

## Problem A. 棋盘

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:          1024 megabytes

Alice 和 Bob……不！这次的游戏是 Mandy 和 brz 发现的，他们将要在一个  $2 \times n$  的棋盘上进行游戏。正如上面所说，他们找到了一个 2 行  $n$  列的棋盘，每个格子上都有个分数，第  $i$  行  $j$  列的格子上的分数是  $a_{i,j}$ 。Mandy 的棋子一开始在左上角（第 1 行第 1 列），brz 的棋子在右下角（第 2 行第  $n$  列）他们首先分别获得自己棋子所在位置的分数，然后这两个位置的分数就消失了。

然后两人交替操作，Mandy 先手。每次操作方可以将自己的棋子移动到相邻的一个格子上（上下左右其中之一），或者不移动。选择移动的话要求不能移出棋盘，也不能移动到没有分数的格子上。移动后操作方就获得了该格子上的分数，然后这个格子上的分数就消失了。

如果双方都没有办法移动了，则游戏结束。最后谁的分数总和最高，那么谁就获胜了。如果两人都是绝顶聪明的，都希望自己尽可能获胜（如果没法获胜，就尽量平局），那么请旁观的你提前计算出来，谁会最后的赢家呢？

### Input

第一行包含一个整数  $T$  ( $1 \leq T \leq 10^5$ )，表示 Mandy 和 brz 进行了  $T$  场游戏。

对于每场游戏，第一行包含一个整数  $n$  ( $1 \leq n \leq 2 \times 10^5$ )。

下面两行每行包含  $n$  个整数，表示  $a_{i,j}$  ( $1 \leq a_{i,j} \leq 10^9$ )。

保证对于  $T$  场游戏， $\sum n \leq 2 \times 10^5$ 。

### Output

输出  $T$  行，每行表示一场游戏的结果。

如果最后 Mandy 会获胜，那么输出 "Mandy"（不包含引号）；如果最后 brz 会获胜，则输出 "brz"（不包含引号）；否则输出 "draw"（不包含引号）表示平局。

### Example

standard input	standard output
3 1 3 2 3 1 2 3 3 2 1 10 3 2 1 3 3 2 3 9 4 2 1 3 1 1 4 3 9 3 4 1	Mandy draw brz

### Note

对于第一组样例，Mandy 一开始获得 3 分，brz 一开始获得 2 分，然后两人都无法移动了，于是游戏结束，Mandy 获胜。

对于第二组样例，两人一开始都获得 1 分，如果 Mandy 选择向下走获得 3 分，则 brz 可以选择向左走获得 2 分，然后 Mandy 就无法移动了，因为周围的格子的分数都已经被取走了，于是 brz 可以获得剩下的分数，于是就会获胜。所以 Mandy 一开始不会向下走，那么就只能向右走获得 2 分，brz 如果向上走，就会让 Mandy 获得剩下的所有分数然后 brz 就输掉了，所以 brz 会选择向左走获得两分，然后最后两人再各自获得 3 分，游戏结束。最后两人都是 6 分，所以是平局。

对于第三组样例，请自行分析。

## Problem B. 石楠花的约定

Input file:           standard input  
Output file:         standard output  
Time limit:          5 seconds  
Memory limit:       1024 megabytes

Rigel 很讨厌石楠花，尤其是在春季，武汉盛开的石楠花铺天盖地的"香气"令他不由自主地远离。  
很不妙的是，当 Rigel 与他的女友约会时，他竟惊恐地嗅到了石楠花的味道！Rigel 约会的地点可以看成一棵包含  $n$  个点的树的结构，在这棵树上有  $m$  个点都盛开着石楠花。现在他只想带着她的女友去往尽可能远离石楠花的地方去约会，否则他会当着他女友的面出糗的。  
不过令人欣喜的是，由于过于讨厌石楠花，Rigel 觉醒了一种特殊能力：他最多可以去掉这棵树上的  $k$  个点的石楠花，让他尽可能远离石楠花。  
请帮助 Rigel 解决这个问题，告诉他发动能力后树上的点离石楠花最远的距离是多远。  
具体来说，你需要去除  $k$  个点上的石楠花，设最后还有石楠花的节点是  $x_1, x_2, \dots, x_{m-k}$ ，你需要最大化  $\max_{i=1}^n \{\min_{j=1}^{m-k} \text{dis}(i, x_j)\}$ ，其中  $\text{dis}(x, y)$  表示树上节点  $x$  到节点  $y$  的距离。

