

Problem A. 备用账号

时间限制: 2 seconds
空间限制: 512 megabytes

在遥远的猫猫国度，有一个著名的喵语写作平台 Catforces，猫猫们常常在上面参加比赛。最近，Catforces 的维护者三花（ミケ）苦恼于注册小号参赛的问题。但他发现，许多猫的大号和小号在名称上会有一定关联。

具体来说，我们认为如下字符是相似的：

- a、A 与 @
- o、0 与 0
- 对应的大小写字母，如 c 和 C

如果两个长度相等的用户名的对应位置字符均相同或相似，我们就认为这一对用户名构成大小号关系。例如 coder 和 c0dEr 构成大小号关系，coder 和 code 则不构成。

给出平台上 n 位用户的用户名，请你判断有多少对用户名是大小号关系。

Input

第一行包含一个整数 n ($2 \leq n \leq 10^5$)，代表用户名的数量。
之后的 n 行，每行包含一个字符串代表用户名，其字符只包含拉丁字母、数字和 @。
保证用户名的总长度不超过 5×10^5 。

Output

输出一个整数，代表构成大小号关系的用户名的对数。

Examples

standard input	standard output
5 cat atc c@t at at	2
3 coder c0der c0deR	3

Problem B. 进制变换

时间限制: 1 second
空间限制: 512 megabytes

小 F 最近刚学完进制，但是小 F 不喜欢 0 次幂，于是决定创造一种基于进制但又不同于进制的变换方式。

对于一个 n 位的十进制数字 $\overline{x_{n-1}x_{n-2}\cdots x_0}$ ，选取一个正整数参数 k 作为进制，则：

$$f(n) = \sum_{i=0}^{n-1} x_i \times k^{i+1}$$

小 F 对这种变换非常满意，便询问小 M 如何通过重复应用这种变换把 a 变为 b ，每次变换可以重新选择 k ，要求变换次数不超过 10 次，且每次得到的数字小于等于 3×10^{18} 。

Input

第一行一个整数 T ($1 \leq T \leq 10^5$)，表示一共有 T 组数据。

对于每组数据：

- 仅一行，包含两个整数 a 和 b ($1 \leq a, b \leq 10^9, a \neq b$)。

Output

对于每组数据：

- 第一行输出操作次数 m ($1 \leq m \leq 10$)。
- 接下来 m 行，每行输出两个整数，分别表示第 i 次操作选择的 k_i ($1 \leq k_i \leq 3 \times 10^{18}$) 和第 i 次操作后得到的数字 x_i ($1 \leq x_i \leq 3 \times 10^{18}$)。

Example

standard input	standard output
2	2
8 32	7 56
1 10	2 32
	1
	10 10

Note

在样例一中，第一次变换选择 $k_1 = 7$ ，变换后的答案 $x_1 = 8 \times 7^1 = 56$ ；第二次变换选择 $k_2 = 2$ ，变换后的答案 $x_2 = 6 \times 2^1 + 5 \times 2^2 = 12 + 20 = 32$ 。

Problem C. 或非

时间限制: 1 second
空间限制: 512 megabytes

对于非负整数 x ，我们定义 $\text{pos}(x, i)$ 表示 x 的第 i 个二进制位（其中 $i \geq 0$ ）。即：

$$\text{pos}(x, i) = \left\lfloor \frac{x}{2^i} \right\rfloor - 2 \left\lfloor \frac{x}{2^{i+1}} \right\rfloor$$

对于两个非负整数 x 和 y ($0 \leq x, y < 2^{30}$)，我们定义它们的**按位或非** $\text{nor}(x, y)$ 为：

$$\text{nor}(x, y) = \sum_{i=0}^{29} [\text{pos}(x, i) = 0 \wedge \text{pos}(y, i) = 0] 2^i$$

其中 $[P]$ 表示艾弗森括号，当命题 P 为真时， $[P]$ 的值为 1，否则为 0。 $\wedge$ 符号表示逻辑与运算。

也就是说，当且仅当 x 和 y 的第 i 位都为 0 的时候， $\text{nor}(x, y)$ 的第 i 位为 1；其余情况下， $\text{nor}(x, y)$ 的第 i 位都为 0。

