

A

Ad-hoc Newbie

时间限制: 2 seconds

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

Yuki 给出了一个长度为 n 的正整数序列 f_1, \dots, f_n ，保证对每个 i 都有 $1 \leq f_i \leq i$ 。她想要请你构造一个 n 阶方阵 A ，满足：

- 对任意 $1 \leq i, j \leq n$ ，都有 $0 \leq A_{i,j} \leq n$ ；
- 对任意 $1 \leq i \leq n$ ， $\text{mex}(A_{i,1}, A_{i,2}, \dots, A_{i,n}) = \text{mex}(A_{1,i}, A_{2,i}, \dots, A_{n,i}) = f_i$ ¹。

可以证明，对任意合法的 f_1, \dots, f_n ，一定有解。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 2 \cdot 10^4$)，表示测试数据组数。每组测试数据的格式如下。

第一行，一个整数 n ($1 \leq n \leq 1\,414$)，表示给定的序列长度。

第二行， n 个整数 f_1, \dots, f_n ($1 \leq f_i \leq i$)，描述给定的序列。

保证所有测试数据中， n^2 的和不超过 $2 \cdot 10^6$ 。

输出格式

对于每组测试数据，输出共 n 行，第 i 行包含 $[0, n]$ 内的 n 个非负整数 $A_{i,1}, A_{i,2}, \dots, A_{i,n}$ 。

样例

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

¹序列 b_1, \dots, b_m 的 mex 值为最小的非负整数 x 满足 x 没有在序列 b 中出现。

提示

对于第一组数据, $f = [1, 1, 2]$, 一个合法的方阵为:

$$A = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

对于第一行, $\text{mex}([0, 2, 0]) = f_1 = 1$, 这是因为 0 在 $[0, 2, 0]$ 内出现, 而 1 没有, 因此 1 为最小的未出现的非负整数; 对于第一列, 同理地, $\text{mex}([0, 0, 0]) = f_1 = 1$ 。容易验证该矩阵也满足其它限制。

B Bitwise Puzzle

时间限制: 1 second 输入文件: standard input
空间限制: 512 megabytes 输出文件: standard output

Yuki 给了你三个非负整数 a 、 b 和 c 。你可以执行以下操作，最多进行 $k = 64$ 次：

1. $a \leftarrow a \cdot 2$;
2. $b \leftarrow \lfloor \frac{b}{2} \rfloor$;
3. $a \leftarrow a \oplus b$ ，其中 \oplus 指按位异或；
4. $b \leftarrow b \oplus a$ 。

请在不超过 k 次操作的情况下使 $a = b = c$ ，或报告没有合法方案。可以证明，在题目的约束条件下，若存在合法方案，一定存在不超过 $k = 64$ 次的合法方案。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 10^4$)，表示测试数据组数。每组测试数据的格式如下。

输入的第一行，也是唯一一行，包含三个整数 a 、 b 和 c ($0 \leq a, b, c < 2^{31}$)。

输出格式

对于每组数据：

- 如果你认为无解，仅需输出一行一个整数 -1 ；
- 否则，请输出两行。第一行包含一个整数 p ($0 \leq p \leq k$)，描述你进行的操作次数；第二行包含 p 个整数，每个整数在 $[1, 4]$ 之间，依次表示你进行的操作对应的序号。

样例

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

提示

对于第一组数据，初始 $a = 3$ ， $b = 5$ ， $c = 6$ ，进行第一次操作后 $a = 3$ ， $b = 6$ ， $c = 6$ ；进行第二次操作后 $a = b = c = 6$ ，符合要求。

对于第二组数据，无论如何进行操作，都有 $a = b = 0$ ，无法满足 $a = b = c$ 。

C Capella

时间限制: 3 seconds 输入文件: standard input
空间限制: 256 megabytes 输出文件: standard output

称一个仅包含小写英文字母的字符串 S 与 **Capella** 类似，当且仅当其满足以下所有条件：

- 在 S 中出现了奇数次的字母有奇数个；
- 在 S 中出现了非零的偶数次的字母有偶数个。

可以验证字符串 “capella” 与 Capella 类似；但字符串 “arcaea” 不与 Capella 类似，因为出现奇数次的字母有 4 个；“lowiro” 也不与 Capella 类似，因为出现非零的偶数次的字母有 1 个，出现奇数次的字母有 4 个。

