

2026 牛客暑期多校训练营 #3



牛客竞赛
AC.NOWCODER.COM

题目列表

A	比特掩码
B	再买一瓶
C	蛋糕店
D	最长的连续的一
E	扫雷
F	Not Aqre 2
G	矩阵标记
H	季节
I	交换大师
J	树.zip
K	转向导航
L	登山对决
M	漫游者

本试题册共 13 题，18 页。
如果您的试题册不完整，请立即通知志愿者。

2026 年 7 月 25 日

承办方



命题方



题目 A. 比特掩码

对于非负整数 x ，定义 $f(x)$ 为 x 的二进制表示（不含前导零）中极长连续全一段的个数。例如， $f(23) = 2$ ，因为 $23 = (10111)_2$ ，其二进制表示包含 2 个极长连续全一段。

给定 n 个整数 $a_1, a_2, \dots, a_n$ 。你需要处理 m 个操作。每个操作由两个整数 $type$ 和 x 描述。

若 $type = 1$ ，将每个 a_i 替换为 $a_i \& x$ 。

若 $type = 2$ ，将每个 a_i 替换为 $a_i | x$ 。

若 $type = 3$ ，将每个 a_i 替换为 $a_i \oplus x$ 。

其中 $\&$ 表示 按位与 运算， $|$ 表示 按位或 运算， $\oplus$ 表示 按位异或 运算。

每个操作之后，输出 $\sum_{i=1}^n f(a_i)$ 。

输入格式

每个测试文件仅包含一组测试数据。

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

第二行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 2^{30}$)。

第三行包含一个整数 m ($1 \leq m \leq 3 \cdot 10^5$)。

接下来 m 行，每行包含两个整数 $type$ 和 x ($1 \leq type \leq 3$, $0 \leq x < 2^{30}$)，描述一个操作。

输出格式

输出 m 行。第 j 行必须包含第 j 个操作之后 $\sum_{i=1}^n f(a_i)$ 的值。

样例

standard input	standard output
4	5
3 5 6 10	7
4	5
1 7	4
2 8	
3 15	
2 3	

题目 B. 再买一瓶

一瓶饮料售价 1 元。

每次小明喝完一瓶饮料，他会以 $\frac{a}{b}$ 的概率获奖。如果获奖，他将得到 c 元；否则，他什么也得不到。他得到的钱可以用来购买更多饮料。每瓶饮料的获奖事件相互独立。

小明一开始有 n 元。求他恰好喝完 m 瓶饮料后花光所有钱的概率。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 200000$)。每组测试数据的格式如下。

接下来 T 行，每行包含五个整数 n, m, c, a, b ($1 \leq n, m, c \leq 2000000, 0 \leq a < b < 998244353$)，表示小明一开始有 n 元，每次获奖得到 c 元，获奖概率为 $\frac{a}{b}$ ；你需要求出他恰好喝完 m 瓶饮料的概率。

输出格式

输出 T 行，每行包含一个整数：对应测试数据的答案对 998 244 353 取模的结果。

形式化地，设 $M = 998\,244\,353$ 。可以证明，答案可以表示为最简分数 $\frac{p}{q}$ ，其中 p 和 q 为整数且 $q \not\equiv 0 \pmod{M}$ 。输出等于 $p \cdot q^{-1} \bmod M$ 的整数。

样例

standard input	standard output
4	499122177
1 1 2 1 2	873463809
1 3 2 1 2	776412275
2 2 3 1 3	0
1 2 2 1 2	