对于两个非负整数 x 和 y ，我们定义它们的**按位异或** $x \oplus y$ 为：

$$x \oplus y = \sum_{i=0}^{\infty} [\text{pos}(x, i) \neq \text{pos}(y, i)] 2^i$$

给定一个 n 个节点的无向完全图，节点编号为 0 到 $n-1$ 。对于每一对满足 $0 \leq u < v < n$ 的节点 u 和 v ， u 和 v 之间有一条边权为 $\text{nor}(u, v)$ 的边。

请你找出该图的一个**生成树**，使得生成树中，所有边的边权的**按位异或**结果尽可能的小。

Input

第一行包含一个整数 T ($1 \leq T \leq 10^5$)，表示一共有 T 组测试数据。

对于每组测试数据：

- 仅一行，包含一个整数 n ($2 \leq n \leq 2 \times 10^5$)，表示节点数量。

保证测试数据的 n 的总和不超过 2×10^5 。

Output

对于每组测试数据：

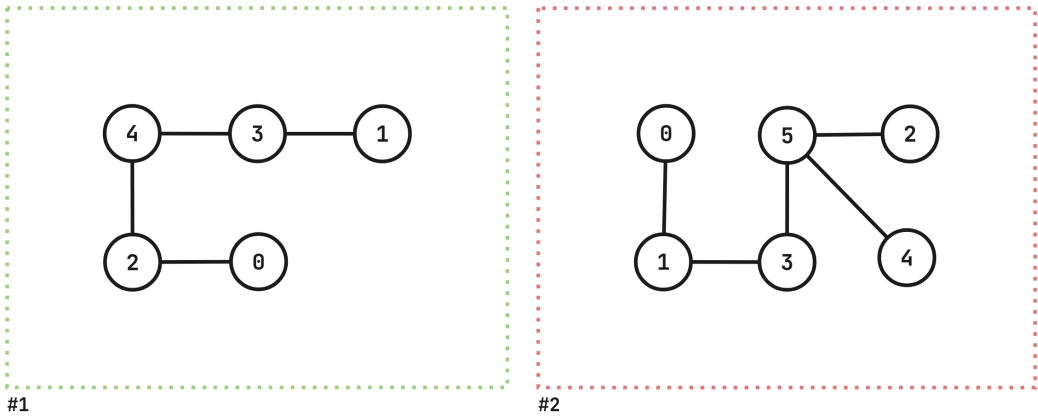
- 输出 $n-1$ 行，第 i 行包含两个整数 u_i 和 v_i ($0 \leq u_i, v_i < n$ 且 $u_i \neq v_i$)，表示你所找到的生成树的第 i 条边连接 u_i 和 v_i 两个节点。
- 如果有多个满足要求的生成树，你可以输出其中任意一个。

Example

standard input	standard output
2	0 2
5	3 1
6	3 4
	2 4
	0 1
	1 3
	2 5
	3 5
	4 5

Note

样例的两组数据中，生成树的形态如下图所示。



注意：样例输出中的空行是为了让两组数据的输出更易区分而添加的，你的程序不必输出该空行。

Problem D. Fever Dash

时间限制: 2 seconds
空间限制: 512 megabytes

小 F 很喜欢玩《Fever Dash》这款音乐游戏。

一个关卡总共有 n 枚音符，第 i 枚音符有落下时刻 t_i 、分数 p_i 、Fever 值 f_i 三个属性。

游戏的 Fever 机制如下：

- 击打音符会累加其 Fever 值到 Fever 槽中，槽的上限为 L 。一旦在某个时刻，Fever 槽的值达到 L ，从下一个时刻开始的每个时刻都可以选择开启 Fever。
- 开启 Fever 后，Fever 槽被清空，并进入持续 k 个时刻的 Fever 状态。也就是说，如果在 i 时刻开启 Fever 状态，则 $[i, i + k - 1]$ 时刻都将处在 Fever 状态中。在该状态下，所有击打的音符分数翻倍，但无法获得 Fever 值。Fever 状态结束后才能重新开始积累。