### Input

第一行包含三个正整数  $n, m, k$  ( $1 \leq m \leq n \leq 10^5, 0 \leq k < m$ )。  
接下来  $n - 1$  行，每行两个正整数  $u, v$  ( $1 \leq u, v \leq n$ )，表示树上的一条边  $(u, v)$ 。  
接下来一行包含  $m$  个正整数  $a_i$  ( $1 \leq a_i \leq n$ )，表示这些点被石楠花占据。保证  $a_i$  两两互不相同。

### Output

共一行，输出 Rigel 发动能力后树上的点离石楠花最远的距离是多远。

### Example

standard input	standard output
7 3 1 1 2 1 3 2 4 3 5 3 6 5 7 3 4 5	3

### Note

将 4 号节点的石楠花去掉，那么树上离石楠花最远的节点是 4 号节点，距离为 3。可以证明方法唯一。

Problem C. 年少的誓约

Input file: standard input  
 Output file: standard output  
 Time limit: 1 second  
 Memory limit: 1024 megabytes

"如果你的总分数比我高，那我就答应和你在一起！"  
 年少的誓约如同她满怀期许的眼眸般闪烁，闪烁在那遥远而又模糊的记忆里……  
 重来一次，请问你能否不再留下遗憾？

简单来说，一共有  $n$  场考试，每场考试的重要程度各不相同，但每场考试的分数都是一个在  $[0, x]$  的整数。对于第  $i$  场考试，你和她的分数分别为  $a_i, b_i$ ，重要程度为  $c_i$ 。而她出于私心，将总分的计算公式改为  $\sum_{i=1}^n d_i \cdot (c_i + k \cdot a_i)$ ，其中  $d_i$  为第  $i$  场的考试分数：当计算你的总分时， $d_i = a_i$ ；当计算她的总分时， $d_i = b_i$ 。  
 请问当  $\sum_{i=1}^n a_i = m$  时，是否有一种分数的分配使得你最后的总分大于她的总分？

Input

第一行包含一个整数  $T$  ( $1 \leq T \leq 2 \cdot 10^5$ )，表示一共有  $T$  次询问。  
 每次询问都有  $n + 1$  行数据，第一行包含四个整数  $n, m, k, x$  ( $1 \leq n, x, k \leq 2 \times 10^5, 1 \leq m \leq 10^9$ )。接下来  $n$  行，每行包含两个整数  $b_i, c_i$  ( $0 \leq b_i \leq x, 1 \leq c_i \leq 2 \times 10^5$ )。  
 保证对于  $T$  组询问，有  $\sum n \leq 5 \times 10^5$ 。

Output

对于每组数据，输出一行 YES 或 NO（请注意大小写），表示是否能使你的总分大于她。  
 请注意，过程中的运算数据可能超过 64 位整数。

Example

standard input	standard output
2	YES
2 10 1 9	NO
8 2	
7 3	
2 10 1 8	
8 2	
7 3	

Note

对于第一次询问，若你两场考试的分数为 1 和 9，则你的总分为  $1 \times 3 + 9 \times 12 = 111$ ，而她的总分为  $8 \times 3 + 7 \times 12 = 108$ ，你的总分可以大于她。

## Problem D. 金麦园

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         1024 megabytes

众所周知，金麦园饺子馆有着 "北邮第二食堂" 的称号，它承载了北邮人对于美食的一种特殊情感。  
金麦园有  $n$  种种类的饺子，其中第  $i$  种饺子的美味度为  $a_i$ 。

Liangsheng 和 Liangcha 每周都会像 NPC 一样固定刷新在金麦园饺子馆，由于两人的口味并不相同，两人会点两种种类不同的饺子  $i$  和  $j$  ( $i \neq j$ )。

定义一种点餐的 "不整齐度" 为  $S_{i,j} = |a_i - a_j|$ ，现在 Liangsheng 请你选择  $k$  个不同的点餐方案，使得这  $k$  个点餐方案的 "不整齐度" 的和最小。