现在 Yuki 有一个长度为 n 的仅由小写英文字母组成的字符串 S 。Yuki 对字符串 S 进行了 q 次操作：第 i 次操作给定 p_i ($1 \leq p_i \leq n$) 与一个小写英文字母 c_i ，将 S 的第 p_i 个位置上的字符 S_{p_i} 修改为 c_i 。你需要在所有 q 次操作执行前以及每次操作执行后，求出 S 的所有连续子串²中最长的与 Capella 类似的子串的长度。

输入格式

第一行，两个整数 n 和 q ($1 \leq n \leq 2 \cdot 10^5$, $1 \leq q \leq 2 \cdot 10^5$)。

第二行，一个长度为 n 的仅包含小写英文字母的字符串 S 。

接下来 q 行，每行包含一个整数 p_i ($1 \leq p_i \leq n$) 与一个小写英文字母 c_i ，表示一次操作。

输出格式

输出包含 $q+1$ 行。每行包含一个整数，依次表示所有 q 次操作执行前以及每次操作执行后， S 的所有连续子串中最长的与 Capella 类似的子串的长度。

样例

standard input	standard output
6 4	1
ababab	5
3 c	5
6 a	5
4 c	1
2 a	

提示

对于所有 q 次操作执行前以及每次操作执行后的情形：

²字符串 S' 是 S 的连续子串，当且仅当 S' 可由 S 删去一段可为空的前缀与后缀得到。

- 所有操作执行前，字符串 S 的初始值为 “ababab”，最长的类似 Capella 的子串为 “a” 或者 “b”，长度为 1。
- 第一次操作后，字符串 S 变为 “abcbab”，最长的类似 Capella 的子串为 “abcba” 或者 “bcbab”，长度为 5。
- 第二次操作后，字符串 S 变为 “abcbaa”，最长的类似 Capella 的子串为 “abcba” 或者 “bcbaa”，长度为 5。
- 第三次操作后，字符串 S 变为 “abccaa”，最长的类似 Capella 的子串为 “abcca” 或者 “bccaa”，长度为 5。
- 第四次操作后，字符串 S 变为 “aaccaa”，最长的类似 Capella 的子串为 “a” 或者 “c”，长度为 1。

D Distant Control

时间限制: 1 second 输入文件: standard input
空间限制: 512 megabytes 输出文件: standard output

你有 n 个机器人朋友，他们排成一排，从左至右的标号依次为 $1, 2, \dots, n$ 。初始时，有些机器人是关机的，而有些机器人是开机的。

你的手机能够遥控机器人的开机与关机，可惜它并非总是那么灵敏。具体地，有一个常数 $1 \leq a \leq n-1$ ，你仅能用你的手机做如下的两种操作：

- 选择标号连续的 a 个 均处于开机状态的 机器人，并将其全部设置为关机。
- 选择标号连续的 $a+1$ 个 均处于关机状态的 机器人，并将其全部设置为开机。

你希望求出，经过若干（可能为 0）次操作后，处于开机状态的机器人个数最大值。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 4 \cdot 10^4$)，表示测试数据组数。每组测试数据的格式如下。

第一行，两个整数 n 和 a ($2 \leq n \leq 2 \cdot 10^5$, $1 \leq a \leq n-1$)。

接下来一行，一个长度为 n 仅包含 01 的字符串 s ，描述初始时机器人的状态。具体地，第 i 个机器人初始处于开机状态，当且仅当 $s_i = 1$ 。

保证所有测试数据中， n 的和不超过 $4 \cdot 10^5$ 。

输出格式

对于每组数据，输出仅一行一个整数，表示处于开机状态的机器人个数最大值。

样例

standard input	standard output
4	3
3 1	2
001	5
4 3	5
0101	
5 2	
10110	
10 4	
1011010001	

提示

对于第一组数据，可以仅一次操作使得所有机器人均开机：选择标号为 1, 2 的机器人，此时他们均处于关机状态，将他们开机。

对于第二组数据，无法进行任何操作，所以答案是初始时开机状态的机器人个数 2。

E

Equal

时间限制: 2 seconds

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

Yuki 给了你一个长度为 n 的正整数序列 a_1, \dots, a_n , 你可以做以下两种操作任意次:

