vec{p}) > 0$ ；此时称 \vec{p}' 处的无人机在 \vec{p} 处无人机的视野内。

无人机机载雷达搜索范围 一架无人机的机载雷达（下文简称“雷达”）的扫描范围可以用以下两个参数描述：

1. 水平扫描范围 $L_x \in \mathbb{R}^+$ 和垂直扫描范围 $H_y \in \mathbb{R}^+$ ；
 - 任意时刻，一架飞行状态为 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 、雷达扫描范围为 L_x, H_y 的无人机的能够扫描到一架位于 \vec{p}' 的无人机，当且仅当， $\vec{d} \cdot (\vec{p}' - \vec{p}) > 0$ 并且 $x, y, s.t. |x| \leq L_x, |y| \leq H_y$ 且 $[\vec{p}' - (\vec{p} + x\vec{l} + y\vec{u})] // \vec{d}$ 。
 - 即若以 \vec{p} 为原点、以 \vec{l} 和 \vec{u} 为 X, Y 轴建一平面 $\alpha = \alpha(\vec{p}; \vec{l}, \vec{u})$ ，则 \vec{p}' 在这一平面上的投影 $\vec{r} = P(\vec{p}'; \alpha)$ 应当落在 $[-L_x, L_x] \times [-H_y, H_y]$ 中。
 - 此时称 \vec{p}' 处的无人机在 \vec{p} 处无人机雷达扫描范围内。

导弹脱锁 当无人机位移结束后，若一枚导弹选定的目标已坠毁，或其不能被该导弹锁定，则称该导弹脱锁，或处于脱锁状态。

此后将一直保持脱锁状态，无论是否此前选定的无人机是否重新可以被导弹锁定。

无人机选定目标策略

任意时刻，无人机（简称“本机”，下同）按下述策略选择目标无人机。

1. 若本机视野内无敌方阵营无人机（简称“敌机”，下同），则本机无选定目标；
2. 否则，若上一时刻本机选择的无人机仍位于本机视野内，则本机仍选定该目标；
3. 否则，若存在至少一架敌机处于本机雷达扫描范围内，则选取其中与本机距离最近的；若与本机距离最近的敌机不唯一，则选取编号最小的。

4. 否则, 对视野内的处于 \vec{p} 的敌机, 记 $\alpha = \alpha(\vec{p}; \vec{u}, \vec{l}), \vec{r} = P(\vec{p}; \alpha) = (r_x, r_y)$, 则选取 $\min\{|r_x - L_x|, |r_x + L_x|\} + \min\{|r_y - H_y|, |r_y + H_y|\}$ 最小的。若有多个最小值, 则同样选择编号最小的。

飞行策略

无人机飞行策略 设无人机飞行状态是 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$, 其飞行极速为 v_m , 机载雷达扫描范围为 L_x, H_y 。

1. 若无人机有位于 \vec{p} 的选定目标:

1. 若无人机能够合法地移动到某个位置, 使敌机现在的位置 \vec{p} 仍处于本机的视野内, 则无人机会合法地移动到飞行状态 $\vec{q} = (x_q, y_q, z_q), \vec{d}_q, \vec{u}_q, \vec{l}_q$, 使敌机现在的位置 \vec{p} 仍处于本机视野内, 且 $\|\vec{p} - \vec{q}\|$ 最小。

1. 若有多个这样的位置, 记 $\alpha_q = \alpha(\vec{q}; \vec{u}_q, \vec{l}_q), \vec{r}_q = P(\vec{p}; \alpha_q) = (r_{qx}, r_{qy})$, 则优先选取使 $\vec{r}_q = (r_{qx}, r_{qy}) \in [-L_x, L_x] \times [-H_y, H_y]$ 的位置:

1. 若仍有多个这样的位置, 则选取使 $\|\vec{r}_q\| = \sqrt{r_{qx}^2 + r_{qy}^2}$ 最小的;
2. 若仍有多个这样的位置, 则选取 \vec{q} 字典序最小的。

2. 若不存在这样的 \vec{r}_q , 则选取使 $\min\{|r_{qx} - L_x|, |r_{qx} + L_x|\} + \min\{|r_{qy} - H_y|, |r_{qy} + H_y|\}$ 最小的;