**注意：**两种点餐方案  $(i_1, j_1)$  和  $(i_2, j_2)$  不同当且仅当集合  $\{i_1, j_1\}$  和  $\{i_2, j_2\}$  不同，即  $(i, j)$  为无序数对。

### Input

第一行输入两个正整数  $n, k$  ( $2 \leq n \leq 10^6, 1 \leq k \leq \min\{\frac{n(n-1)}{2}, 10^{11}\}$ )，分别表示金麦园中饺子的种类数和点餐方案的个数。  
第二行输入  $n$  个正整数  $a_i$  ( $1 \leq a_i \leq 10^8$ )，表示第  $i$  种饺子的美味度。

### Output

输出一行一个整数，表示 "不整齐度" 之和的最小值。

### Example

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

### Note

选择  $(1, 4), (2, 3), (3, 4), (2, 5)$  这 4 对数，"不整齐度" 之和为 4。

## Problem E. 圣巢万神殿

Input file:           standard input  
Output file:          standard output  
Time limit:          2 seconds  
Memory limit:        1024 megabytes

嘿嘿.....

丝之歌.....

我真幸运.....

GeometryFeng 的朋友小涵最近在玩一款叫《空洞骑士》的游戏, 小涵以为自己很幸运, 刚玩完空洞骑士就能玩上它的第二代——《丝之歌》。但是小涵不知道的是, GeometryFeng 已经幸运了整整五年, 还是没有玩上丝之歌。

小涵在游戏里已经打到了圣巢万神殿。这个模式非常困难, 小涵需要连续击败  $n$  个 Boss 才能通关, 现在他想让你帮他算算他通关的期望挑战次数。

具体规则如下:

1. 小涵初始血量为满血  $h$ , 从头开始每成功打完  $k$  个 boss 会有一个休息点, 小涵会在休息点回满血。
2. 对于每一个 boss, 小涵都有概率被 boss 秒杀, 被秒杀的概率会受到掉血量的影响, 满血时概率为  $p_i$ , 每掉 1 滴血, 这个概率会乘上一个倍数  $x$ 。
3. 如果小涵被第  $i$  个 boss 秒杀, 那么本次挑战失败; 否则, 他有  $q_i$  的概率受伤, 受伤会掉 1 滴血, 若血量变为 0 则也挑战失败。
4. 每次挑战失败, 都需要从头开始 (即第一个 boss), 成功打败第  $n$  个 boss 并且血量不为 0 则挑战成功。

### Input

第一行输入四个整数  $n, h, k, x'$  ( $1 \leq n \leq 5 \times 10^4, 1 \leq h \leq 5 \times 10^4, 0 \leq k < n, 100 < x' \leq 200$ ), 意义如上所述, 其中  $x = \frac{x'}{100}$ 。特殊地,  $k = 0$  时表示不存在休息点。

第二行输入  $n$  个整数  $q'_i$  ( $0 \leq q'_i < 100$ ),  $q_i = \frac{q'_i}{100}$  表示小涵在第  $i$  个 boss 受伤的概率。

第三行输入  $n$  个整数  $p'_i$  ( $1 \leq p'_i < 100$ ),  $p_i = \frac{p'_i}{100}$  表示小涵满血时被第  $i$  个 boss 秒杀的概率。

数据保证小涵通关概率不为 0。

### Output

仅一行, 包含一个整数, 代表小涵通关的期望尝试次数对  $10^9 + 7$  取模的结果。

正式地说, 答案对  $10^9 + 7$  取模表达了如下含义。令  $M = 10^9 + 7$ , 可以证明答案可表示为既约分数  $\frac{p}{q}$ , 其中  $p$  和  $q$  均为整数, 且  $q \not\equiv 0 \pmod{M}$ 。你需要输出  $p \cdot q^{-1} \pmod{M}$ 。换句话说, 你需要输出满足  $0 \leq x < M$  且  $x \cdot q \equiv p \pmod{M}$  的整数  $x$ 。

Examples

standard input	standard output
3 10 0 150 0 0 0 50 50 50	8
1 1 0 101 25 50	666666674
3 2 2 150 50 50 50 10 10 10	899138140
2 2 1 200 60 60 50 50	4