具体来讲，每一时刻内，以下事件依次发生：

- 如果 Fever 值达到了上限 L ，小 F 可以选择是否开启 Fever 状态。开启 Fever 后，Fever 槽被清空。
- 如果处在 Fever 状态中，并且当前时刻 x 与开启 Fever 的时刻 y 满足 $x - y \geq k$ ，则 Fever 状态结束。
- 如果该时刻有音符落下，小 F 将会击打该音符：记这枚音符的编号为 i ，若处于 Fever 状态，获得 $2 \times p_i$ 分，不累加 Fever 值；否则，获得 p_i 分，并将 f_i 累加到 Fever 槽。

游戏开始时不在 Fever 状态中，且 Fever 槽为 0。Fever 可以重复开启任意次，开启后无法手动关闭。注意在最后一个音符落下后 Fever 状态可以还没有结束。

给出 n 个音符落下的时刻 t_i ，分数 p_i 和 Fever 值 f_i ，小 F 想知道，如果他以最优方案开启 Fever 他能够获得多少分，你能帮帮他吗？

（题意与任何现实游戏的机制无关）

Input

第一行包含 1 个整数 T ($1 \leq T \leq 1000$)，表示数据组数。

每组数据的第一行为 n, L, k ($1 \leq n \leq 5 \times 10^5, 1 \leq L, k \leq 10^9$) 三个整数，分别是音符个数、Fever 槽上限、Fever 持续时长。

下一行包含 n 个整数 t_i ($1 \leq t_i \leq 10^9$)，代表第 i 个音符落下的时刻。

下一行包含 n 个整数 p_i ($1 \leq p_i \leq 10^9$)，代表击打第 i 个音符获得的分数。

下一行包含 n 个整数 f_i ($1 \leq f_i \leq 10^9$)，代表击打第 i 个音符累积的 Fever 值。

数据保证：

- 多组数据的 n 之和不超过 5×10^5 。
- t_i 单调递增，即对于任何 $1 \leq i < n$, $t_i < t_{i+1}$ 。

Output

对于每组数据输出一个整数，表示获得的最大分值。

Example

standard input	standard output
4	16
3 5 5	58
1 2 7	66
4 4 4	80
5 10 7	
7 3 2	
1 2 3 4 5 6 7	
2 5 4 1 1 9 9	
3 2 2 2 2 1 1	
8 2 5	
2 4 5 6 7 8 9 10	
6 4 8 3 1 2 7 6	
6 2 9 4 8 5 5 4	
7 1 6	
1 2 3 5 6 8 10	
9 5 9 7 3 3 10	
6 10 8 9 2 3 3	

Note

（方便起见，以下假设时刻的单位是秒）

对于第一组样例，在第 1 秒获得 4 分并获得 5 Fever 值，达到 Fever 上限；在第 2 秒开启 Fever，获得 8 分，不获得 Fever 值；在第 7 秒时 Fever 已经结束，获得 4 分和 7 Fever 值，共计 16 分。

对于第二组样例，在第 [2,3] 秒和第 [6,7] 秒处于 Fever 状态最优。

Problem E. 简单的数据结构题

时间限制: 5 seconds
空间限制: 512 megabytes

你需要维护一个非负整数序列 $[a_1, a_2, a_3, \dots, a_n]$ ($0 \leq a_i < 16$)，支持以下三种操作：

- 1 $l\ r$: 将子区间 $[a_l, a_{l+1}, a_{l+2}, \dots, a_r]$ 内的每个 a_i 变为 $(a_i + 1 - 16 \cdot \lfloor \frac{a_i+1}{16} \rfloor)$ 。
- 2 $l\ r$: 将子区间 $[a_l, a_{l+1}, a_{l+2}, \dots, a_r]$ 内的每个 a_i 变为 $(2 \cdot a_i - 15 \cdot \lfloor \frac{a_i}{8} \rfloor)$ 。
- 3 $l\ r$: 我们令 cnt_x 表示子区间 $[a_l, a_{l+1}, a_{l+2}, \dots, a_r]$ 内数字 x 出现的次数。计算并输出 $\bigoplus_{i=0}^{15} cnt_i$ ，即 $cnt_0, cnt_1, \dots, cnt_{15}$ 按位异或的结果。

