

【2025SD省队第三轮(济南)集训第5场】

2025.5.30

题目名称	绝对正常	互相抵消	基本无害
题目类型	传统型	传统型	传统型
输入文件名	a.in	b.in	c.in
输出文件名	a.out	b.out	c.out
时间限制	1s	1s	3s
内存限制	512 MiB	512 MiB	1024 MiB
子任务数目	10	20	7
子任务是否等分	是	是	否

注意事项

- 文件名（包括程序名和输入输出文件名）必须使用英文小写。
- 编译选项：`-lm -std=c++14`。
- C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 `0`。
- 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格分隔。
- 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
- 每道题目的栈空间与所给的空间限制一致。
- 原则上，每个测试点时限为标准程序在该测试点上的运行时间的 2 倍及以上。

绝对正常 (a)

题目描述

给定一棵 n 个节点的叶向树，根节点为 r ，叶向树即该树由 $n - 1$ 条有向边构成，每条有向边指向根节点所在的另一端，经过该边所需要的时间均为 1。

接下来，将有 n 个人按照编号从小到大的顺序依次空降到根节点 r ，第 i 个人的目的是到节点 i 完成任务，因此人 i 将沿着叶向树上的唯一路径到达点 i ，随后在点 i 上停留 a_i 单位时间执行任务，之后人 i 即完成任务。由于有向边的“特殊性”，人们不能在有向边上停留，且仅能在节点上停留整数单位时间。

由于任务的“特殊性”，这些人在进行任务的过程中不能相互干扰，即，**正在执行任务的人**（不包括已经完成任务的人）不能同一时刻在同一个节点。也就是说，当人 i 在点 i 处正在执行任务时，若人 j 抵达点 i 的父节点且点 i 是点 j 的必经节点，则人 j 必须在原位等待直到人 i 任务完成，若人 i 在时刻 t 完成任务，则人 j 在时刻 t 可以沿有向边出发并在时刻 $t + 1$ 抵达点 i 。类似的，若人 j 抵达某点后发现下一个必经节点处有人 i 在等待，则人 j 必须等待人 i 离开该点后才能移动至该点，即若人 i 在时刻 t 决定移动并于时刻 $t + 1$ 移动到了其他节点，则人 j 同样可以在时刻 t 沿有向边出发并在时刻 $t + 1$ 到达人 i 在时刻 t 所在的节点。空降过程同理，若人 i 发现人 $i - 1$ 由于堵塞而停留在出发点 r ，则人 i 不会空降（施展魔法在空中悬浮），直到人 $i - 1$ 离开点 r 。

指挥官希望这 n 个人能够在最短时间内完成任务，询问在采用最优策略的情况下这 n 个人都完成任务所需要的最少单位时间。

时间限制 1 秒，空间限制 512 MB。

输入格式

输入的第一行包含两个正整数 n, r ，表示树的点数和根节点编号。

输入的第二行包含 n 个整数 a_1, a_2, \dots, a_n ，表示这 n 个人停留在对应节点上执行任务所需要的时间。

接下来 $n - 1$ 行每行包含两个正整数 u, v ，描述树上的一条边 (u, v) 。

输出格式

输出的第一行包含一个整数，表示答案。

数据范围

对于 100% 的数据，保证 $1 \leq n \leq 10^5$ ， $0 \leq a_i \leq 10^9$ 。

测试点编号	$n \leq$	特殊性质
1 ~ 2	100	无
3 ~ 4	1000	无
5 ~ 6	10^5	A
7 ~ 8	10^5	B
9 ~ 10	10^5	无

特殊性质 A：保证给定的树形态为一条链，且出发点 r 为该链的一个端点。

特殊性质 B：保证所有人完成任务所需要的时间均相等，即 $a_1 = a_2 = \cdots = a_n$ 。

互相抵消 (b)

题目描述

给定一个长度为 n 的序列，接下来进行 q 次操作，每次操作形如：

- 操作一：给定 l, r, x ，将区间 $[l, r]$ 内的每个 $a_i \leftarrow a_i + x$ 。
- 操作二：给定 l, r ，查询 $\sum_{l'=l}^r \sum_{r'=l'}^r ((\sum_{i=l'}^{r'} a_i)^2 + (r-l+2) \times (r'-l') \times a_{l'} \times a_{r'})$ 。

对于所有操作二，给出对应的答案，对 998244353 取模。

时间限制 1 秒，空间限制 512 MB。

输入格式

本题开启强制在线。

输入的第一行包含一个整数 type ，表示强制在线参数。

输入的第二行包含两个整数 n, q ，表示序列长度和操作次数。

接下来一行包含 n 个整数 a_1, a_2, \dots, a_n ，表示初始序列形态。

接下来 q 行，每行首先输入一个整数 op ，若为 1 则再输入三个整数 l, r, x 表示操作一；若为 2 则再输入两个整数 l, r 表示操作二。由于本题强制在线，不妨设上一次操作二的答案为 lastans （初始为 0），对于操作一，需要将 l, r, x 均异或上 $(\text{lastans} \times \text{type})$ 得到真实的 l, r, x ，对于操作二，需要将 l, r 均异或上 $(\text{lastans} \times \text{type})$ 得到真实的 l, r 。（注意 lastans 是取模之后的结果）