1. 若仍有多个这样的位置, 则选取 \vec{q} 字典序最小的。

2. 否则, 无人机会合法地移动到某个位置 \vec{q} , 使 $\|\vec{q} - \vec{p} - v_m \vec{d}\|$ 最小;

1. 若有多个这样的位置, 则选取 \vec{q} 字典序最小的。

2. 否则, 无人机通过眼镜蛇机动, 将飞行状态变为 $\vec{p} = (x, y, z), \vec{u}, -\vec{d}, \vec{l}$ 。

保证在上述 1. 的情况下, 无人机总能合法地移动到某个位置。

导弹飞行策略 设导弹当前时刻飞行状态 \vec{p}, \vec{d} , 其选定的敌机飞行状态为 $\vec{p}', \vec{d}', \vec{u}', \vec{l}'$;

若上一时刻结束时, 导弹未脱锁, 则记 \vec{q}' 为敌机根据其飞行策略, 下一时刻会移动到的位置。

若导弹能合法地移动到 \vec{q}' , 则导弹会直接位移到 \vec{q}' 。

否则, 导弹会合法地移动到能使敌机位移后的位置 \vec{q}' 处于锁定范围内的位置 \vec{q} , 且 $\|\vec{q} - \vec{q}'\|$ 最小。

1. 若有多个这样的 \vec{q} , 则选取位移后锁定角最小的;
2. 若仍有多种可能, 则选取 \vec{q} 字典序最小的。

若不存在这样的位置, 或者上个时刻结束时, 导弹已经脱锁, 则导弹会合法地位移到某个位置 \vec{q} , 使 $\|\vec{q} - \vec{p} - v_m \vec{d}\|$ 最小。

保证导弹总能合法地移动到某个位置。

无人机发射导弹规则

一飞行状态为 $\vec{p} = (x, y, z), \vec{d}, \vec{u}, \vec{l}$ 的无人机（简称“本机”）向被本机选定的、处于 \vec{p} 的目标无人机（简称“敌机”）发射导弹的规则为：

在每个时刻开始时，若选定的敌机已处于本机雷达扫描范围内，且当前不存在由本机发射且未爆炸（或消失）的导弹，则向敌机发射一初始飞行状态为 $\vec{p}, \vec{d} = \frac{\vec{p} - \vec{p}}{\|\vec{p} - \vec{p}\|}$ 的未激活的导弹，该导弹选定敌机。

同一时刻内各事件发生顺序

1. 所有无人机选定目标，并确定当前时刻内的飞行策略；
2. 所有能发射导弹的无人机发射导弹；
3. 所有导弹确定飞行策略并位移，该过程中部分无人机可能被摧毁；
4. 所有可空爆的导弹爆炸并消失；
5. 所有无人机按 1. 中确定的飞行策略位移，该过程中部分无人机可能被摧毁；
6. 所有可空爆的导弹爆炸并消失；
7. 所有位置相同的无人机发生碰撞并坠毁。
8. 所有超过制导时长和脱锁且已激活的导弹消失。
9. 所有可激活的导弹被激活。

任务

给定空域开始时（即第 1 时刻开始时），各无人机的飞行性能与状态、导弹的飞行性能，假定这场空战会持续 T 个时刻，双方指挥官希望你能按时间顺序依次给出每个时刻发生的所有重要事件。

【输入格式】

从标准输入读入数据。

第一行两个正整数 n, T ，表示共有 $2n$ 架无人机，模拟前 T 个时刻。其中，前 n 架阵营是 $|$ 国，后 n 架阵营是 $()$ 国。

接下来有 $2n$ 组数据，每一组包含若干行，其中第 i 组描述了编号为 i 的无人机。

在每组数据中：

第一行三个整数表示 $\vec{p} \in \mathbb{Z}^3$ ；保证所有的 \vec{p} 两两不同，且坐标的绝对值不超过 100。

第二行六个整数依次表示无人机的 \vec{d}, \vec{u} ，保证 $\vec{d}, \vec{u} \in S_v$ ；

• 其中， $S_v = \{(1, 0, 0), (-1, 0, 0), (0, 1, 0), (0, -1, 0), (0, 0, 1), (0, 0, -1)\}$ ；