- 选择正整数 i, j, d 满足 $1 \leq i < j \leq n$ 且 $d \mid a_i, d \mid a_j$, 然后将 $a_i \leftarrow \frac{a_i}{d}, a_j \leftarrow \frac{a_j}{d}$;
- 选择正整数 i, j, d 满足 $1 \leq i < j \leq n$, 然后将 $a_i \leftarrow a_i \cdot d, a_j \leftarrow a_j \cdot d$ 。

判断通过若干次操作后, 能否使 $a_1 = a_2 = \dots = a_n$ 。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 10^5$), 表示测试数据组数。每组测试数据的格式如下。

- 第一行, 一个整数 n ($1 \leq n \leq 10^6$), 表示给定的序列长度。
- 第二行, n 个整数 a_1, \dots, a_n ($1 \leq a_i \leq 5 \cdot 10^6$), 描述初始的序列。
- 保证所有测试数据中, n 的和不超过 $2 \cdot 10^6$ 。

输出格式

对于每组测试数据: 输出一行一个字符串表示答案。你应当输出 “YES”, 当且仅当你认为能够通过若干次操作后, 使得序列 a 中的元素均相等; 否则, 输出 “NO”。

你可以以任意大小写形式输出答案。例如, “yEs”、“yes”、“Yes” 和 “YES” 都被认作是肯定的回答。

样例

standard input	standard output
6	YES
1	NO
6	YES
2	NO
2 4	YES
3	YES
1 3 3	
4	
5 3 15 2	
5	
1 3 8 7 6	
6	
13 15 39 169 9 5	

提示

对于第一组数据，由于 $n = 1$ ，已经满足所有数字相同，答案为 “YES”。

对于第二组数据，容易发现无论如何操作， a_1 都不可能与 a_2 相等。

对于第三组数据，可以选取 $i = 2$ ， $j = 3$ ， $d = 3$ 并进行第一种操作，使得原序列变为 $[1, 1, 1]$ ，此时所有数字均相同，因此输出 “YES”。

F Flower

时间限制: 1 second

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

Yuki 的面前有一朵花，这朵花共有 n 片花瓣。

她将会进行若干轮操作。每轮操作，她会先摘下花上 a 片花瓣，再摘下 b 片花瓣。如果剩余的花瓣数量不足，则全部摘下。若某次操作结束后，没有留下任何花瓣，她就会结束操作。

Yuki 曾经许下这样的约定：当且仅当她摘下的最后一片花瓣属于某轮操作里摘下的前 a 片花瓣时，她就会离开，否则她才能留下。出于怜惜，你希望先摘下一些花瓣，但不能摘下所有的花瓣，使得她最后会留下。求出你最少需要摘下的花瓣个数；特别地，如果无论如何她都会离开，输出“Sayonara”。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 100$)，表示测试数据组数。每组测试数据的格式如下。

输入的第一行，也是唯一一行，包含三个正整数 n , a 和 b ($1 \leq n \leq 10^9, 1 \leq a \leq 10^9, 1 \leq b \leq 10^9$)。

输出格式

对于每组测试数据，输出一行：

- 若你可以使她留下，则输出一个整数，表示你至少需要摘下的花瓣的数量；
- 若你无法使她留下，则输出一个字符串“Sayonara”。

样例

standard input	standard output
3	Sayonara
1 2 3	0
10 2 3	1
11 2 3	

提示

对于第一组数据， $n = 1, a = 2, b = 3$ 。你无法摘下任何一片花瓣，而 Yuki 则会在第一轮摘下所有的花瓣，从此离开。故输出“Sayonara”。

对于第二组数据，容易发现，即使初始不摘下花瓣，Yuki 也会在两轮操作结束后恰好摘掉所有花瓣，从而留下，所以答案为 0。



Gellyfish and Priority Queue

时间限制: 5 seconds

输入文件: standard input

空间限制: 1024 megabytes

输出文件: standard output

有一个小根堆 Q ，初始为空。Gellyfish 将对其执行 m 次操作。每次操作属于以下两种类型之一：

- **插入**: 给定 l, r 。他将从 $[l, r]$ 中均匀随机地生成一个整数 x ，然后将其插入到 Q 中。
- **弹出**: 删除 Q 中当前的最小元素。保证 Q 此时不是空的。

在所有操作完成后，Gellyfish 希望你计算 Q 中所有剩余元素的乘积的期望值。答案对 998244353 取模。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 200$)，表示测试数据组数。每组测试数据的格式如下。