可以证明，在以上三种操作下，任何时刻，序列中的元素 a_i 都满足 $0 \leq a_i < 16$ 。

Input

第一行包含两个整数 n 和 q ($1 \leq n, q \leq 5 \cdot 10^5$)，分别表示序列长度与操作次数。
第二行包含 n 个整数 $a_1, a_2, a_3, \dots, a_n$ ($0 \leq a_i < 16$)，表示序列的初始值。
接下来的 q 行，第 i 行包含 ty_i, l_i, r_i 三个整数 ($1 \leq ty_i \leq 3$ 且 $1 \leq l_i \leq r_i \leq n$)，表示第 i 次操作的类型，与操作涉及的区间。

Output

对于每个类型为 3 的操作，输出一行，包含一个整数，表示答案。

Example

standard input	standard output
6 10	1
1 1 4 5 1 4	0
1 1 6	1
1 4 5	2
2 2 4	
3 1 1	
2 2 3	
1 4 4	
3 2 5	
3 1 5	
1 1 4	
3 1 6	

Problem F. 乱序法杖

时间限制: 1 second
空间限制: 512 megabytes

小 F 有一根法杖，其中装填有 $x + y$ 个法术。
初始时刻，有 m 只小羊排成一排，第 i 只小羊的血量为 a_i 。 a_i 是一个整数。
这 $x + y$ 个法术分别编号为 1 到 $x + y$ ，并且分成以下两类：
编号为 1 到 x 的是「治疗法术」，施放时产生如下效果：

- 对于所有血量 > 0 的小羊，将它们血量 $+1$ 。
- 对于所有血量 < 0 的小羊，将它们血量 -1 。

编号为 $x + 1$ 到 $x + y$ 的是「伤害法术」，施放时产生如下效果：

- 对于所有血量 > 0 的小羊，将它们血量 -1 。
- 对于所有血量 < 0 的小羊，将它们血量 $+1$ 。

然而小 F 的这根法杖是乱序的。也就是说，当使用这根法杖的时候，会随机产生一个长度为 $x + y$ 的排列¹（从所有长度为 $x + y$ 的排列中等概率选取），记作 $[P_1, P_2, P_3, \dots, P_{x+y}]$ 。随后，依次施放编号为 $P_1, P_2, \dots, P_{x+y}$ 的法术。

小 F 想知道，在他使用这根法杖后，每只小羊的血量的期望是多少？
请你计算并输出答案对 998 244 353 取模的结果。

具体而言，若答案为最简分数 $\frac{p}{q}$ （ p 与 q 互质且 q 不被 998 244 353 整除），请输出满足 $0 \leq x < 998\,244\,353$ 且 $x \cdot q \equiv p \pmod{998\,244\,353}$ 的最小整数 x 。

Input

第一行包含三个整数 x, y, m （ $0 \leq x, y \leq 10^5$ 且 $1 \leq m \leq 3 \times 10^5$ ）。
第二行包含 m 个整数 $a_1, a_2, \dots, a_m$ （ $-10^7 \leq a_i \leq 10^7$ ），表示小羊的初始血量。

Output

输出一行，包含 m 个整数，分别表示每只小羊的血量的期望，对 998 244 353 取模后的结果。

Examples

standard input	
2 3 11	
-5 -4 -3 -2 -1 0 1 2 3 4 5	
standard output	
998244349 998244350 598946610 499122176 0 0 0 499122177 399297743 3 4	
standard input	
0 0 1	
23333	
standard output	
23333	

¹如果一个长度为 k 的整数序列 $[c_1, c_2, \dots, c_k]$ 满足：1 到 k 中的每个整数（包括 1 和 k ）都恰好在 c 中出现一次，那么我们称 c 是一个长度为 k 的排列。例如 $[1, 2, 3]$ 、 $[1, 5, 3, 2, 4]$ 都是排列，而 $[1, 1, 4]$ 不是。

standard input
233 123 4 -123 1145 81 0
standard output
625392963 1255 783909838 0