Note

对于第一个样例，小涵满血时被 boss 秒杀的概率均为  $\frac{50}{100}$ ，受伤概率均为 0，中途没有休息点，故通关的概率为  $\left(1 - \frac{50}{100}\right)^3 = \frac{1}{8}$ ，因此期望挑战次数为 8。

对于第二个样例，只有一个 boss，小涵首先有  $\frac{50}{100}$  的概率被秒杀，否则，有  $\frac{25}{100}$  的概率受伤并掉一滴血，此时血量减为0，也会失败，故通关的概率为  $\left(1 - \frac{50}{100}\right) \times \left(1 - \frac{25}{100}\right) = \frac{3}{8}$ 。

对于第四个样例，小涵首先有  $\frac{50}{100}$  的概率打败第一个 boss，成功打败后有  $\frac{60}{100}$  概率受伤，接着小涵遇到休息点，无论受伤与否都会回满血，接着又有  $\frac{50}{100}$  的概率打败第二个 boss，故通关概率为  $\frac{50}{100} \times \frac{50}{100} = \frac{1}{4}$ 。

## Problem F. 不死国的生命树

Input file:           standard input  
Output file:          standard output  
Time limit:          3 seconds  
Memory limit:        1024 megabytes

吸食人脑的尤加特拉希宇宙树，神奇的密密尔智慧之泉，这一切，才刚刚开始……  
不死国的生命树是一棵以  $n$  个点和  $n - 1$  条边组成的有根树，根为 1。  
生命树的每个结点  $i$  上都长有一个价值为  $a_i$  的果实。  
为了破坏生命树的核心，主角唐晓翼将进行若干次旅行。他拥有一个特殊的背包，用来装载旅途中的果实，背包特性如下：

- 背包中果实的价值互不相同；
- 初始时，唐晓翼的背包中仅含有起点上对应的果实；
- 唐晓翼可以把价值为  $x$  的果实装进背包，当且仅当背包中含有价值为  $x - 1$  的果实；

现给出唐晓翼  $q$  条旅行路线：路线的起点为  $s$ ，终点为  $t$ ，并且保证  $t$  是  $s$  的祖先。在唐晓翼仅能严格按照  $s$  到  $t$  的简单路径进行旅行的前提下，请你帮助唐晓翼计算他最多可以获取多少个果实。  
注意：由于生命树强大的核心力量，果实被取走后会立刻生成一个同价值的新果实，即  $q$  次旅行相互独立，互不影响。

### Input

第一行输入一个正整数  $n$  ( $1 \leq n \leq 10^6$ )，表示生命树的结点数量。  
第二行输入  $n$  个正整数  $a_i$  ( $1 \leq a_i \leq 10^6$ )，分别表示  $n$  个果实的价值。  
第三行输入  $n - 1$  个正整数  $f_i$  ( $1 \leq f_i \leq n$ ) 表示第  $i$  个结点的父亲，保证输入是一棵树。  
第四行输入一个正整数  $q$  ( $1 \leq q \leq 5 \times 10^5$ )，表示唐晓翼的旅行次数。  
接下来  $q$  行，每行输入两个正整数  $s, t$  ( $1 \leq s, t \leq n$ ) 分别表示当次旅行的起点和终点，输入保证  $t$  为  $s$  的祖先。

### Output

输出  $q$  行，其中第  $i$  行输出第  $i$  次旅行的答案，即唐晓翼在该次旅行中最多获取多少个果实。

### Example

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

### Note

第一次询问，背包只能装下起点的价值为 2 的果实。  
第二次询问，背包可以装下路径中所有的果实，其中果实的价值分别为 3, 4, 5。



## Problem G. 精确的浪漫

Input file:           standard input  
Output file:          standard output  
Time limit:          1 second  
Memory limit:        1024 megabytes

Mandy 和 brz 来到吧吧国旅游, 这个国家可以用一个二维平面描述, 国家的  $x$  轴边界是  $-10^3$  和  $3 \times 10^3$ ,  $y$  轴边界也是  $-10^3$  和  $3 \times 10^3$ , 这意味着这两人只能在  $x, y \in [-10^3, 3 \times 10^3]$  的范围内走动, 高大的城墙不允许他们随意走出国家边界。

