

NOIP 模拟赛

题目名称	矩阵	毛毛虫	虚无之地	军训
英文名称	matrix	caterpillar	nothingness	training
输入/输出文件	matrix.in/out	caterpillar.in/out	nothingness.in/out	training.in/out
时间限制	1.5s	2s	2s	2s
空间限制	512M	512M	512M	512M
测试点/ 子任务数量	5	9	6	4
测试点是否等分	否	否	否	否

矩阵 (matrix)

小 A 现在有一个 $n \times m$ 的矩阵 x , 设 $A_i = \min_{j=1}^m x_{i,j}$, $B_j = \max_{i=1}^n x_{i,j}$, 他想问你在所有 $x_{i,j} \in [1, k]$ 的整数矩阵中, 生成的不同序列对 (A, B) 有多少种。由于答案可能很大, 请输出答案对 998244353 取模的结果。

输入格式

从文件 `matrix.in` 输入。

一行三个正整数 n, m, k , 意义同问题面中所述。

输出格式

输出到文件 `matrix.out`。

一行一个非负整数代表问题的答案。

样例 1

输入

```
2 2 2
```

输出

```
7
```

解释

(A_1, A_2, B_1, B_2) 四元组可以为
 $(1, 1, 1, 1), (1, 1, 1, 2), (1, 1, 2, 1), (1, 1, 2, 2), (1, 2, 2, 2), (2, 1, 2, 2), (2, 2, 2, 2)$ 。

样例 2~6

见下发文件中的 `matrix*.in/out`, 分别满足第 1~5 个子任务的数据范围。

数据范围

对于所有数据保证 $1 \leq n, m \leq 10^9$, $1 \leq k \leq 3 \times 10^7$, 输入的数都是正整数。

子任务编号	$n \leq$	$m \leq$	$k \leq$	分值
1	1	10^6	10^6	10
2	4	4	2	10
3	3	3	3×10^7	20
4	10^6	10^6	10^6	20
5	10^9	10^9	3×10^7	40

毛毛虫 (caterpillar)

小 B 现在有一颗 n 个点的树，即一个点数为 n ，边数为 $n - 1$ 的简单无向连通图，点集为 $V = \{1, 2, \dots, n\}$ ，对于两个点 u, v 定义 $\text{dis}(u, v)$ 为它们之间的距离，即 u, v 之间的简单路径的边数。

定义对于三元组 (u, v, k) ，定义其毛毛虫为 $\text{Caterpillar}(u, v, k) = \{p \mid \exists q \in V, \text{dis}(u, q) + \text{dis}(v, q) = \text{dis}(u, v), \text{dis}(p, q) \leq k\}$ ，即与路径 (u, v) 之间距离 $\leq k$ 的所有点构成的集合。

小 B 会问你 q 个问题，问题形如：

给定一个点集 $T \subseteq V$ ，询问最小的 k ，使得 $\exists v \in V, T \subseteq \text{Caterpillar}(1, v, k)$ 。