Problem G. Good Number

时间限制: 2 seconds
空间限制: 512 megabytes

对于非负整数 x 和 i ($0 \leq x, 0 \leq i \leq 9$)，我们定义 $F(x, i)$ 表示 x 的十进制表示下数字 i 的出现次数。

对于非负整数 x ($0 \leq x$)，我们定义它的美观程度 $G(x)$ 为：

$$G(x) = \sum_{\substack{0 \leq i \leq 9 \\ F(x, i) > 0}} F(x, i)^{F(x, i)}$$

例如， $x = 1145$ 时，由于 $F(x, 1) = 2, F(x, 4) = 1, F(x, 5) = 1$ ，其余 $F(x, i)$ 均为 0，因此 $G(x) = 2^2 + 1^1 + 1^1 = 6$ 。

给定三个正整数 L, R, K ，请你计算一下：

- 在 $[L, R]$ 范围内的所有整数中，有多少个数的美观程度 $G(x) \geq K$ ？
- 这些满足条件的数的美观程度的总和是多少？由于结果可能很大，请输出答案对 998 244 353 取模后的结果。

注意只有第二个问题的答案需要对 998 244 353 取模。

Input

仅一行，包含三个整数 L, R, K ($1 \leq L \leq R \leq 10^{15}$ 且 $1 \leq K \leq 10^{18}$)。

Output

输出一行，包含两个整数，分别表示两个问题的答案。

Examples

standard input
1 100 1
standard output
100 212
standard input
1000000000000000 1000000000000000 1
standard output
1 567381139
standard input
1 1000000000000000 114514
standard output
2968291384813 671467889

Problem H. 快速排列置换

时间限制: 2 seconds
空间限制: 512 megabytes

对于一个长度为 n 的整数序列 $[a_1, a_2, a_3, \dots, a_n]$ ，如果其中 1 到 n 之间的每个数刚好出现一次，那么我们称它是一个长度为 n 的排列。

例如， $[1, 3, 2, 4]$ 、 $[1, 6, 2, 3, 4, 5]$ 、 $[1]$ 都是排列，而 $[1, 1, 4]$ 不是。

对于两个长度同样为 n 的排列 a 和 b ，我们定义 b 对 a 的排列置换 $F(a, b)$ 为：

$$F(a, b) = [a_{b_1}, a_{b_2}, a_{b_3}, \dots, a_{b_n}]$$

对于长度为 n 的排列 a 以及非负整数 x ，我们定义 a 的 x 次自身置换 $G(a, x)$ 为：

$$G(a, x) = \begin{cases} a & \text{if } x = 0 \\ F(G(a, x-1), G(a, x-1)) & \text{else} \end{cases}$$

给定一个长度为 n 的排列 a 以及非负整数 x ，请你计算并输出 a 的 x 次自身置换 $G(a, x)$ 。

Input

第一行包含一个整数 n ($1 \leq n \leq 10^5$)。
第二行包含 n 个整数 $a_1, a_2, a_3, \dots, a_n$ ，用空格分隔，表示排列 a 。
第三行包含一个整数 x ($0 \leq x \leq 10^{10^5}$)。

Output

输出一行，包含 n 个整数，用空格分隔，表示计算结果。

Examples

standard input	standard output
5 2 3 1 5 4 3	3 1 2 4 5
12 9 7 8 1 2 3 12 6 11 5 4 10 1145141919810	1 5 8 4 10 3 2 6 9 12 11 7

Problem I. Emotional Flutter

时间限制: 5 seconds
空间限制: 512 megabytes

给定 3 个二维平面中的简单凸多边形² P_A 、 P_B 、 P_C 。

除此之外，还有三个质点 A 、 B 、 C 。它们的质量分别是 m_A 、 m_B 、 m_C 。

接下来你需要回答 q 次查询。第 i 次查询将会给你一个坐标 (x_i, y_i) ，你需要回答，是否存在一种方案，将 A 、 B 、 C 分别放置在 P_A 、 P_B 、 P_C 的内部及其边界上的某个坐标位置，使得 A 、 B 、 C 的质心 G 恰好位于坐标 (x_i, y_i) 处。