吧吧国有一项古老而浪漫的传统: 如果你爱一个人, 就要给她送一个喜欢的数字。Mandy 已经选好了一个数字, 在吧吧国的一处等待着 brz, 而 brz 也拿着一个初始数字在另一处站好了, 这个数字未必是 Mandy 想要的, 为了精确送给 Mandy 她想要的数字, brz 还要经过一项传统仪式, 然后在仪式中完成送数字, 这项仪式被称为"精确的浪漫"。

在仪式中, brz 可以携带着他的数字进行移动, 每次移动可以向上、下、左、右某个方向移动一个单位, 然后数字会受到仪式力量的影响, 发生一些变化。假设 brz 当前的坐标为  $(x, y)$ , 具体为:

1. 向左移动。坐标变为  $(x - 1, y)$ , brz 手里的数字会  $-1$ 。
2. 向右移动。坐标变为  $(x + 1, y)$ , brz 手里的数字会  $+1$ 。
3. 向上移动。坐标变为  $(x, y + 1)$ , brz 手里的数字会  $\times 2$ 。
4. 向下移动。坐标变为  $(x, y - 1)$ , brz 手里的数字会  $\div 2$  (向 0 取整)。

brz 最后的目标是, 恰好走到 Mandy 所在的位置, 并且此时手里的数字和 Mandy 所希望的一样, 于是 brz 就可以送出他的"精确的浪漫"。

除此之外, 仪式还有一些额外的限制:

- brz 初始选取的位置和 Mandy 的位置需要满足  $x, y \in [0, 2 \times 10^3]$ , brz 在移动过程中可以离开这个范围, 但是不能走出国家的边界。
- brz 在仪式过程中, 手里的数字可以变成负数, 但是绝对值必须保持小于等于  $10^9$ , 否则数字会超出仪式的能力范围, 那么仪式就失败了。
- brz 行走的总路程不能超过  $5 \times 10^3$  个单位, 即最多只能走这么多步。

还有需要补充的一点是, brz 可以在中途经过 Mandy 所在的位置而不必送出数字, 只要在仪式的最后送出即可。

brz 当然早就规划好了一条路线, 但是他现在想考考你, 你可以找出一条路线, 让 brz 送出他的"精确的浪漫"吗?

### Input

第一行包含一个正整数  $T$  ( $1 \leq T \leq 100$ ), 表示测试数据组数。

对于每组测试数据, 仅有一行输入, 该行包含六个整数  $S_x, S_y, T_x, T_y, x, y$  ( $0 \leq S_x, S_y, T_x, T_y \leq 2 \times 10^3$ ,  $1 \leq x, y \leq 10^9$ ), 其中  $(S_x, S_y)$  表示 brz 的初始位置,  $(T_x, T_y)$  表示 Mandy 的位置,  $x$  是 brz 手中初始的数字,  $y$  是 Mandy 期待的数字。

### Output

对于每组测试数据, 输出一行由 WASD 组成的字符串表示 brz 的行动路线。字符串从左到右表示 brz 依次进行的移动, W 表示向上移动一个单位, A 表示向左移动一个单位, S 表示向下移动一个单位, D 表示向右移动一个单位。

如果有多种路线方案, 输出任意一种即可。

Example

standard input	standard output
3 1 1 2 2 2 5 2 2 1 1 6 2 1 1 3 1 1 2	WD SA DDADAWDDSA

## Problem H. What is all you need?

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         1024 megabytes

自 2017 年 6 月, Transformer 架构的原始论文 "Attention is All You Need" 发表以来, 许多以 "X is All You Need" 为标题的文章先后涌现。

Siri 是一只爱学习的猫猫, 她在河北省赛的前一天晚上打算看一篇论文。她认为, 如果一篇论文的标题以 **is All You Need** 为结尾, 那么 **is All You Need** 前面的内容则是 Siri 要学习的内容, 否则 Siri 会因为这个东西不是她所需要的而不进行学习。

Siri 告诉你这篇论文的标题, 你需要告诉 Siri 她要不要进行学习。如果要学习的话, Siri 学习的内容是什么。由于 Siri 觉得, 处理空格与大小写是个非常麻烦的事情, 因此 Siri 告诉你论文标题时会将所有空格去掉, 并将所有的大写字母转为对应的小写字母。