备注

在第一组测试数据中，恰好喝完 1 瓶的唯一方式是第一瓶没有获奖，其概率为 $\frac{1}{2}$ 。

在第二组测试数据中，恰好喝完 3 瓶的唯一方式是第一瓶获奖且接下来两瓶都没有获奖，其概率为 $\frac{1}{8}$ 。

在第四组测试数据中，小明一开始有 1 元，每次获奖得到 2 元，因此他不可能恰好喝完 2 瓶。

题目 C. 蛋糕店

La Soleil 是一家位于数轴上点 0 处的蛋糕店。Coconut 是该店唯一的配送员，她的速度为每单位时间移动 1 单位距离。

共有 n 个配送订单。第 i 个订单在时刻 t_i 到达蛋糕店，其目的地为数轴上的点 x_i 。所有 t_i 和 x_i 均为正整数。

蛋糕店只有在订单到达之后才能按照该订单的要求打包蛋糕。打包蛋糕不花费时间，因此订单一到达就可以取走，也可以在之后任意时刻取走。Coconut 每次离开蛋糕店时，可以带走任何已经打包好的蛋糕；她不必全部带走，留下的订单可以在之后的某次配送中取走。在一次配送中，她沿数轴正方向移动，配送她带走的全部订单，全部配送完毕后立即返回蛋糕店。在返回蛋糕店之前，Coconut 不能再接一批新订单。

如果第 i 个订单在时刻 d_i 被配送，那么它的等待时间为 $d_i - t_i$ 。你需要安排配送行程，使得所有订单都被配送，并且最小化所有订单中最大的等待时间。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 100$)。每组测试数据的格式如下。

每组测试数据包含三行。

第一行包含一个整数 n ($1 \leq n \leq 10^5$)，表示订单数量。

第二行包含 n 个整数 $t_1, t_2, \dots, t_n$ ($1 \leq t_i \leq 10^9$)，其中 t_i 为第 i 个订单的到达时间。

第三行包含 n 个整数 $x_1, x_2, \dots, x_n$ ($1 \leq x_i \leq 10^9$)，其中 x_i 为第 i 个订单的目的地坐标。

保证同一测试文件中所有测试数据的 n 之和不超过 10^5 。

输出格式

对于每组测试数据，输出一行一个整数：最大等待时间的最小可能值。

样例

standard input	standard output
2	8
4	246
1 2 8 9	
2 3 7 3	
12	
140 59 151 1 97 12 94 155 220 48 27 53	
21 93 109 62 98 71 35 15 120 83 76 12	

备注

在第一个样例中，一种最优安排如下。

在时刻 2，Coconut 带着订单 1 和 2 离开蛋糕店。两个订单分别在时刻 4 和 5 被配送。Coconut 在时刻 8 返回蛋糕店。

在时刻 9，Coconut 带着订单 3 和 4 离开蛋糕店。订单 4 在时刻 12 被配送，订单 3 在时刻 16 被配送。

等待时间分别为 3, 3, 8, 3，因此最大等待时间为 8。

题目 D. 最长的连续的一

给定一个有 n 个顶点和 m 条边的有向无环图 (DAG)。顶点编号从 1 到 n ，每个顶点 i 有一个标号 $a_i \in \{0, 1\}$ 。

图的**拓扑序**是顶点的一个排列 $p_1, p_2, \dots, p_n$ ，满足对于每条有向边 $u \rightarrow v$ ，顶点 u 在排列中出现在顶点 v 之前。

固定一个拓扑序 $p_1, p_2, \dots, p_n$ 。沿此顺序读出标号 $a_{p_1}, a_{p_2}, \dots, a_{p_n}$ ，考虑其中极长的连续全一段。该排列的**得分**为最长这样的段的长度，即所有满足 $a_{p_l} = a_{p_{l+1}} = \dots = a_{p_r} = 1$ 的下标区间 $[l, r]$ 中 $r - l + 1$ 的最大值。如果没有顶点标号为 1，则得分为 0。

在图的所有拓扑序中，求可能的最大得分，并输出一个达到该得分的拓扑序。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 200$)。每组测试数据的格式如下。

每组测试数据的第一行包含两个整数 n 和 m ($1 \leq n \leq 2000$, $0 \leq m \leq \min(10^5, \frac{n(n-1)}{2})$)，分别表示顶点数和边数。

接下来 m 行，每行包含两个整数 u 和 v ($1 \leq u, v \leq n$, $u \neq v$)，表示一条从顶点 u 指向顶点 v 的有向边。保证给定的图是有向无环图，且不存在两条相同的边。