第一行，包含两个整数 m 和 v ($1 \leq m \leq 500, 1 \leq v \leq 500$)。

接下来有 m 行；第 i 行包含一个或三个整数，表示一个操作。如果操作是插入，则会有三个整数 $o = 1, l$ 和 r ($1 \leq l \leq r \leq v$)，以空格分隔；如果操作是弹出，则这一行仅由一个整数 $o = 2$ 组成。

保证所有测试数据中， m 的和不超过 2000。还保证在执行弹出操作时，优先队列非空。

输出格式

对于每组数据，输出一行一个整数，表示优先队列中所有剩余元素的乘积的期望值，对 998244353 取模后的结果。

样例

standard input	standard output
5	1
2 1	873463812
1 1 1	5
2	255107002
3 4	624423011
1 1 4	
1 1 4	
2	
5 6	
1 1 3	
1 4 6	
2	
1 2 4	
2	
7 8	
1 1 6	
1 2 4	
2	
1 1 5	
1 3 7	
2	
1 2 8	
9 10	
1 3 8	
1 2 9	
1 1 6	
2	
1 2 5	
2	
1 7 10	
1 4 8	
2	

提示

对于第一组数据，注意空优先队列中所有元素的乘积被视为 1。

对于第二组数据，前两次操作我们 $[1,4]$ 中的随机整数插入优先队列，然后，删去较小的一个，使得所有剩余元素的乘积恰好等于较大的整数。因此，期望乘积为 $\frac{1}{16} \cdot (1 + 2 + 3 + 4 + 2 + 2 + 3 + 4 + 3 + 3 + 3 + 4 + 4 + 4 + 4 + 4) = \frac{25}{8}$ ，并且 $\frac{25}{8} \equiv 873463812$

(mod 998244353)。

对于第三组数据，由于 $3 < 4$ 且 $4 \leq 4$ ，第二次插入操作插入的数始终是优先队列中的最大元素。由于最终优先队列的大小变为 1，因此期望答案为第二次被插入的数字的期望，即 $\frac{1}{3} \cdot (4 + 5 + 6) = 5$ 。

H Head out to the Target

时间限制: 3 seconds

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

给定一棵包含 n 个节点的有根树。根节点是节点 1。对于每个节点 i ($2 \leq i \leq n$), 其父亲节点为 f_i , 这表示节点 i 和节点 f_i 之间有一条边。一个棋子最初位于节点 1。

在这个问题中, 我们认为时间取从 1 开始的所有离散时刻。共有 k 个 **不重叠的** 时间区间。在每个区间 $[l_i, r_i]$ 内, 节点 u_i 上会出现一个目标。你可以在任何时刻 (包括时刻 0, 也可以执行任意多次) 切断树中的任意数量的边。一旦一条边被切断, 它将被 **永久移除**。

在任何时刻:

- 如果目标存在 (即当前时间处于某个 $[l_i, r_i]$ 内), 并且
- 棋子和目标在同一个连通块内, 并且
- 棋子当前不在目标节点上,

那么, 棋子将沿着通往目标的唯一简单路径移动一步。如果棋子到达目标节点 (或在时刻开始时已经在目标节点上), 我们称发生了一次 **重合**。

你的目标是通过选定一个切断边的策略, 确定棋子可以 **重合** 于某个目标的最早时刻。如果永远无法与任何目标 **重合**, 则输出 -1 。

输入格式

输入的第一行包含两个整数 n 和 k ($1 \leq n \leq 10^6, 1 \leq k \leq 10^6$) — 树的大小和区间的数量。

第二行包含 $(n-1)$ 个整数 f_2, \dots, f_n ($1 \leq f_i < i$) — 节点 $2, \dots, n$ 的父节点。

接下来有 k 行。第 i 行包含三个整数 u_i, l_i 和 r_i ($1 \leq u_i \leq n, 1 \leq l_i \leq r_i \leq 10^9$), 表示在时刻 l_i 到 r_i (包含) 之间, 目标出现在节点 u_i 。