经过这样的转换后, 你只需要判断论文标题是否以 **isallyouneed** 为结尾。如果是, 则告诉 Siri 其前面的内容即可。

### Input

一行一个字符串, 表示论文的标题。论文标题的长度  $n$  满足  $13 \leq n \leq 100$ 。

论文标题保证:

- 只由小写英文字母组成。
- 由于  $n \geq 13$ , 因此不会出现论文标题就是 **isallyouneed** 的情况。

### Output

如果 Siri 需要进行学习, 第一行输出 **Yes**; 否则输出 **No** 并无需进行后续输出。

如果 Siri 需要进行学习, 则需要在**第二行**输出 Siri 想要学习的内容。

### Examples

standard input	standard output
attentionisallyouneed	Yes attention
hbcpcisallyoneed	No

## Problem I. 感染

Input file:           standard input  
Output file:          standard output  
Time limit:          2 seconds  
Memory limit:        1024 megabytes

oql 在家里种了一棵龙眼树。这棵龙眼树是一个由  $n$  个节点  $n - 1$  条边组成的无向连通图。龙眼树上每个节点都有一串甜美多汁的龙眼。马上就到龙眼收获的季节了，oql 在某天的清晨前去查看龙眼树的情况。他发现了一个非常不幸的消息：他的龙眼树被害虫攻击了！有若干节点上的龙眼已经被害虫感染。他必须立即采取措施，以防治害虫感染更多的节点。但在此之前，他更好奇害虫在龙眼树上的传播方式。

具体地，害虫通过寄生虫卵的方式感染龙眼树上的节点。一个节点被感染当且仅当该节点上至少拥有一个虫卵。在第 0 天，害虫会选择在树上的一个节点寄生一个虫卵。每过一天，所有当前被感染的节点都会往它所有相邻的节点寄生一颗虫卵（无论相邻的节点是否已经被感染，两个节点相邻当且仅当它们间有一条边直接相连）。**请注意**，如果一个被感染节点拥有多颗虫卵，那么它每天内仍只会往相邻节点寄生一颗虫卵。

显然，害虫一开始选择寄生虫卵的节点将会影响后续虫卵的寄生顺序以及虫卵的总个数。oql 想知道哪些节点作为初始寄生虫卵的节点，满足第  $L = 2025^{5^{18}}$  天后整棵龙眼树的虫卵个数和是在所有情况中最大的。

具体来说，设  $f(i)$  ( $1 \leq i \leq n$ ) 表示初始时只在第  $i$  个节点有一颗虫卵，经过  $L$  天后整棵龙眼树的虫卵个数。oql 想要找到所有满足  $f(i) = \max_{1 \leq j \leq n} f(j)$  的节点  $i$ 。

### Input

第一行一个整数  $T$  ( $1 \leq T \leq 10^4$ )，表示数据组数。  
对于每组数据，第一行一个正整数  $n$  ( $1 \leq n \leq 5 \times 10^5$ )。  
下面  $n - 1$  行每行两个正整数  $u, v$  ( $1 \leq u, v \leq n$ )，表示树的一条边。  
数据保证对于  $T$  组数据，有  $\sum n \leq 5 \times 10^5$ 。

### Output

对于每组数据，输出两行。第一行一个正整数  $s$  ( $1 \leq s \leq n$ )，表示满足条件的节点总数；第二行  $s$  个正整数，按升序依次输出所有满足条件的节点编号。

### Example

standard input	standard output
2	2
2	1 2
1 2	2
4	1 2
1 2	
2 4	
1 3	

## Problem J. Generate 01 String

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         1024 megabytes

Siri 发现了一个神奇的标识符  $S$ ，可以进行两种操作：

- 1. 拍一下某个标识符  $S$ ，这个标识符将会变成  $0S1S$ 。
- 2. 拍两下某个标识符  $S$ ，这个标识符将会变成  $1S0S$ 。

最终在某个时刻，你可以选择让所有的标识符都消失，只保留字符串中的  $01$  字符。  
现在 Siri 想知道，能否有某种操作方案，得到给定的  $01$  串  $T$ 。

