

Problem A. 深度自同构

Input file:standard input

Output file:standard output

Time limit:1 second

Memory limit:256 megabytes

对于无向图中的点，定义一个点的度为与其相连的边的条数。

对于一棵有根树，定义一个点的深度为该点到根的距离。

对于由若干有根树构成的森林，定义该森林是深度自同构的，当且仅当森林中任意两个深度相同的节点都有相同的度。

请计数有多少个深度自同构的无编号有根树森林，使得其恰好由 n 个节点构成。

答案对 998244353 取模。

Input

一个整数 N ($1 \leq N \leq 10^6$)，代表计数的范围。

Output

输出一行 N 个整数，第 i 个数代表 $n = i$ 时的答案，对 998244353 取模。

Example

standard input	standard output
5	1 2 3 5 6

Problem B. 旅行

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 512 megabytes

有一棵 n 个结点的无根树，每个结点都有对应的类型 c_i 和权重 w_i 。
你需要在这棵树上规划若干次旅行。对于一次旅行，你将从一个树上的一个结点出发，沿着树上的边进行旅行，最终到达另一个和起点类型相同的结点。
你会进行很多次旅行，但你希望对于每个结点，在所有旅行路线中最多只会经过一次。
一次旅行的价值是起始点和终止点的权重和，你需要规划旅行的方案使得旅行的总权重和最大。

Input

第一行输入一个正整数 t ($1 \leq t \leq 3$) 代表数据组数。
对于每组数据：
第一行输入一行一个正整数 n ($1 \leq n \leq 2 \times 10^5$)，代表树的点数。
第二行输入一行 n 个正整数 c_i ($1 \leq c_i \leq n$)，代表每个结点的类型。
第三行输入一行 n 个正整数 w_i ($1 \leq w_i \leq n$)，代表每个结点的权重。
接下来 $n - 1$ 行每行两个正整数 u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$)，代表树上一条边，输入保证是一棵树。

Output

输出一行一个正整数代表答案。

Example

standard input	standard output
1 7 3 1 1 2 2 2 3 2 4 1 5 4 6 2 1 2 1 3 2 4 2 5 3 6 3 7	13

Note

一种最优的旅行方案为：
旅行路线1: $4 \rightarrow 2 \rightarrow 5$ ，价值为 9。
旅行路线2: $1 \rightarrow 3 \rightarrow 7$ ，价值为 4。
价值总和为 13。

Problem C. 游走

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 256 megabytes

一条街道上有 n 个路灯排成一排，编号 $1, 2, \dots, n$ 。一个酒鬼初始在时刻 0 时站在 1 号路灯旁。
酒鬼从某个时间 t_0 ($t_0 \geq 0$) 开始游走。在时间 t_0 之后，酒鬼每隔一个单位时间就会移动到相邻的路灯旁。如果酒鬼在时间 t 在 i 号路灯旁，则他在时间 $t + 1$ 必然移动到 $i - 1$ 号路灯或 $i + 1$ 号路灯旁。特殊的，酒鬼不会移动到街道边界之外，即他始终只停留在路灯 1 和 n 之间（包含边界）。例如，酒鬼在时间 $t_0 + 1$ 必然在 2 号路灯旁。
你得知了一些路人提供的信息，每条信息都可以用整数 p_i 和 q_i 表示，代表在 q_i 时刻某位路人看到到酒鬼在路灯 p_i 旁边。你想找到酒鬼开始到处走动的时间 t_0 。在收到信息的同时，你还想根据当前收到的信息推断 t_0 可能的最大值和最小值。
路人提供的信息也有可能不一致。一旦你发现提供的信息有矛盾，只需停止计算并报告信息不一致。

Input

第一行一个整数 t ($1 \leq t \leq 100$)，代表数据组数。对于每组数据：
第一行包含两个整数 n, m ($2 \leq n \leq 10^9, 1 \leq m \leq 2 \times 10^5$)，代表街道上路灯的数量和操作次数。
接下来 m 行，每行描述一个操作,为以下三种类型之一：