接下来一行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($a_i \in \{0, 1\}$)，表示顶点的标号。

保证同一测试文件中所有测试数据的 n 之和不超过 2000， m 之和不超过 10^5 。

输出格式

对于每组测试数据，输出两行。

第一行包含一个整数 L ，表示所有拓扑序中可能的最大得分。

第二行包含 n 个整数 $p_1, p_2, \dots, p_n$ ，表示一个得分等于 L 的拓扑序。即 $p_1, \dots, p_n$ 必须是 $1, \dots, n$ 的排列，使得每条边 $u \rightarrow v$ 满足 u 在 v 之前，且沿此排列最长的连续全一段的长度恰好为 L 。

如果有多个答案，输出任意一个即可。

样例

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

备注

在第一组测试数据中，唯一的拓扑序为 1, 2, 3，对应标号为 1, 0, 1。顶点 2（标号 0）位于从顶点 1

到顶点 3 的唯一路径上，因此在任何拓扑序中它都被迫排在两者之间。这些一永远无法相邻，因此最优得分为 1。

在第二组测试数据中，排列 1, 2, 3, 4 对应标号 1, 1, 0, 1，产生一个长度为二的连续全一段。标号为 1 的三个顶点为 1, 2, 4。由于边 $1 \rightarrow 3$ 和 $3 \rightarrow 4$ 的存在，在任何拓扑序中顶点 3（标号 0）必须排在顶点 1 之后、顶点 4 之前，因此顶点 1, 2, 4 永远无法全部相邻。故可达到的最优得分为 2。

题目 E. 扫雷

给定一个正整数 x 。你的任务是构造一个扫雷局面，使得它恰好有 x 种合法的雷的放置方案。

该局面是一个矩形网格。每个格子要么未打开，用字符 $*$ 表示；要么已打开，用 0 到 8 之间的一个数字表示。

只有未打开的格子可能含有雷。已打开的格子绝不含雷。如果一个已打开的格子上的数字为 d ，那么它的相邻格子（最多八个）中恰好有 d 个格子含有雷，其中两个格子相邻当且仅当它们共享一条边或一个角。

如果所有已打开的格子都满足其数字要求，则称该雷的放置方案为合法的。雷的总数不固定，因此两个合法的雷的放置方案可能含有不同数量的雷。

对于每组测试数据，构造任意一个恰好有 x 种合法雷放置方案的网格。你的网格中未打开格子的数量不得超过 40 。可以证明，在给定限制下必定存在合法的构造。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 10$)。每组测试数据的格式如下。

接下来 T 行，每行包含一个整数 x ($1 \leq x \leq 300$)。

输出格式

对于每组测试数据，首先输出两个整数 h 和 w ($1 \leq h, w \leq 40$) — 分别表示你的网格的高度和宽度。然后输出 h 行。每行必须是一个长度为 w 的字符串。每个字符必须是 $*$ 或 0 到 8 之间的一个数字。对于每组测试数据，你的网格必须满足以下所有条件：

- 含有 $*$ 的格子数量不超过 40 ；
- 每个数字格子都是合法的已打开扫雷格子；
- 合法的雷放置方案数恰好为 x 。

样例

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

	1

备注

在第一组测试数据中， $x = 4$ 。输出的网格为

*
*

有两个未打开的格子且没有数字约束，因此每个格子都可以独立地选择是否含雷，共有 $2^2 = 4$ 种合法的雷放置方案。

在第二组测试数据中， $x = 5$ 。输出的网格为

$a \quad b \quad c$
 $d \quad 1 \quad e$

格子 a, b, c, d, e 是未打开的格子。数字 1 要求这五个相邻的未打开格子中恰好有一个含雷。因此恰好有 5 种合法的雷放置方案。

题目 F. Not Aqre 2

给定一个 $n \times m$ 的网格。你需要在每个格子中填入 0、1、2 中的一个整数。
如果任意两个共享一条边的格子填有不同的整数，则称该填法为好的。
输出好的填法数对 998 244 353 取模的结果。