第三行六个实数依次表示无人机的 $\theta_u, \theta_d, \gamma, v_m, L_x, H_y$ ；

第四行五个实数和一个正整数依次表示导弹的 $\theta_r, v_m, d_s, d_p, \beta_s, t_z$ ；

【输出格式】

输出到标准输出。

输出 T 组数据，第 i 组数据表示第 i 个时刻发生的重要事件。

在每组数据中：

第一行两个非负整数 p_1, p_2, p_3 ，表示在这个时刻的导弹位移过程中被摧毁的无人机数量、在这个时刻的无人机位移过程中被摧毁的无人机数量、这个时刻结束时，有多少组无人机因位置相同而两两碰撞坠毁。

接下来 p_1 行，每行形如 $id_0 k id_1 id_2 \cdots id_k$ ，表示编号为 id_0 的无人机在这个时刻的导弹位移过程中被摧毁，并且在该过程中摧毁该无人机的导弹共有 k 枚，分别来自编号为 id_1, id_2, \cdots, id_k 的无人机。

为保证输出唯一，这 p_1 行中的每一行内， id_1, \dots, id_k 应当从小到大输出，行之间按 id_0 从小到大输出。

接下来 p_2 行，每行形如 $id_0 k id_1 id_2 \cdots id_k$ ，表示编号为 id_0 的无人机在这个时刻的无人机位移过程中被摧毁，并且在该过程中摧毁该无人机的导弹共有 k 枚，分别来自编号为 id_1, id_2, \cdots, id_k 的无人机。

为保证输出唯一，这 p_2 行中的每一行内， id_1, \dots, id_k 应当从小到大输出，行之间按 id_0 从小到大输出。

接下来 p_3 行，每行形如 $k id_1 id_2 \dots id_k$ ，表示该时刻结束时，有 k 架无人机位置相同，它们的编号是 id_1, \dots, id_n 。

为保证输出唯一，这 p_3 行中的每一行内， id_1, \dots, id_k 应当从小到大输出，行之间按 id_1 从小到大输出，并且每个编号出现至多一次。

【样例 1 输入】

```

1 1 1
2 0 0 0
3 1 0 0 0 0 1
4 1 1 1 4 1 1
5 1 3 1 1 1 1
6 8 0 0
7 -1 0 0 0 0 1
8 1 1 1 4 1 1
9 1 3 1 1 1 1

```

【样例 1 输出】

```

1 0 0 1
2 2 1 2

```

【样例 1 解释】

在第 1 时刻，两架飞机于 (4, 0, 0) 处相撞。

【样例 2 输入】

```
1 1 4
2 0 0 0
3 1 0 0 0 0 1
4 1 1 1 3 1 1
5 1 15 3 2 1 10
6 60 0 0
7 -1 0 0 0 0 1
8 1 1 1 3 1 1
9 1 15 3 2 1 10
```

【样例 2 输出】

```
1 0 0 0
2 0 0 0
3 0 0 0
4 0 2 0
5 1 1 2
6 2 1 1
```

【样例 2 解释】

在第 4 时刻，两枚导弹分别摧毁了敌机。

【数据范围】

$T, n \leq 100, 3 \leq v_m \leq 20$;

$v_m > 10$ 的无人机和导弹总数不超过 10;

$\theta_u, \theta_d, \gamma, \theta_r, \beta_s \in (\frac{\pi}{4}, \frac{\pi}{2})$;

$1 \leq d_s, d_p \leq 20, 1 \leq t_z \leq 100$;

$|x|, |y|, |z| \leq 100$;

$1 \leq L_x, H_y \leq 100$ 。

所有输入的实数精确到小数点后不超过 6 位。

最初时， \vec{p} 两两不同。

J . LIAR GAME! / J

时间限制： 1.0 秒

【题目背景】

在《LIAR GAME》中，小 E 看到了一个有趣的游戏。

【题目描述】

这个游戏名叫《走私游戏》。游戏规则大概是这样的：一名玩家扮演走私者，一名玩家扮演检察官。走私者可以将 x 日元 (x 为 $[0, n]$ 内的整数，由走私者决定) 秘密放入箱子中，而检查官需要猜测箱子中的金额。假设检查官猜了 y (y 也必须是整数)。如果 $x = y$ ，则走私失败，走私者一分钱也拿不到。如果 $x > y$ ，则走私成功，走私者可以从检查官那里拿走 x 日元。如果 $x < y$ ，则走私失败，但是由于冤枉检查官需要赔付给走私者 $y/2$ 日元。游戏分有限回合进行。双方轮流做走私者和检查官。