- 0 $p_i q_i$: 路人在时间 q_i 看到酒鬼在路灯 p_i 旁边 ($1 \leq p_i \leq n, 0 \leq q_i \leq 10^9$)。
- 1: 根据当前收到的信息推断， t_0 可能的最小值。
- 2: 根据当前收到的信息推断， t_0 可能的最大值。

保证所有数据的 m 之和不会超过 5×10^6 。

Output

对于每个询问，输出一行代表询问的答案。
对于 2 询问（即询问 t_0 可能的最大值），如果 t_0 可以为任意大，输出 `inf`。
如果当前收到的信息已经不一致，对于后续的所有询问均输出 `bad`。

Example

standard input	standard output
1	0
11 9	inf
1	3
2	1
0 3 5	bad
2	bad
0 1 3	
1	
0 10 6	
2	
1	

Problem D. 游戏

Input file: standard input
Output file: standard output
Time limit: 3 seconds
Memory limit: 512 megabytes

有 n 名玩家进行游戏，每个人有一个初始能力值 a_i 。
游戏会进行 t 轮，每一轮等概率随机选择两个不同的人将他们的能力值分别加一。
求游戏结束后 $\sum_{i=1}^{n-1} \sum_{j=i+1}^n [a_i = a_j]$ 的期望，答案对998244353取模。

Input

第一行两个正整数 n, t ($2 \leq n \leq 10^6, 1 \leq t \leq 10^7$)。
第二行 n 个正整数代表 a_i ($1 \leq a_i \leq 10^6$)。

Output

一行一个整数，代表答案对 998244353 取模后的值。

Example

standard input	standard output
3 2 1 2 3	221832079

Problem E. 数论

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

给定长度为 n 的正整数数列 a_1, a_2, \dots, a_n 。

定义不交区间集为若干不交的区间 $[l_1, r_1], [l_2, r_2], \dots, [l_k, r_k]$ 的集合，其中任意集合 $[l_i, r_i]$ 满足 $1 \leq l_i \leq r_i \leq n$ 。我们称两个不交区间集不同，当且仅当这两个区间的集合不同。

我们称一个不交区间集是好的，当且仅当

$$\gcd_{i \in [l_1, r_1]} \{a_i\} = \gcd_{i \in [l_2, r_2]} \{a_i\} = \dots = \gcd_{i \in [l_k, r_k]} \{a_i\}$$

请你对每个 a_i 求出，有多少个**好的不交区间集**，会将其选入在某一个区间中。

Input

第一行一个整数 n ($1 \leq n \leq 10^5$)，代表数列的长度。

第二行 n 个整数， a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$)。

Output

输出一行 n 个整数，第 i 个数表示有几个**好的不交区间集**，会将 a_i 其选入在某一个区间中。

由于答案很大，输出的数字对 998244353 取模。

Example

standard input	standard output
6 3 6 12 2 4 1	9 13 15 15 12 9

Problem F. 字符串

Input file:standard input

Output file:standard output

Time limit:4 seconds

Memory limit:512 megabytes

给定由 ‘a’, ‘b’ 两种小写字母组成的串 s , 下标从 1 开始。
定义 $occ(t)$ 表示字符串 t 在 s 中出现的次数。

有 q 次询问, 每次询问有如下两种类型:

1. 操作1, 给定 l, r , 询问有多少本质不同的串 t , 满足 $s[l, r]$ 是 t 的子串, 且 $occ(s[l, r]) = occ(t)$
2. 操作2, 给定 l, r , 询问有多少本质不同的串 t , 满足 t 是 $s[l, r]$ 的子串, 且 $occ(s[l, r]) = occ(t)$