输入格式

每个测试文件仅包含一组测试数据。
唯一一行包含两个整数 n 和 m ($1 \leq n < 10, 1 \leq m < 998\,244\,353$)。

输出格式

输出一个整数表示答案。

样例

standard input	standard output
2 2	18

备注

在样例中，网格有 2 行 2 列。
第一列有 6 种选法：01, 02, 10, 12, 21, 20。对于第一列的任意一种选法，第二列恰好有 3 种合法的选法。因此答案为 $6 \cdot 3 = 18$ 。

题目 G. 矩阵标记

给定一个 $n \times m$ 的数字网格，每个格子中包含一个正整数。

如果存在两个格子 (r_1, c_1) 和 (r_2, c_2) 满足 $r_1 < r_2$ 且 $c_1 < c_2$ ，且这两个格子中的数字相等，那么所有满足 $r_1 \leq r \leq r_2$ 且 $c_1 \leq c \leq c_2$ 的格子 (r, c) 都会被标记。

输出整个网格最终的标记情况。

输入格式

每个测试文件仅包含一组测试数据。

第一行包含两个整数 n, m ($1 \leq n, m, n \times m \leq 1\,000\,000$)。

接下来 n 行，每行包含 m 个整数；其中第 i 行包含 $a_{i,1}, a_{i,2}, \dots, a_{i,m}$ ($1 \leq a_{i,j} \leq n \times m$)，描述该数字网格。

输出格式

输出 n 行，每行包含一个长度为 m 、由字符 0 和 1 组成的字符串。

第 i 行的第 j 个字符为 1 表示格子 (i, j) 最终被标记，为 0 表示未被标记。

样例

standard input	standard output
2 6	001110
1 2 3 4 5 6	001110
7 8 9 3 4 10	

备注

在样例中，第 1 行第 3 列的格子和第 2 行第 4 列的格子都含有数字 3，因此它们标记了两行中第 3 列到第 4 列的所有格子。

第 1 行第 4 列的格子和第 2 行第 5 列的格子都含有数字 4，因此它们标记了两行中第 4 列到第 5 列的所有格子。两个矩形的并集即为最终答案。

题目 H. 季节

如果一个数组 $B_0, B_1, \dots, B_{n-1}$ 满足对所有满足 $0 \leq j < j+p < n$ 的合法下标 j 都有 $B_j = B_{j+p}$, 则称该数组以 p 为周期。

给定一个在模 998 244 353 意义下的数组 $A_0, A_1, \dots, A_{n-1}$ 。

求最小的非负整数 T , 使得存在数组 $B^{(1)}, B^{(2)}, \dots, B^{(T)}$ 满足:

- 对每个 $1 \leq i \leq T$, 数组 $B^{(i)}$ 以 i 为周期;
- 对每个 $0 \leq j < n$,

$$A_j \equiv \sum_{i=1}^T B_j^{(i)} \pmod{998\,244\,353}$$

当 $T = 0$ 时, 求和为空, 因此只有全零数组能被表示。

输入格式

每个测试文件仅包含一组测试数据。

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 $A_0, A_1, \dots, A_{n-1}$ ($0 \leq A_i < 998\,244\,353$)。

输出格式

输出一个整数: T 的最小可能值。

样例

standard input	standard output
3 1 2 3	3
5 7 7 7 7 7	1
4 0 0 0 0	0

备注

在第一个样例中, 仅使用周期 1 和 2 会迫使位置 0 和 2 相等, 因此周期 3 是必需的。
在第二个样例中, 数组本身就以 1 为周期。

题目 I. 交换大师

对于一个数组 $a_1, a_2, \dots, a_n$ ，定义其价值为 $\sum_{i=1}^{n-1} |a_i - a_{i+1}|$ 。

你可以选择两个不同的位置并交换这两个位置上的元素，该操作最多执行一次。你也可以选择不改变数组。

求数组可能的最大价值。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 100$)。每组测试数据的格式如下。