可以证明，最优情况下每个回合走私者会采用同一种策略，检查官也会采用同一种策略。小 E 想知道在一个回合中，双方的最优策略分别是什么。

【输入格式】

从标准输入读入数据。

一行一个正整数 n 。

【输出格式】

输出到标准输出。

输出两行，每行 $n + 1$ 个数，其中第 i 个表示放/猜 $i - 1$ 日元的概率。

第一行输出走私者的策略，第二行输出检查官的策略。

你需要保证，在一方的策略不变的情况下，另一方无论如何改变自己的策略，都不能使自己的期望收益比原来多。

可以证明，这样的策略是唯一的。

答案对 998244353 取模。

【样例输入】

1 1

【样例输出】

```
1 665496236 332748118
2 332748118 665496236
```

【样例解释】

这 4 个数分别为 $2/3, 1/3, 1/3, 2/3$ 。

【子任务】

保证 $1 \leq n \leq 400000$ 。

K. 众数 / K

时间限制：0.5 秒

【题目描述】

你有若干个 $[1, n]$ 内的正整数：对于 $1 \leq i \leq n$ ，你有 a_i 个整数 i 。设 $S = \sum_{i=1}^n a_i$ 。

对于一个序列 p_1, p_2, \dots, p_l ，定义其众数 $\text{maj}(p_1, p_2, \dots, p_l)$ 为出现次数最多的数。若有多个数出现次数最多，则其中最大的数为其众数。

现在你需要把这 S 个数排成一个序列 b_1, b_2, \dots, b_S ，使得 $\sum_{i=1}^S \text{maj}(b_1, b_2, \dots, b_i)$ 最大。输出该最大值。

【输入格式】

从标准输入读入数据。

第一行一个整数 n ，表示值域。

接下来一行 n 个正整数 a_1, a_2, \dots, a_n ，表示每种数的个数。

【输出格式】

输出到标准输出。

输出一行一个正整数表示 $\sum_{i=1}^S \text{maj}(b_1, b_2, \dots, b_i)$ 的最大值。

【样例 1 输入】

```
1 3
2 1 3 2
```

【样例 1 输出】

```
1 17
```

【样例 1 解释】

一个达到最大值的序列为 $(3, 2, 3, 1, 2, 2)$ 。

【数据范围】

对于所有测试数据， $1 \leq n \leq 10^5$ ， $1 \leq a_1, a_2, \dots, a_n \leq 10^5$ 。

L . 最后的活动 / L

时间限制： 1.0 秒

【题目背景】

各位亲爱的《La Lumière: Scarlet Intense Flame》玩家：

感谢您一直给予《La Lumière: Scarlet Intense Flame》的支持与厚爱。我们非常遗憾地宣布，《La Lumière: Scarlet Intense Flame》将于 2023 年 3 月 5 日 16:00 停止运营服务。

停止运营相关时间表如下：

.....

【题目描述】

元老级二次元手游《La Lumière: Scarlet Intense Flame》将于今年 3 月停止运营服务。作为这款游戏的忠实玩家，小 S 希望能在游戏的最后一次活动中刷到一个特殊的分数，以此为近十年来与这款游戏共度的难忘时光画上一个圆满的句号。

《La Lumière: Scarlet Intense Flame》中的每种活动都有其独特的规则，而最后一次活动是 Chase Festival。在 Chase Festival 中，玩家需要多次攻略每次随机生成的多层迷宫，每次退出迷宫时根据在迷宫中各层击杀怪物的评价独立结算本次随机迷宫的分数。每次挑战迷宫时的流程简化如下：