Input

第一行一个整数 t ($1 \leq t \leq 5$) 代表数据组数。对于每组数据:
第一行输入一个串 s ($1 \leq |s| \leq 5 \times 10^5$)。
第二行一个整数 q ($1 \leq q \leq 5 \times 10^5$) 表示有 q 次询问。
接下来 q 行每行三个数 op, l_0, r_0 ($op \in \{1, 2\}, 1 \leq l_0, r_0 \leq |s|$), 表示询问的类型和询问的区间 $[l, r]$ 。
其中 $l = (l_0 + lastans - 1) \% |s| + 1, r = (r_0 + lastans - 1) \% |s| + 1$, 保证 $1 \leq l \leq r \leq |s|$ 。
 $lastans$ 代表上一次询问的答案, 特别的对于每组数据的第一次询问时 $lastans = 0$ 。
保证所有测试数据 $\sum |s| \leq 5 \times 10^5, \sum q \leq 5 \times 10^5$ 。

Output

对于每组数据输出 q 行, 每行一个整数表示每个询问的答案。

Example

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

Problem G. 单峰数列

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

对于一个整数数列，如果其先严格递增，然后在某一点后严格递减，我们称这个数列为单峰数列（严格递增和严格递减的部分均要是非空）。

给定长度为 n 的整数数列 a_1, a_2, \dots, a_n ，请你支持 q 次操作：

- 1. 1 l r x: 将 a_l, a_{l+1}, \dots, a_r 的每个数加 x 。
- 2. 2 l r: 判断 a_l, a_{l+1}, \dots, a_r 的元素是否全都相同。
- 3. 3 l r: 判断 a_l, a_{l+1}, \dots, a_r 是否严格升序排序。当 $l = r$ 时，认为符合严格升序排序。
- 4. 4 l r: 判断 a_l, a_{l+1}, \dots, a_r 是否严格降序排序。当 $l = r$ 时，认为符合严格降序排序。
- 5. 5 l r: 判断 a_l, a_{l+1}, \dots, a_r 是否为单峰数列。保证 $r - l + 1 \geq 3$ 。

Input

第一行输入包含一个整数 n ($3 \leq n \leq 10^5$)。
第二行输入包含 n 个整数 a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^9$)。
第三行输入包含一个整数 q ($1 \leq q \leq 2 \times 10^5$)。
接下来的 q 行，每行描述一个操作，格式见题目描述。对于第一类操作，保证 $-10^9 \leq x \leq 10^9$ 。

Output

对于每个询问输出一行一个整数，如果查询符合要求输出 1，否则输出 0。

Example

standard input	standard output
7	0
3 1 4 1 5 9 2	1
8	1
5 1 3	0
5 4 7	1
1 3 3 -3	1
2 2 4	1
3 3 5	
3 4 5	
4 1 1	
4 1 2	

Problem H. 比特跳跃

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

比特国由 n 个城市组成，编号为 $1, 2, \dots, n$ 。
有 m 条双向道路连接这些城市，第 i 条连通城市 u_i 和城市 v_i ，通过这条道路需要花费 t_i 的时间。
此外，比特国的人们还可以使用“比特跳跃”来通行于任意两个城市之间。
从城市 x 通过比特跳跃移动到城市 y 需要花费 $k \times (x|y)$ 的时间，其中 $|$ 表示按位或。比特跳跃可以使用任意多次。
现在请你计算出，从 1 号城市移动到每个城市所需的最短时间。

Input

第一行一个整数 t ($1 \leq t \leq 15$)，代表数据组数。对于每组数据：
第一行三个整数 n, m, k ($2 \leq n \leq 10^5, 0 \leq m \leq 10^5, 0 \leq k \leq 10^6$)，代表城市个数，道路条数，比特跳跃的系数。
接下来 m 行，每行三个整数 u_i, v_i, t_i ($1 \leq u_i, v_i \leq n, 0 \leq t_i \leq 10^9$)，代表一条道路的信息。
保证所有测试数据的 $\sum n \leq 10^6, \sum m \leq 10^6$ 。