输入格式

从文件 `caterpillar.in` 输入。

第一行三个正整数 n, q 。

接下来 $n - 1$ 行，每行两个正整数 u, v 代表树上的一条边。

接下来 q 行，每行第一个数代表询问集合 T 大小，接下来 $|T|$ 个数代表集合 T 。

输出格式

输出到文件 `caterpillar.out`。

q 行，每行一个非负整数代表对应询问的答案。

样例 1

输入

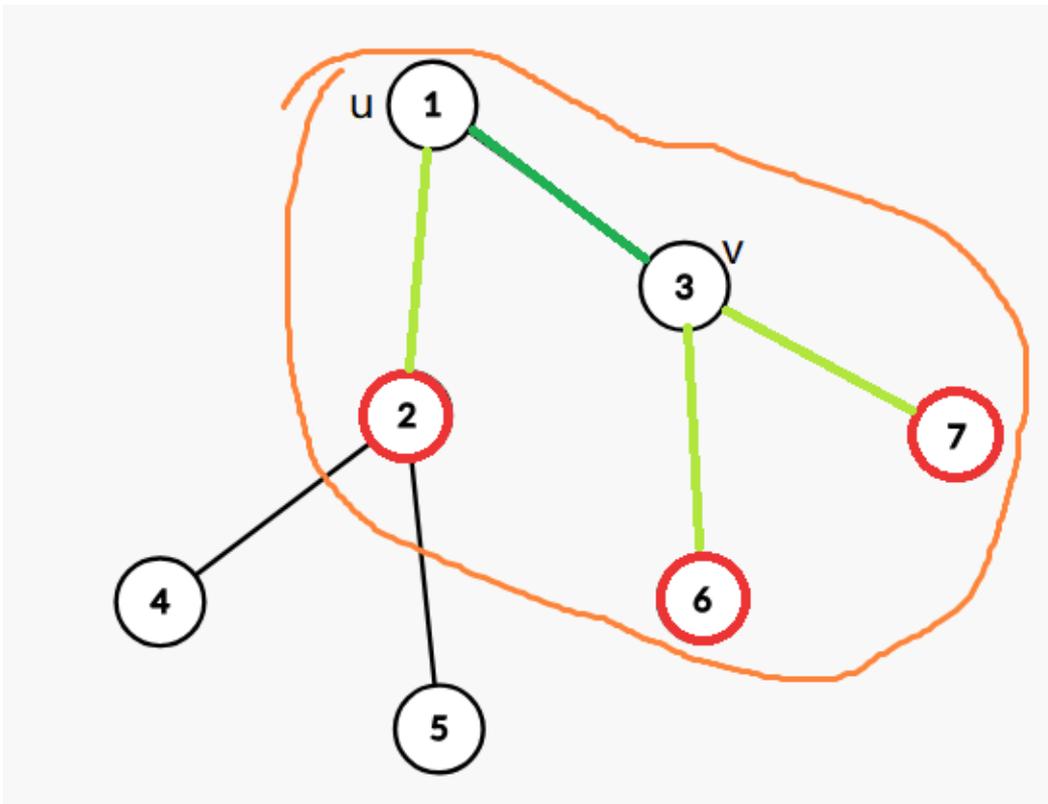
7 3
1 2
1 3
2 4
2 5
3 6
3 7
3 2 6 7
2 3 6
4 4 5 6 7

输出

1
0
2

解释

对于第一组询问， $(1, 3, 1)$ 对应的毛毛虫如图所示为 $\{1, 2, 3, 6, 7\}$ ，满足题目要求：



可以证明 k 最小只能为 1，因此该询问的答案为 1。

样例 2~10

见下发文件中的 caterpillar*.in/out，分别满足第 1~9 个子任务的数据范围。

数据范围

下面记 $\sum |T| = m$ 。

对于所有数据保证 $2 \leq n, m \leq 10^6$ ， $|T| \geq 1$ ，输入构成合法的树，输入的数都是正整数。

子任务编号	$n \leq$	$m \leq$	特殊性质	分值
1	10	10^5	无	5
2	300	300	无	10
3	2000	2000	无	15
4	10^5	10^5	A	1
5	10^5	10^5	B	9
6	10^5	10^5	C	10
7	10^5	10^5	D	15
8	10^5	10^5	无	10
9	10^6	10^6	无	25

特殊性质 A: $|T| = 1$;

特殊性质 B: 输入的树是一颗以 1 为根的菊花图，即对于 $u \in \{2, 3, \dots, n\}$ 树中存在边 $(1, u)$ 。

特殊性质 C: $|T| \leq 2$;

特殊性质 D: 输入的树以这种方式随机生成：对于点 $u \in \{2, 3, \dots, n\}$ ，等概率随机一个点 $1 \leq v < u$ ，连边 (u, v) ;

虚无之地 (nothingness)

题目描述

虚无之地可以看做是一个长宽都近乎无限大的棋盘格，我们称处于第 x 行第 y 列的区域为 (x, y) ，每块区域都有一个指路人。

现有若干支探险队想要探索虚无之地，他们将根据指路人的指引进行探索。具体的，在指引下，

第 k 支到达 (x, y) 的队伍将会在下一步前往
$$\begin{cases} (x, y + 1) & , 2 \nmid k \\ (x + 1, y) & , 2 \mid k \end{cases}$$

初始时 (第 0 天) 有一支队伍到达 $(0, 0)$ 。

从第 1 天开始，每天会依次发生如下事件：

- 现有的队伍同时前进一步。
- 一支新的队伍到达 $(0, 0)$ 。

你需要回答 q 个问题，问题形如：

输入 t, x, y ，询问在第 t 天将要结束时， (x, y) 上是否存在队伍。

输入格式

从文件 [nothingness.in](#) 输入。

第一行三个正整数 q 。

接下来 q 行，每行三个数， t, x, y ，意义见题目描述。

输出格式

输出到文件 [nothingness.out](#)。

q 行，如果存在输出 YES 否则输出 NO。

样例 1

输入

8
0 0 0
0 1 0
1 0 1
1 1 0
2 0 1
2 1 0
2 1 1
2 0 2

输出

YES
NO
YES
NO
NO
YES
NO
YES

解释

第 0 天末 1 支队伍在 $(0, 0)$

第 1 天末 2 支队伍分别在 $(0, 0), (0, 1)$

第 2 天末 3 支队伍分别在 $(0, 0), (1, 0), (0, 2)$

样例 2~7

见下发文件中的 `nothingness*.in/out`, 分别满足第 1~6 个子任务的数据范围。

数据范围

对于所有数据保证 $1 \leq q \leq 10^4$, $x \leq 100$, $y \leq 100$ 。

子任务编号	$t \leq$	$x \leq$	$y \leq$	特殊性质	分值
1	10^{18}	0	100	无	5
2	10^6	100	100	无	15
3	10^{18}	2	2	无	10