1. 选择挑战的随机迷宫的难度。小 S 是这款游戏的资深玩家，因此在本题中假定小 S 总是挑战最高难度的迷宫。最高难度的迷宫最深为 N 层。确定难度后，从随机生成的迷宫的第 1 层开始挑战。
2. 进行第 i 层的挑战。挑战第 i 层时，小 S 有可能挑战失败，挑战成功并获得普通评价，或者挑战成功并获得高评价。如果小 S 选择保守的挑战策略，则有 $p_{i,0}$ 的概率挑战失败，有 $p_{i,1}$ 的概率挑战成功并获得普通评价，有 $p_{i,2}$ 的概率挑战成功并获得高评价；如果小 S 选择激进的挑战策略，则有 $q_{i,0}$ 的概率挑战失败，有 $q_{i,1}$ 的概率挑战成功并获得普通评价，有 $q_{i,2}$ 的概率挑战成功并获得高评价。
 - 获得普通评价时，在当前层获得 $s_{i,1}$ 的分数；获得高评价时，在当前层获得 $s_{i,2}$ 的分数。这部分获得的分数不会直接加算到玩家的总分数中，而是在退出迷宫时结算。如果挑战成功，且当前不是最后一层 ($i < N$)，则跳转到第 3 步，选择是否继续挑战；否则 ($i = N$)，退出迷宫并跳转到第 4 步进行结算。
 - 如果挑战失败，则强制退出迷宫，跳转到第 4 步。
3. 如果当前不是最后一层，玩家可以选择是否继续挑战下一层。如果选择继续，则返回第 2 步；否则退出当前迷宫，跳转到第 4 步进行结算。
4. 本次迷宫的分数结算：如果因为失败而强制退出，则当前层不获得任何奖励，且本次迷宫中之前各层累积的分数需要乘上惩罚系数 c （为了使最终分数为整数，

游戏会对惩罚后的分数先求和再下取整)；除了强制退出之外，玩家主动退出或者通关迷宫后退出都可以获得全部尚未结算的分数。

小 S 想得到的目标分数是一个比较大的分数，因此小 S 需要先大量刷最高难度的迷宫，再在接近目标分数时根据当前剩余的分数选择相对稳定的策略，以确保活动结束后能恰好获得目标分数。小 S 不会编程，因此小 S 找到了你，希望你能帮忙计算当剩余分数在 1 至 M 分之间，仅按照上述的流程挑战迷宫，并采用最佳策略时，最终能够恰好达到目标分数的最大概率。

【输入格式】

从标准输入读入数据。

输入的第一行包含三个整数 N, M, c' ，其中 N 和 M 的含义与题面相同， $c' = 100c$ 。保证 $1 \leq N \leq 6$ ， $1 \leq M \leq 10000$ ， $0 \leq c' \leq 100$ 。

接下来 N 行，每行输入八个整数 $s_{i,1}, s_{i,2}, u_{i,0}, u_{i,1}, u_{i,2}, v_{i,0}, v_{i,1}, v_{i,2}$ ，其中 $s_{i,1}$ 与 $s_{i,2}$ 分别表示挑战时普通评价和高评价对应的分数； $u_{i,j}$ 和 $v_{i,j}$ 分别表示使用保守的挑战策略及激进的挑战策略时，对应结果的概率权重： $p_{i,j} = \frac{u_{i,j}}{u_{i,0} + u_{i,1} + u_{i,2}}$ ， $q_{i,j} = \frac{v_{i,j}}{v_{i,0} + v_{i,1} + v_{i,2}}$ 。保证 $1 \leq s_{i,1} \leq s_{i,2} \leq 10000$ ， $0 \leq u_{i,j}, v_{i,j} \leq 10000$ 且 $u_{i,1} + u_{i,2} \geq 1$ ， $v_{i,1} + v_{i,2} \geq 1$ 。

【输出格式】

输出到标准输出。

输出一行 M 个实数，其中第 i ($1 \leq i \leq M$) 个实数表示当距离目标分数恰好还剩 i 分时，在最优策略下能够恰好获得 i 分的最大概率。当你输出中的每个实数与相应标准输出的绝对误差不超过 10^{-6} 时，我们认为你的输出是正确的。

【样例 1 输入】

```
1 2 8 50
2 3 4 0 1 1 0 1 1
3 4 5 1 2 1 1 1 2
```

【样例 1 输出】

```
1 0.125000000000000000 0.140625000000000000 0.515625000000000000
   0.564453125000000000 0.135009765625000000 0.328369140625000000
   0.548858642578125000 0.625278472900390625
```