保证所有时间区间按顺序给出且不重叠; 即, 对于每个 $1 \leq i < k$, 都有 $r_i < l_{i+1}$ 。

输出格式

输出一个整数: 物体可以与目标重合的最小时刻。如果不可能, 输出 -1 。

样例

standard input	standard output
8 3 1 1 1 4 4 6 7 5 1 1 8 2 3 4 5 5	5
5 5 1 2 3 3 4 4 6 5 7 7 1 8 8 2 9 9 1 10 10	6
5 1 1 2 3 4 3 1 1	-1

提示

在第一个测试用例中，可能的最佳策略之一描述如下：

- 时刻 1：物体从节点 1 移动到节点 4。
 - 移动完成后，我们切断节点 6 和节点 7 之间的边。
- 时刻 2 和 3：节点 4（物体当前所在的节点）和节点 8（目标节点）不连通，因此物体不移动。
- 时刻 4：树上没有目标，因此物体也不移动。
- 时刻 5：目标恰好出现在物体所在的节点 4，导致 重合。

可以很容易证明，没有策略允许在时刻 5 之前发生 重合。因此，答案是 5。

I

Infinity

时间限制: 3 seconds

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

设 S_n 是所有 n 阶置换构成的集合。对于 $\sigma \in S_n$, 设 $\nu(\sigma)$ 是集合 $\{\mu^{-1}\sigma\mu \mid \mu \in S_n\}$ 中的元素数量。
固定 k , 多组询问 n , 请计算

$$\sum_{\sigma \in S_n} \nu(\sigma)^k,$$

答案对 998244353 取模。

输入格式

第一行包含两个整数 t 和 k ($1 \leq t \leq 10^3, 1 \leq k \leq 10^9$), 分别表示数据组数, 和一个题目描述给定的常量。
接下来 t 行, 每行一个正整数 n ($1 \leq n \leq 2 \cdot 10^5$) 表示一次询问。

输出格式

输出共 t 行。对于每组询问, 输出一行一个整数, 表示答案对 998244353 取模后的结果。

样例

standard input	standard output
7 1	1
1	2
2	14
3	146
4	2602
5	71412
6	2675724
7	
8 5	2
2	738630138
10	158601508
50	770726379
250	894346889
1250	987370821
6250	445104440
31250	460384381
100000	



时间限制: 1 second 输入文件: standard input
空间限制: 512 megabytes 输出文件: standard output

Yuki 和 Ena 在打某回合制扑克游戏。

初始时, Yuki 有 x 个筹码, Ena 有 y 个筹码。每回合会确定该回合的获胜方, 确定方式如下:

- 若 Yuki 的筹码量和 Ena 的筹码量不相等, 则筹码量较少的一方为获胜方;
- 若 Yuki 的筹码量和 Ena 的筹码量相等, 则 Yuki 为获胜方。

另一方需要向获胜方支付的筹码量等同于获胜方所持有的筹码量。

当某一方的筹码量变为 0 时, 游戏立即结束。请求出游戏是否会在有限回合内结束。特别地, 若游戏会结束, 则你还要求出游戏从开始到结束历经的回合数。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 10^5$), 表示测试数据组数。每组测试数据的格式如下。

输入的第一行, 也是唯一一行, 包含两个正整数 x 和 y ($1 \leq x, y \leq 10^9$), 表示 Yuki 和 Ena 初始的筹码数。

输出格式

对于每组测试数据, 输出一行一个整数: 若游戏能够在有限回合内结束, 输出一行一个整数表示回合数; 否则仅输出 -1 即可。

样例

standard input	standard output
5	1
100 100	2
30 90	-1
15 55	-1
11 4514	6
2 126	

提示

对于第一组数据, 第一回合中 Yuki 和 Ena 的筹码量都为 100, 因此 Yuki 为获胜方。Yuki 的筹码量变为 $100 + 100 = 200$, Ena 的筹码量变为 $100 - 100 = 0$ 。此时, 游戏立即结束, 经过的回合数为 1。

对于第二组数据, 第一回合中 Yuki 的筹码量为 30, Ena 的筹码量为 90, Yuki 的筹码量较少, 因此 Yuki 为获胜方。Yuki 的筹码量变为 $30 + 30 = 60$, Ena 的筹码量变为 $90 - 30 = 60$ 。第二回合中, Yuki 的筹码量和 Ena 的筹码量都为 60, 同理, 经过的回合数为 2。