子任务编号	$t \leq$	$x \leq$	$y \leq$	特殊性质	分值
4	10^{18}	8	8	无	15
5	10^6	100	100	A	15
6	10^{18}	100	100	无	40

特殊性质 A: $t - x - y$ 为定值。

军训 (training)

题目描述

巴拉姆学院的新生军训开始了。

巴拉姆学院的操场可以视为一个由 n 个格子组成的环，格子依次编号为 $0, 1, \dots, n - 1$ ，其中 i 号格子的下一个格子是 $(i + 1) \bmod n$ 号格子。

$\forall i \in [0, n)$ ， i 号格子有训练值 d_i 。军训过程中，由于学生们不喜欢走得太快，当**进入** i 号格子的时候，他们会依次执行以下两个操作：

- 若当前行进速度 $v > \frac{1}{d_i}$ ，令 v 变为 $\frac{1}{d_i}$ ；否则 v 不变。
- 以速度 v 匀速通过这个格子，花费 $\frac{d_i}{v}$ 的时间。

由于新生太多，军训是分批进行的。同时，军训期间校长希德可能会适当修整操场，以获得更好的训练效果。

具体地，军训共有 q 天，每天会有一次训练或一次操场修整发生：

训练：

由校长希德指定一个起始格子 s 和一个训练时长 t ，训练过程如下：

- 学生们的起始速度 v 为 1。
- 军训开始时，所有学生**进入** s 号格子。
- 通过一个格子后，学生们会离开当前格子，进入**下一个**格子。
- t 时长后训练结束，学生们立刻停止移动。

操场修整：

校长会指定数对 (l, r) 并进行以下两种修整之一：

- 拓宽：对编号在 $l, (l + 1) \bmod n, \dots, r$ 的所有格子，其训练值**加上** Δ 。
- 铺平：对编号在 $l, (l + 1) \bmod n, \dots, r$ 的所有格子，其训练值**变为** d' 。

按照惯例，校长希德会提前走到终点所在的格子迎接学生。对于每次训练，他希望你告诉他他应该到哪个格子去。

注意：这里我们将两个格子的边界视为属于**下一个**格子，即：若 t 时间后学生们恰好离开 i 号格子，则终点所在的格子编号为 $(i + 1) \bmod n$ 。

输入格式

从文件 `training.in` 输入。

第一行两个正整数 n, q 。

第二行 n 个正整数，第 i 个正整数表示 d_{i-1} 。

接下来 q 行，每行为以下三种之一：

- $P\ l\ r\ \Delta$ ：以增加值 Δ 拓宽第 $l \sim r$ 号格子。
- $R\ l\ r\ d'$ ：以训练值 d' 铺平第 $l \sim r$ 号格子。
- $Q\ s\ t$ ：开展一次起始格子为 s ，时长为 t 的训练。

输出格式

输出到文件 `training.out`。

每行一个非负整数代表每次训练的终点编号。

样例 1

输入

```
2 4
1 2
Q 0 0
Q 0 1
Q 0 4
Q 0 5
```

输出

```
0
1
1
0
```

解释

训练从 0 号格子开始：

- $t = 0$ 时位于 0 号格子；
- $t = 1$ 时 **恰好离开** 0 号格子进入 1 号格子，故此时终点为 1 号格子；
- 进入 1 号格子后，速度 v 变为 $\frac{1}{2}$ ，花费 4 的时间匀速通过 1 号格子；

- $t = 5$ 时 恰好离开 1 号格子重新进入 0 号格子，此时终点为 0 号格子。

样例 2

输入

```
5 7
2 5 1 4 3
Q 3 28
Q 0 39
P 4 0 3
Q 2 60
R 4 1 2
Q 1 22
Q 3 19260817
```

输出

```
0
3
0
4
1
```

样例 3~6

见下发文件中的 training*.in/out，分别满足第 1~4 个子任务的数据范围。

数据范围

对于所有数据： $2 \leq n, q \leq 2 \times 10^5$ ， $1 \leq d_i, \Delta, d' \leq 10^6$ ， $0 \leq l, r, s < n$ ， $0 \leq t \leq 2 \times 10^{17}$ 。保证任意时刻 $\forall i \in [0, n), d_i \leq 10^6$ 。

子任务编号	$n \leq$	$q \leq$	特殊性质	分值
1	5000	5000	无	10
2	2×10^5	2×10^5	A	25
3	10^5	10^5	无	35
4	2×10^5	2×10^5	无	30

特殊性质 A: 保证没有 P, R 事件。