每组测试数据的第一行包含一个整数 n ($2 \leq n \leq 5 \cdot 10^5$)。

第二行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq 10^9$)。

保证同一测试文件中所有测试数据的 n 之和不超过 $5 \cdot 10^5$ 。

输出格式

对于每组测试数据，输出一个整数：最多进行一次交换后数组可能的最大价值。

样例

standard input	standard output
3	14
5	0
1 3 2 7 4	45
4	
5 5 5 5	
6	
10 1 10 1 10 1	

备注

在第一组测试数据中，一种最优选择是交换 a_2 和 a_5 。数组变为 $[1, 4, 2, 7, 3]$ ，其价值为 $3 + 2 + 5 + 4 = 14$ 。

题目 J. 树.zip

给定一棵以顶点 1 为根的树，树有 n 个顶点，编号从 1 到 n 。

你需要构建一棵新树，同样以顶点 1 为根，使用相同的 n 个顶点，并满足以下条件：对于每个顶点 $x \neq 1$ ，它在新树中的父顶点必须是 x 在原树中的祖先。

此外，给定 q 个约束。每个约束的形式为 (u, v) ，表示在新树中 u 必须是 v 的后代，等价地， v 必须是 u 的祖先。

设 dep_i 为顶点 i 在新树中的深度，其中根 1 的深度为 0。最小化 $\sum dep_i$ ，并输出该最小值。

输入格式

每个测试文件仅包含一组测试数据。

第一行包含两个整数 n, q ($2 \leq n \leq 500\,000, 0 \leq q \leq 500\,000$)。

接下来 $n - 1$ 行，每行包含一个整数；其中第 $(i - 1)$ 行包含一个整数 p_i ($1 \leq p_i < i$)，表示在原树中顶点 i 的父顶点是 p_i 。

接下来 q 行，每行包含两个整数 u, v ($2 \leq u \leq n, 1 \leq v \leq n$)，描述一个约束：在新树中 u 必须是 v 的后代。保证 $u \neq v$ ，且 v 是 u 在原树中的祖先。

输出格式

输出一个整数： $\sum dep_i$ 的最小可能值。

样例

standard input	standard output
7 4 1 2 3 3 2 6 4 2 4 3 5 2 7 2	11

备注

在样例中，原树的边为 $1 - 2, 2 - 3, 3 - 4, 3 - 5, 2 - 6, 6 - 7$ 。

新树中父顶点的一种最优选择为：

$$parent(2) = 1, parent(3) = 2, parent(4) = 3, parent(5) = 2, parent(6) = 1, parent(7) = 2.$$

新树中的深度分别为 0, 1, 2, 3, 2, 1, 2，其和为 11。

题目 K. 转向导航

车载导航系统引导驾驶员沿着地图上绘制好的规划路线行驶。我们采用平面的标准方向： x 轴正方向指向东， y 轴正方向指向北。

路线由 n 个航点 $P_1, P_2, \dots, P_n$ 组成，每个航点具有整数坐标。汽车沿直线段行驶，始终向前移动：先从 P_1 行驶到 P_2 ，再从 P_2 行驶到 P_3 ，依此类推，直到到达 P_n 。

每当汽车经过一个中间航点 P_i ($2 \leq i \leq n-1$) 时，系统会播报驾驶员在该处的转弯情况。系统将汽车到达的方向（从 P_{i-1} 到 P_i ）与离开的方向（从 P_i 到 P_{i+1} ）进行比较：

- 如果离开方向是由到达方向**逆时针**旋转一个严格介于 0° 与 180° 之间的角度得到的，则驾驶员向左转，系统播报 **LEFT**；
- 如果是由到达方向**顺时针**旋转一个严格介于 0° 与 180° 之间的角度得到的，则驾驶员向右转，系统播报 **RIGHT**；
- 如果方向完全不变，则汽车继续直行，系统播报 **STRAIGHT**。

保证汽车绝不会掉头，因此离开方向绝不会与到达方向恰好相反。于是，每个中间航点处恰好适用三种播报中的一种。

给定路线，按顺序输出每个中间航点的播报。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 500$)。每组测试数据的格式如下。