### Input

一行一个  $01$  串  $T$  ( $1 \leq |T| \leq 10^6$ )，含义如题。

### Output

如果存在操作方案，第一行输出  $n$  ( $1 \leq n \leq 10^6$ )，表示操作次数。  
接下来  $n$  行，每行为两个由空格分隔的数  $pos, op$ ，表示对当前字符串中的第  $pos$  个标识符进行第  $op$  种操作。  
否则输出  $-1$ 。

### Example

standard input	standard output
0011	2 1 1 1 1

### Note

样例解释：  
最初为  $S$ 。  
第一次操作  $1\ 1$ ，得到  $0S1S$ 。  
第二次操作  $1\ 1$ ，得到  $00S1S1S$ 。  
标识符  $S$  消失后即得到  $0011$ 。

Problem K. UNO!

Input file:           standard input  
 Output file:         standard output  
 Time limit:          1 second  
 Memory limit:       1024 megabytes

你说的对，但是"UNO！"是一款……后面忘了。  
 不过菜菜子为了让这个常玩的纸牌游戏更加有趣，自定义了一系列的规则：

1. 每次游戏从 1 号玩家开始，共  $n$  名玩家，他们按顺序顺时针坐在一张圆桌上进行游戏。**最开始按顺时针依次出牌**。第  $i$  号玩家手里初始有  $a_i$  张牌。
2. 纸牌分为 4 种牌，分别是普通牌（C）、禁止牌（S）、反转牌（R）和罚抽两张牌（D）。普通牌并没有效果；禁止牌可以让你的下一名玩家（简称下家）暂停出牌一次；反转牌可以使当前出牌的顺序翻转，即接下来轮到你的上家出牌；罚抽两张牌可以使你的下家的牌数 +2，并且暂停出牌一次。所有牌都没有任何出牌的限制，玩家可以随意出任意一种牌。
3. 如果一名玩家手里的牌打光，该玩家会**立马退出对局**。

菜菜子知道了这把游戏中的出牌结果，请你告诉他每位玩家最后剩下几张牌。

Input

第一行包含两个整数  $n, m$  ( $2 \leq n, m \leq 2 \times 10^5$ )，如题意所描述。  
 第二行包含  $n$  个正整数  $a_i$  ( $1 \leq a_i \leq 2 \times 10^5$ )。  
 第三行包含一个长度为  $m$  的字符串，表示这把游戏所有玩家的出牌结果。其中 C 为普通牌，S 为禁止牌，R 为反转牌，D 为罚抽两张牌。保证出牌顺序合法，即直到最后也有至少两位玩家有牌。

Output

输出包含  $n$  行，第  $i$  行为第  $i$  名玩家最终的手牌数量。

Example

standard input	standard output
3 6	3
3 2 3	0
SRDCCD	3

Note

首先一号玩家出禁止牌，二号玩家被禁止一回合，三号玩家出反转牌，使得出牌顺序反转，接下来轮到二号玩家出牌，打出 +2 牌使得一号玩家的手牌数量 +2，并使其禁止一回合。接下来三号、二号玩家各出一张普通牌，此时二号玩家手牌打完，一号玩家打出 +2 牌使三号玩家手牌数量 +2。最后三人的手牌数量为分别为 3 0 3 。

## Problem L. 通信拦截

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         1024 megabytes

网络强国，网信报国！  
让我们回到那个战火纷飞的年代，一起感受通信的魅力。

在一处前线的战斗中，敌军一共有  $n$  个阵营，编号从 1 到  $n$ ，狡猾的敌人使用电报从后方向前线传递信息。为了能将信息传达到位，第  $i$  个阵营会向  $[i + 1, \min(i + r_i, n)]$  区间内的所有阵营发送一封电报。现在，你是一位通信兵，拥有一个拦截器，你可以潜伏进敌军的阵线，将它放置在敌军的某个阵营中。若有一封电报从阵营  $p$  发送到阵营  $q$ ，拦截器位于  $x$ ，并且  $p \leq x \leq q$ ，那么该电报就会被拦截，拦截价值为  $\min(x - p, q - x)$ 。请问对于每个阵营  $i$ ，在该位置放置拦截器的拦截价值之和为多少？