【样例 2】

见题目目录下的 *2.in* 与 *2.ans*。

【子任务】

对于 100% 的数据，保证 $1 \leq N \leq 6$ ， $1 \leq M \leq 10000$ ， $0 \leq c' \leq 100$ ， $1 \leq s_{i,1} \leq s_{i,2} \leq 10000$ ， $0 \leq u_{i,0}, u_{i,1}, u_{i,2}, v_{i,0}, v_{i,1}, v_{i,2} \leq 10000$ ， $u_{i,1} + u_{i,2} \geq 1$ ， $v_{i,1} + v_{i,2} \geq 1$ 。

【提示】

《La Lumière: Scarlet Intense Flame 2》将于 2023 年春暖花开的时节与大家相见！

M . 世界杯 / M

时间限制： 1.0 秒

【题目描述】

2022 年卡塔尔世界杯已于 12 月 18 日落下帷幕，作为一项始于 1930 年的悠久赛事，迄今已走过 22 届，四年一度的足球盛宴，是陪伴全世界无数足球运动员和爱好者成长的荣耀时光和青春记忆。

作为一个新球迷，小 M 在这届世界杯前对足球几乎一无所知，但受到身边小伙伴的带动，他也决定加入到球迷的大家庭中。赛前，小 M 对世界杯进行了一定的了解，得知世界杯正赛共分为小组赛和淘汰赛，其中淘汰赛尤为刺激，16 强、8 强、半决赛、决赛，只要输一场便立刻淘汰出局，唯有全胜者方能捧得桂冠。因此，小 M 心想，夺得世界杯冠军当然是无可争议的世界巅峰，而能否从小组赛出线杀入淘汰赛、以及在淘汰赛阶段走得多远，也同样是衡量一支队伍强弱的重要指标。

世界杯正赛共有 32 支球队参加，而遗憾落选的队伍则更多。全世界这么多支球队，对于萌新小 M 来说，逐一去了解难免是一个浩大的工作。因此，小 M 决定先从联合国五常来入手，对应的 5 支球队分别为：中国队（China）、英格兰队（England）、法国队（France）、俄罗斯队（Russia）和美国队（United States）。值得一提的是，苏格兰队和威尔士队也曾打进过世界杯正赛，但以小 M 的知识水平来说，显然忽略了它们。

随后，小 M 仔细调查了这 5 支球队的历届世界杯战绩：英格兰队、法国队和美国队在近 30 年来均有至少 4 次杀入世界杯淘汰赛的经历，其中法国队收获了 1998、2018 世界杯的冠军，虽也有过小组赛 2 败出局的经历，但已足够向世界展示其强大；英格兰队唯一的世界杯冠军在 1966 年，近年来最好战绩为四强，而美国队近年来最好战绩为八强，这两支队伍的实力也都不容小觑；俄罗斯队近 30 年来的前 3 次世界杯之旅都不顺利，在小组赛阶段便早早接连失利并被淘汰，但 2018 年以东道主的身份打入淘汰赛并杀入八强；而中国队则实力稍逊，唯一一次打入世界杯正赛是在 2002 年，但小组赛三战全败，只得遗憾出局。

正像无数球迷心中都有一支“主队”，小 M 也希望从这 5 支球队中选出一支“最强球队”，作为他今后支持的对象。在他进一步了解后，也知道了评价一支球队有各项指标，这几支球队也似乎各有所长。到底该怎样衡量“最强”呢？小 M 想来想去，决定以“世界杯正赛中输球总场数最少”作为衡量依据。又经过了细致的查找资料和分析，他心中的“最强球队”已经呼之欲出。

此时，他想考考屏幕前的你：能猜出他心中的“最强球队”是哪一支吗？

【输入格式】

从标准输入读入数据。

无输入。

【输出格式】

输出到标准输出。

输出一个字符串，一定是 `China, England, France, Russia, United States` 中的一个，表示小 M 心中的“最强球队”。

【提示】

题面所述信息已经足以推断出问题的答案，无需任何额外的足球知识。