对于第三组数据，可以证明游戏不能在有限回合内结束。

K Kaleidoscope

时间限制: 2 seconds

输入文件: standard input

空间限制: 512 megabytes

输出文件: standard output

小鸠和小佐内正在调查一个排列的盗窃案³ p_1, \dots, p_n 。已经有一些证词 (x, y) 表明 $p_x = y$ 。可以保证这些证词没有矛盾。为了找出嫌疑人，他们提出了另一个问题：

- 当且仅当 p 和 q 的长度相同，并且前缀最大位置构成的集合相同时，称两个排列 p 和 q 相似。排列 p 的位置 k 被称为前缀最大位置，当且仅当 $p_k = \max(p_1, \dots, p_k)$ 。
- 小佐内想要计算不同排列 q 的数量，使得存在一种方法填补 p_1, \dots, p_n 中所有缺失的位置，使得 p 与 q 相似。

请帮助小鸠算出答案。由于答案可能很大，只需输出对 998244353 取模后的结果即可。

输入格式

本题单个测试点内有多组数据。第一行仅包含一个整数 t ($1 \leq t \leq 1.8 \cdot 10^3$)，表示测试数据组数。每组测试数据的格式如下。

输入的第一行包含一个整数 n ($1 \leq n \leq 5 \cdot 10^3$)。

输入的第二行包含 n 个整数 p'_1, \dots, p'_n ($0 \leq p'_i \leq n$)。如果 $p'_i = 0$ ，则当前没有关于 p_i 的信息；否则，有证词表明 $p_i = p'_i$ 。保证不存在数对 (i, j) 满足 $1 \leq i < j \leq n$ 且 $p'_i = p'_j > 0$ 。

保证所有测试数据中， n 的和不超过 10^4 。

输出格式

对于每组数据，输出仅一行一个整数，表示可能与 p 相似的不同排列的数量，答案对 998244353 取模。

³长度为 n 的排列是一个由 1 到 n 的 n 个不同整数按任意顺序组成的数组。例如， $[2, 3, 1, 5, 4]$ 是一个排列，但 $[1, 2, 2]$ 不是一个排列（2 在数组中出现了两次），而 $[1, 3, 4]$ 也不是一个排列（ $n = 3$ 但数组中有 4）。

样例

standard input	standard output
13	4
3	1
0 0 2	12
3	2
0 1 3	11
4	12
0 1 0 0	3
4	374
1 3 2 0	5040
4	202820
0 3 0 0	1360800
4	16287920
0 0 2 1	399815880
5	
3 1 4 2 5	
6	
0 0 3 0 5 0	
7	
0 0 0 0 0 0 0	
9	
0 0 6 0 0 7 0 0 0	
10	
0 0 3 1 0 0 2 5 0 0	
11	
0 0 6 0 3 8 9 0 0 0 0	
16	
0 0 3 5 0 0 6 0 0 12 0 0 16 0 7 0	

提示

对于第一组数据，有两种可能的排列 p :

- $p = [1, 3, 2]$ 。前缀最大位置的集合是 $\{1, 2\}$ 。有 2 个排列 q 与 p 相似: $[1, 3, 2]$ 和 $[2, 3, 1]$ 。
- $p = [3, 1, 2]$ 。前缀最大位置的集合是 $\{1\}$ 。有 2 个排列 q 与 p 相似: $[3, 1, 2]$ 和 $[3, 2, 1]$ 。

因此，答案是 $2 + 2 = 4$ 。

对于第二组数据，唯一可能的排列是 $p = [2, 1, 3]$ 。前缀最大位置的集合是 $\{1, 3\}$ ，唯一与 p 相似的排列 q 就是 p 本身，所以答案是 1。

对于第三组数据，所有可能与 p 相似的 12 种排列 q 如下: $[2, 1, 3, 4]$, $[2, 1, 4, 3]$, $[3, 1, 2, 4]$, $[3, 1, 4, 2]$,

$[3, 2, 1, 4]$, $[3, 2, 4, 1]$, $[4, 1, 2, 3]$, $[4, 1, 3, 2]$, $[4, 2, 1, 3]$, $[4, 2, 3, 1]$, $[4, 3, 1, 2]$, 和 $[4, 3, 2, 1]$ 。注意, $p = [3, 2, 1, 4]$ 和 $p = [3, 1, 2, 4]$ 都有前缀最大集合 $\{1, 4\}$, 但对应的相似排列 q 仍然只计算一次。