Output

对于每组数据，输出一行 $n - 1$ 个整数，代表从 1 号城市移动到编号为 $2, 3, \dots, n$ 的城市所需的最短时间。

Example

standard input	standard output
1 6 4 3 1 3 2 1 5 20 2 4 1 4 6 10	9 2 10 15 20

Problem I. 圣芙蕾雅

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

为了对抗崩坏，圣芙蕾雅学院培养了 k 位女武神（从 1 到 k 编号），这些女武神参加了一系列战斗。现共有 n 次事件记录（从 1 到 n 编号），共有两种类型的记录，第 1 种为装甲分配记录，包含 1 位女武神的编号，表示分配给该女武神 1 套装甲；第 2 种为战斗记录，包含 1 位女武神的编号，表示该女武神参加了这次战斗。受到崩坏的影响，事件记录的先后顺序发生了错乱，只能确保 1 号事件是最早发生的，同时有 m 条可能的先后关系。

简单来说，可以把 n 次事件看成 n 个点， m 条先后关系看成 m 条有向边；1 条由 x 连向 y 的有向边表示 x 事件的下一次事件可能是 y ，没有边连向 1 号点。注意，图可能有环，即同一事件可能多次发生。

装甲归还的记录已经丢失，现已知每位女武神至多只会被分配 1 次装甲，且每次参加战斗时，必须已经被分配装甲且未归还。一套装甲在分配给一位女武神且未归还的情况下不能分配给其他女武神，但是在归还后可以分配给其他女武神。现在希望你尝试还原装甲归还记录，但是由于信息不足，无法真正还原，因此你只需要计算圣芙蕾雅学院至少拥有多少套装甲，才有可能满足所有要求。你可以将装甲归还事件添加在任意事件的前后，且添加次数不限，只要保证在每个战斗事件时，该女武神一定已分配装甲且未归还即可。

Input

第一行输入一个正整数 t ($1 \leq t \leq 5$) 代表数据组数。对于每组数据：

第一行 3 个正整数 n, m, k ($1 \leq n, k \leq 3 \times 10^3, 1 \leq m \leq 10^4$)，分别表示事件的数量，可能的先后关系的数量和女武神的数量。

接下来共 n 行，其中第 i 行 2 个正整数 opt_i, x_i ($opt_i \in \{1, 2\}, 1 \leq x_i \leq k$):

- 如果 $opt_i = 1$ 则表示事件为给 x_i 女武神分配 1 套装甲。
- 如果 $opt_i = 2$ 则表示 x_i 女武神参加了战斗。

接下来共 m 行，其中第 i 行 2 个正整数 u, v ($1 \leq u, v \leq n, u \neq v$) 表示 u 事件的下一事件可能是 v 事件。

数据保证任意事件都有可能发生，保证不会出现战斗事件发生时，参战女武神还未被分配装甲。

Output

对每组数据输出 1 行 1 个整数，表示圣芙蕾雅学园至少拥有的装甲数量。

Example

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

Problem J. 绘世之卷

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 512 megabytes

「旭光」之后，便是万千「繁星」

格蕾修正在画室里画画，她一共有 n 种颜色（从 1 到 n 编号）的颜料。

一开始，画布上什么颜色也没有，接下来格蕾修进行了 q 次操作，操作共有两种：

- 1.在画布上添加一种原本没有的颜色；
- 2.在画布上擦去一种原本存在的颜色。

定义两种不同的颜色 x 关于 y 的和谐度为 x 除以 y 的商加上余数，即 $\lfloor \frac{x}{y} \rfloor + x - \lfloor \frac{x}{y} \rfloor \times y$ ；定义整张画布的和谐度为从中任选两种存在的不同颜色得到的和谐度的最小值。