输出格式

输出包含若干行，对于每组操作二，输出本次询问的答案，对 998244353 取模。

数据范围

对于 100% 的数据，保证 $\text{type} \in [0, 1]$ ， $1 \leq n \leq 5 \times 10^5$ ， $0 \leq a_i, x < 998244353$ 。

测试点编号	$n \leq$	$q \leq$	特殊性质
1 ~ 2	600	600	AB
3 ~ 4	5×10^3	5×10^3	AB
5 ~ 6	10^5	10^5	A
7 ~ 8	5×10^5	5×10^5	A
9	10^5	10^5	C
10 ~ 11	5×10^5	5×10^5	C
12	10^5	10^5	D
13 ~ 14	5×10^5	5×10^5	D
15 ~ 16	10^5	10^5	无
17 ~ 18	3×10^5	3×10^5	无
19 ~ 20	5×10^5	5×10^5	无

特殊性质 A：保证 $\text{type} = 0$ 。

特殊性质 B：保证询问为操作一的概率为 $3/4$ ，操作二的概率为 $1/4$ 。

特殊性质 C：保证仅存在至多一个操作二。

特殊性质 D：保证不存在操作一。

基本无害 (c)

题目描述

有一个无限大的二维网格图，我们计划在网格图上建造 n 座塔，第 i 座塔的所在格子的坐标为 (a_i, b_i) 。接下来我们将尝试着规划出一种建造的顺序依次建塔。不妨设我们建塔的顺序为长度为 n 的排列 p_1, p_2, \dots, p_n ，由于一些特殊的原因，建塔顺序 p 需要满足一些特定的条件。

首先，由于政府需要让人们确信建塔的工程是有组织有计划的，政府需要确保自己的建塔的过程显得很有规划。具体来说，建造的第一座塔可以是任何一座塔，但是对于之后建造的所有塔，例如在建造第 i 座塔 ($2 \leq i \leq n$) 时，需要确保该塔与前 $i - 1$ 座已经建造的塔有着一定的联系，进一步的，我们要求前 $i - 1$ 座塔中存在至少一座塔与该塔在网格图中**有公共边或公共点**，这一要求对第一座之后建造的所有塔均有效。

其次，尽管网格图内幅员辽阔、物资充沛，但在建造一座塔时，若该塔被四周的若干塔包围，则建造工作仍然会受到极大影响（外围边界的物资受到其他塔的阻隔无法运输进来），为了保证计划的正常进行，我们要求当正在建造第 i 座塔时，确保第 i 座塔所在的格子满足以该塔作为起点，之后可以移动到**有公共边**的相邻空格子（要求移动到的目标格子上没有已经完成建造的塔），并通过这种公共边移动的方式可以移动到网格的无限远处（显然无限远处是互相连通的）。

最后，出于各种原因，人们会对新建造的塔显得更加记忆深刻，越新建造的塔如果编号越大，则越对人们有着吸引力，因此政府**可能**希望我们规划出的顺序排列 p 满足 $\{p_n, p_{n-1}, p_{n-2}, \dots, p_1\}$ 的字典序最大化。当然，这在一定程度上取决于政府对此事的关心程度，因此**有时**我们对此不做要求，规划任意一种合法顺序即可。

时间限制 3 秒，空间限制 1024 MB。

输入格式

输入的第一行包含一个整数 n ，表示网格中计划建造的塔的数量。

输入的第二行包含一个整数 type ，表示规划类型，若 $\text{type} = 1$ 则构造任意一组合法顺序；若 $\text{type} = 2$ 则构造满足 $\{p_n, p_{n-1}, p_{n-2}, \dots, p_1\}$ 字典序最大的合法顺序。

接下来 n 行，每行两个整数 a_i, b_i ，表示第 i 座塔在网格图中的所在坐标 (a_i, b_i) 。

输出格式

输出的第一行一个字符串，若不存在合法顺序，则输出“NO”，结束。否则输出“YES”，继续构造。

接下来输出包含 n 行，第 i 行表示在计划中建造的第 i 座塔在输入中的编号。我们将根据 type 类型进行评分。

数据范围

本题开启子任务评测。

对于所有数据，保证 $1 \leq n \leq 1.5 \times 10^5$ ， $\text{type} \in [1, 2]$ ，且 $-10^9 \leq a_i, b_i \leq 10^9$ 。

子任务编号	$n \leq$	type =	特殊性质	子任务分值
1	10	1		10
2	200	1		20
3	2×10^3	1		10
4	2×10^3	2		10
5	1.5×10^5	1		15
6	7×10^4	2	A	15
7	1.5×10^5	2		20

特殊性质 A: 保证 $-10^3 \leq a_i, b_i \leq 10^3$ 。