每组测试数据的第一行包含一个整数 n ($3 \leq n \leq 10^5$)，表示路线上的航点数。

接下来 n 行，每行包含两个整数 x_i 和 y_i ($-10^9 \leq x_i, y_i \leq 10^9$)，表示第 i 个航点 P_i 的坐标，按汽车经过的顺序给出。保证每两个相邻航点互不相同（对所有 $1 \leq i \leq n-1$ 都有 $P_i \neq P_{i+1}$ ），且汽车绝不会掉头（对所有 $2 \leq i \leq n-1$ ，从 P_i 到 P_{i+1} 的方向与从 P_{i-1} 到 P_i 的方向不会恰好相反）。

保证同一测试文件中所有测试数据的 n 之和不超过 $2 \cdot 10^5$ 。

输出格式

对于每组测试数据，输出中间航点 $P_2, P_3, \dots, P_{n-1}$ 的播报。每个播报是字符串 **LEFT**、**RIGHT** 或 **STRAIGHT** 之一。

样例

standard input	standard output
2	LEFT LEFT
4	STRAIGHT RIGHT
0 0	
1 0	
1 1	
0 1	
4	
0 0	
2 0	
4 0	
4 -2	

备注

在第一组测试数据中，汽车从原点出发逆时针绕一个单位正方形行驶。在 $P_2 = (1, 0)$ 处，它原本朝东行驶，转向朝北，是一次左转；在 $P_3 = (1, 1)$ 处，它原本朝北行驶，转向朝西，又是一次左转。因此答案为 LEFT LEFT。

在第二组测试数据中，汽车先从 $(0, 0)$ 朝东行驶到 $(2, 0)$ ，然后继续朝东行驶到 $(4, 0)$ ，因此在 $P_2 = (2, 0)$ 处它继续直行。在 $P_3 = (4, 0)$ 处，它原本朝东行驶，转向朝南，是一次右转。因此答案为 STRAIGHT RIGHT。

题目 L. 登山对决

两名登山者，先手玩家和后手玩家，在一张山地地图上博弈。

地图是一个 n 行 m 列的网格。第 i 行第 j 列的格子有一个整数高度 $h_{i,j}$ 。地图上所有高度互不相同：没有两个格子具有相同的高度。

一个格子上放有一面旗帜。两名玩家交替移动旗帜，先手玩家先手。一名玩家的回合中：

- 该玩家必须将旗帜从当前格子移动到一个四相邻（共享一条边）且高度**严格更高**的格子；
- 如果当前格子不存在四相邻且高度严格更高的格子，则轮到移动的玩家无法移动并**输掉**游戏，另一方获胜。

由于每次移动都到一个严格更高的格子，旗帜永远不会回到它已经离开的格子，因此游戏总会在有限步之内结束。双方都采取最优策略。

给定 q 个独立的询问。每个询问指定旗帜的起始格子；地图本身在询问之间不会改变。对于每个询问，判断在双方最优博弈下谁会获胜：先手玩家还是后手玩家。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 500$)。每组测试数据的格式如下。

每组测试数据的第一行包含两个整数 n 和 m ($1 \leq n, m \leq 10^5$, $n \cdot m \leq 10^5$)，分别表示地图的行数和列数。

接下来 n 行描述高度。其中第 i 行包含 m 个整数 $h_{i,1}, h_{i,2}, \dots, h_{i,m}$ ($1 \leq h_{i,j} \leq 10^9$)，其中 $h_{i,j}$ 为第 i 行第 j 列格子的高度。保证同一组测试数据内的 $n \cdot m$ 个高度互不相同。

接下来一行包含一个整数 q ($1 \leq q \leq 10^5$)，表示询问数。

接下来 q 行，每行包含两个整数 r 和 c ($1 \leq r \leq n$, $1 \leq c \leq m$)，表示该询问中旗帜的起始格子为第 r 行第 c 列。

保证同一测试文件中所有测试数据的 $n \cdot m$ 之和不超过 10^5 ， q 之和不超过 10^5 。

输出格式

对于每个询问，输出一行：如果先手玩家在最优博弈下获胜则输出 **First**，如果后手玩家获胜则输出 **Second**。

样例

standard input	standard output
1	First
2 3	Second
1 2 3	Second
6 5 4	Second
5	First
1 1	
1 2	
2 1	
2 3	
2 2	