请你计算格蕾修每次操作完成后画布的和谐度，如果画布中存在的颜色不足 2 种，则输出 -1 。

Input

第一行输入一个正整数 t ($1 \leq t \leq 5$) 代表数据组数。

对于每组测试数据：

第一行 2 个正整数 n, q ($1 \leq n, q \leq 5 \times 10^4$)，分别表示颜色的数量和格蕾修操作的次数。

接下来共 q 行，其中第 i 行 2 个整数 opt_i, x_i ($opt_i \in \{0, 1\}, 1 \leq x_i \leq n$)。

如果 $opt_i = 0$ 则表示在画布上擦去颜色 x_i ，保证操作前画布上存在颜色 x_i ；

如果 $opt_i = 1$ 则表示在画布上添加颜色 x_i ，保证操作前画布上不存在颜色 x_i 。

Output

共 q 行，其中第 i 行输出 1 个整数表示格蕾修第 i 次操作后画布的和谐度。

Example

standard input	standard output
1	-1
10 4	4
1 4	3
1 7	3
1 9	
0 7	

Problem K. 抓拍

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

学校里有 n 名同学，初始时第 i 位同学从 (x_i, y_i) 出发，以每秒 1 米的速度散步。
同学们散步的方向有东南西北四种可能。假设有以为同学正在位于 $(0, 0)$ ，则下一秒

- 如果向东走，到达 $(1, 0)$
- 如果向西走，到达 $(-1, 0)$
- 如果向南走，到达 $(0, -1)$
- 如果向北走，到达 $(0, 1)$

假设散步过程会进行无限长的时间，同学们散步的方向不会改变，并且忽略碰撞的情况（允许某个时刻多人在同一个点，互不影响）。

现在你可以选择某个**非负整数**秒时刻抓拍照片。

一张照片可以用长方形 $((e, n), (w, s))$ 表示，东北角为 (e, n) ，西南角为 (w, s) 。

只有抓拍的照片包含了所有同学时，我们才称这张照片是完美的。

请选择某个时刻抓拍一张完美的照片，使得照片的周长最小。

Input

第一行一个整数 n ($1 \leq n \leq 2 \times 10^5$)，代表人数。

接下来 n 行，第 i 行包含两个整数 x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$) 和一个字符 d_i ，由空格分隔，描述第 i 位同学。 d_i 是 'E', 'W', 'S', 'N' 之一，分别代表第 i 位同学散步的方向是东、西、南、北。

Output

输出一行一个整数，代表最短的完美照片的周长。

Example

standard input	standard output
5 0 2 E 0 6 S 2 0 N 2 6 S 4 4 W	8

Problem L. 死亡之组

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

有 n 个队伍参与了世界杯，编号为 $1, 2, \dots, n$ 。第 i 号队伍实力用一个整数 a_i 描述。
现在所有队伍被划分为若干小组，每个小组由四个队伍构成，每个队伍恰好属于一个小组。
我们称一个小组是"死亡之组"，当其至少符合以下条件之一：
1. 小组中存在多个球队的实力都相当出色，有至少两支队伍的实力不小于 L 。
2. 四支球队水平极其接近，小组中最高实力队伍和最低实力队伍的实力之差不超过 D 。
现在假设你可以决定分组，问是否存在一种分组方案，使得 1 号队伍所在的组不是"死亡之组"。

Input

第一行一个整数 t ($1 \leq t \leq 20$)，代表数据组数。
对于每组数据：
第一行 3 个整数 n, L, D ($1 \leq n \leq 10^5, 1 \leq L, D \leq 10^9$)，保证 n 是 4 的倍数。
第二行 n 个整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$)，代表 n 个队伍的实力。

Output

对于每组数据输出一行，如果可以构造出一种符合要求的方案，输出 ‘Yes’，否则输出 ‘No’。

Example

standard input	standard output
1 8 6 4 1 2 3 4 5 6 7 8	Yes

Note

可以选择 1, 3, 5, 7 组成一个小组。