### Input

第一行包含一个整数  $n$  ( $2 \leq n \leq 2 \cdot 10^5$ )，如题意所描述。  
第二行包含  $n$  个正整数  $r_i$  ( $1 \leq r_i \leq n$ )，每个数都被单个空格隔开，如题意所描述。

### Output

一共  $n$  行，第  $i$  行在第  $i$  个阵营放置拦截器的拦截价值之和。

### Example

standard input	standard output
6	0
4 4 3 2 2 3	3
	6
	6
	3
	0

### Note

对于样例中第 2 个军营，放置拦截器的拦截价值之和为  $2 + 1 = 3$ 。其余不再赘述。

## Problem M. 第九届河北省大学生程序设计竞赛

Input file:           standard input  
Output file:          standard output  
Time limit:          1 second  
Memory limit:        1024 megabytes

oql 和他的队友们又需要为河北省大学生程序设计竞赛（以下简称河北省赛）准备题目了。一套合理的 CCPC 赛题一般由 10 ~ 13 道题目组成。这一次，他们准备了  $n$  道题目，想从中选择若干道作为河北省赛的正式赛题。

现在，oql 提前得知了河北省赛的所有参赛队伍，并通过使用魔法得到了每支队伍是否会做某一道题的数据。假设每支队伍都能在河北省赛中发挥出色（当然此刻的你也一定能发挥出色），所有队伍都能在赛场上全部通过他们会做的题。当然，如果一道题某支队伍不会，那么他们也不会通过该题。

众所周知，CCPC 按照每支队伍的通过题数以及罚时进行排序。由于本题只关注通过题数，罚时可以忽略不计。oql 提前得知了本次河北省赛金牌、银牌、铜牌线为  $rk_g, rk_s, rk_b$ ，即金牌、银牌、铜牌的最后一名分别是第  $rk_g, rk_s, rk_b$  名（冠军为第一名）。而为了给大家更多的比赛体验，oql 想能不能在  $n$  个题目里选择 10 ~ 13 道作为正式题目，使得金牌、银牌、铜牌的最后一名的通过题数分别是  $p_g, p_s, p_b$ 。如果可以的话，oql 想让你告诉他一个合法的选题方案。

### Input

第一行两个整数  $n, m$  ( $10 \leq n \leq 18, 3 \leq m \leq 200$ )，分别表示所准备的题目总数和参赛队伍数。

接下来  $m$  行，第  $i$  行一个长度为  $n$  的字符串  $a_{i,1}a_{i,2}\cdots a_{i,n}$ ，表示第  $i$  支队伍所有题目的通过情况。如果  $a_{i,j}$  为 1，则表示该队伍可以通过第  $j$  道题目；否则  $a_{i,j}$  为 0，表示不通过。

接下来一行三个整数  $rk_g, rk_s, rk_b$ , ( $1 \leq rk_g < rk_s < rk_b \leq m$ )，依次表示金、银、铜牌最后一名的排名。

最后一行三个整数  $p_g, p_s, p_b$ , ( $1 \leq p_b \leq p_s \leq p_g \leq \min(n, 13)$ )，依次表示 oql 想要的金、银、铜牌最后一名的通过题数。

### Output

如果不存在合法的方案，输出 -1 并无需进行后续输出。

否则，第一行输出一个整数  $k$ ，表示一个合法选题方案可以选择多少道题。

第二行输出  $k$  个整数，表示一个合法方案选择的题目。题目标号从 1 开始。你的输出方案需要满足题目编号是 1 ~  $n$  的正整数且不重复出现。

如果有多种合法方案，输出任意一种即可。

### Example

standard input	standard output
13 3 1111111111111 1111111111110 11111111110000 1 2 3 12 11 8	12 1 2 3 4 5 6 7 8 10 11 12 13

### Note

第一支队伍所有题都会做，肯定可以成为第一名。要求第一名通过 12 道题，因此最终合法方案只可能是 12 题。

只考虑第一名，可以删除标号为 1 ~ 13 的任意一题。

只考虑第二名，可以删除标号为 1 ~ 12 的任意一题。



只考虑第三名，可以删除标号为 1 ~ 9 的任意一题。

综上，一个合法的方案是只删除第 9 题，保留其他所有题目。