备注

在样例中, 地图有 2 行 3 列。第 1 行为 1 2 3, 第 2 行为 6 5 4, 例如 $h_{2,1} = 6$ 是最高的格子, $h_{1,1} = 1$ 是最低的格子。

站在没有严格更高相邻格子的玩家立即输掉。

- 询问 (2,1) 的高度为 6, 是整张地图的最大值, 因此先手玩家无法移动并立即输掉: 答案为 **Second**。
- 询问 (2,3) 的高度为 4。它唯一严格更高的相邻格子是 (2,2), 高度为 5, 而 (2,2) 唯一严格更高的相邻格子是 (2,1), 高度为 6。因此博弈过程是强制的: $4 \rightarrow 5 \rightarrow 6$ 。先手玩家移动到 5, 后手玩家移动到 6, 此时先手玩家无路可走并输掉。尽管 (2,3) 不是最高格子, 答案仍为 **Second**, 这说明胜负不能仅凭简单的局部特征判断。
- 询问 (1,1) 的高度为 1。先手玩家可以移动到 (1,2) (高度 2), 而该位置对后手玩家而言是必败局面, 因此答案为 **First**。

题目 M. 漫游者

有一棵 n 个顶点的树。一个漫游者从顶点 s 出发，想要到达顶点 t 。漫游者希望尽可能避免走回头路，但他的记忆力很差，只能记住上一步。

如果 $s = t$ ，旅程立即结束，步数为 0。

否则，漫游者按以下规则移动。

- 第一步，他从 s 的相邻顶点中等概率随机选择一个并移动过去。
- 在之后的每一步中，假设他当前位于顶点 x ，上一步所在的顶点为 y 。如果 x 的度数大于 1，他从 x 的相邻顶点中除 y 之外的顶点中等概率随机选择一个并移动过去。如果 x 的度数等于 1，那么 y 是 x 唯一的相邻顶点，因此他必须回到 y 。
- 漫游者一到达顶点 t ，旅程即结束。

求旅程步数的期望，对 998 244 353 取模。

输入格式

每个测试文件包含多组测试数据。第一行包含测试数据的组数 T ($1 \leq T \leq 100$)。每组测试数据的格式如下。

第一行包含三个整数 n, s, t ($2 \leq n \leq 5 \cdot 10^5, 1 \leq s, t \leq n$)。

第二行包含 $n - 1$ 个整数 $p_2, p_3, \dots, p_n$ ($1 \leq p_i < i$)。对每个 $2 \leq i \leq n$ ，顶点 i 和 p_i 之间有一条边。保证给定的边构成一棵树。

保证同一测试文件中所有测试数据的 n 之和不超过 $5 \cdot 10^5$ 。

输出格式

对于每组测试数据，输出一个整数：步数期望对 998 244 353 取模的结果。

形式化地，设 $M = 998\,244\,353$ 。可以证明，答案可以表示为最简分数 $\frac{p}{q}$ ，其中 p 和 q 为整数且 $q \not\equiv 0 \pmod{M}$ 。输出等于 $p \cdot q^{-1} \bmod M$ 的整数。

样例

standard input	standard output
4	3
4 2 4	2
1 2 3	2
3 2 1	665496239
1 2	
3 3 1	
1 2	
4 1 2	
1 1 1	

备注

对于第一个样例，边为 $1 - 2, 2 - 3, 3 - 4$ 。

漫游者从顶点 2 出发，想要到达顶点 4。

第一步，他以各 $\frac{1}{2}$ 的概率选择顶点 1 或顶点 3。

如果他移动到顶点 3，则下一步必须是顶点 4，因此旅程为 2 步。

如果他移动到顶点 1，则他必须回到顶点 2，再到顶点 3，最后到顶点 4，因此旅程为 4 步。
因此，步数期望为 $\frac{2+4}{2} = 3$ 。