质心的计算公式如下：

若 A 、 B 、 C 分别位于 (x_A, y_A) 、 (x_B, y_B) 、 (x_C, y_C) ，则其质心 G 位于：

$$G = \left(\frac{m_A x_A + m_B x_B + m_C x_C}{m_A + m_B + m_C}, \frac{m_A y_A + m_B y_B + m_C y_C}{m_A + m_B + m_C} \right)$$

本题强制在线。具体来说，第 i 次查询的输入为一个加密的坐标 (x'_i, y'_i) ，我们用 t 表示之前查询中 YES 的个数，那么解密后的实际查询坐标 (x_i, y_i) 为：

$$x_i = \begin{cases} x'_i + t, & x'_i < 0 \\ x'_i \oplus t, & x'_i \geq 0 \end{cases} \quad y_i = \begin{cases} y'_i + t, & y'_i < 0 \\ y'_i \oplus t, & y'_i \geq 0 \end{cases}$$

其中 $\oplus$ 符号表示按位异或。

Input

第一行包含 3 个整数 m_1, m_2, m_3 ($1 \leq m_1, m_2, m_3 \leq 10$)，表示三个质点的质量。

接下来 3 个部分代表 3 个凸多边形，其中第 i 个部分的输入为：

- 第一行一个整数 n_i ($3 \leq n_i \leq 5 \cdot 10^5$)，表示第 i 个凸多边形共有 n_i 个顶点。
- 接下来 n_i 行，每行两个整数 x, y ($-10^8 \leq x, y \leq 10^8$)，表示凸多边形的各个顶点的坐标，按照逆时针方向给出。

接下来一行包含一个整数 q ($1 \leq q \leq 5 \cdot 10^5$)，表示查询个数。

接下来 q 行，第 i 行包含两个整数 x'_i, y'_i ($-10^8 \leq x'_i, y'_i \leq 10^8$)，代表一次加密查询。

保证输入的 3 个凸多边形都是凸多边形。保证本题 validator 所用的判凸包代码能够通过 Convex Checker (QOJ 7730)。

Output

对于每次查询，如果存在方案，则输出一行 YES，否则输出一行 NO。

输出的大小写是无关紧要的。也就是说，Yes、yEs 也会被认为是 YES，nO 也会被认为是 NO。

²简单多边形是平面中由线段构成的闭合曲线，这些线段首尾相连，除了因连接而共用的线段端点，任何两个线段都不能彼此相交。在此基础上，如果一个简单多边形还满足每个内角都严格小于 180° ，则我们称它为简单凸多边形。

Example

standard input	standard output
1 1 1	YES
5	NO
0 0	NO
-2 11	YES
-6 7	YES
-6 6	
-5 3	
5	
0 0	
4 3	
-1 7	
-4 5	
-4 4	
3	
0 0	
2 -2	
-1 3	
5	
-1 5	
-4 0	
3 4	
1 0	
3 -2	

Problem J. Eternal Feather II

时间限制: 1 second
空间限制: 512 megabytes

翼なんてくてかまわない 即使没有翅膀也没关系
だから届けて ふたりへ 因而我们两个人能够到达
永（とわ）にわらぬ あの空 永恒不变的 那片天空
在音羽的教堂见过优子最后一面之后，夕毅然转头离开音羽教堂。一曲 *ever forever* 响起，ef 的故事就此落幕。夕不愿忘记与优子的回忆，但他仍然需要一个人坚强的活下去。
圣诞节这天，他发现了一个长度为 n 的，由 $\{y, u, k, o\}$ 为字符集构成的字符串。他要求出这样的串的个数，使得这个串不包含一个子串，这个子串经过重新排列之后可以构成 *yuuko*。
夕是建筑领域专家，但他并不会算法竞赛。你能帮助他吗？
由于答案可能很大，你只需要输出答案对 998 244 353 取模的结果。

Input

一个数 n ($1 \leq n \leq 10^{18}$)，代表字符串长度。

Output

一个数，表示答案对 998 244 353 取模的结果。

Examples

standard input	standard output
2	16
5	964