

Problem A. csl 找妈妈 III

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

众所周知, *csl* 喜欢找 妈妈, 妈妈喜欢玩超级马里奥。

于是 *csl* 决定帮 妈妈速通超级马里奥, 但是早期的超级马里奥色彩十分单调, 只有黑白两种颜色, 于是 *csl* 决定自己给超级马里奥上色 (显然只能上各种灰色)。

全能的 *csl* 发现他能通过每种灰色的两个属性: 黑度和白度 (都是 $0\% - 100\%$), 相互组合调配出其他灰色。

比如: *csl* 有一个 $(20\%, 20\%)$ 的灰色 1 和 $(10\%, 30\%)$ 的灰色 2。那么 *csl* 就可以用 50% 的灰色 1 和 50% 的灰色 2 调配出 $(15\%, 25\%)$ 的灰色。也可以用 10% 的灰色 1 和 90% 的灰色 2 调配出 $(11\%, 29\%)$ 的灰色。

严格来说, 假设你有 (a_1, b_1) 和 (a_2, b_2) , 那么 *csl* 能得到 $(k \times a_1 + (1-k) \times a_2, k \times b_1 + (1-k) \times b_2), 0 \leq k \leq 1$ 。

现在他想知道, 给出 n 种灰色, 他能不能通过这些灰色, 调配出他想要的灰色?

【注意】 $(50\%, 50\%)$ 和 $(20\%, 20\%)$ 是不一样的灰色!

Input

第一行 T 表示 $T(1 \leq T \leq 5)$ 组数据。

每组数据第一行 $n(1 \leq n \leq 10000), m(1 \leq m \leq 100)$ 。

n 代表 *csl* 有 n 种灰色, m 代表询问次数。

接下来的 n 行每行两个整数 $a, b(0 \leq a, b \leq 100)$, 分别代表 *csl* 拥有灰色的黑度和白度。

接下来的 m 行每行两个整数 $a, b(0 \leq a, b \leq 100)$, 代表 *csl* 查询他是否能调配出这种颜色。

输入给出的颜色中可能会有相同的颜色。

Output

对于每一次询问, 如果能调配出, 输出 Yes, 否则输出 No。

Examples

standard input	standard output
1	Yes
2 2	No
10 10	
20 20	
15 15	
30 30	
1	Yes
3 2	Yes
0 0	
100 0	
50 100	
50 0	
25 50	

Problem B. compute 的取石子游戏

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

众所周知, 在程序员中盛行着一种取石子游戏, compute 对此很感兴趣, 于是拉来了 lin88 一起玩这个游戏, 游戏规则是这样的:

有一堆石子, 共 n 个, 每个人轮流从中取出若干个石子, 无法行动的人就输了。

compute 觉得这样的规则过于无聊, 于是他决定让这个游戏更好玩一点, 每局游戏前, 两人先选定一个数字 x , 表示取的石子的个数只能是 x 的幂次, 如 $x = 2$ 时, 两人每次只能从中取走 1, 2, 4, 8, ... 个石子, 当然, 是不能取走超过石子实际数量的石子数的。

现在知道他们选择的数 x , 和石子的数量 n , compute 慷慨的把先手的机会交给了 lin88, lin88 想知道, 在她们都按最优策略行动时, 她是否能赢。

Input

两个数 $x(1 \leq x \leq 10^5)$, $n(1 \leq n \leq 5 \times 10^6)$ 分别表示他们选择的数和石子的数量。

Output

如果 lin88 能赢, 输出"YES" (不含引号), 否则输出"NO" (不含引号)。

Examples

standard input	standard output
1 10	NO
2 8	YES

Problem C. compute 的小区规划

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

众所周知，散步是人类的一大爱好。

compute 最近接到了一份小区规划的工作，工作的主要内容是设计小区内的道路。

小区内有 n 个路口和 m 条可选的道路，第 i 条道路连接了 u_i 和 v_i ，且通过这条道路的时间为 w_i 。

作为一个散步爱好者，她希望小区道路的复杂度越大越好，这使她能走的足够长，当然小区内必须要是联通的。

复杂度被定义为 $\sum_{i=1}^n \sum_{j=i+1}^n \text{dis}(i, j)$ ， $\text{dis}(i, j)$ 表示 i 到 j 之间的最短路。

现在给定可供选择的道路，请你求出最大的复杂度。

Input

第一行两个整数 n, m ($1 \leq n \leq 14, 1 \leq m \leq \frac{n \times (n-1)}{2}$)，分别表示小区内的路口数量和可供选择的道路数量。

接下来 m 行，每行 3 个整数 u_i, v_i, w_i ($1 \leq u_i, v_i \leq n, 1 \leq w_i \leq 10^9$)，描述了一条道路。

数据保证没有重边和自环且图能够联通。

Output

输出一个整数表示小区内最大可能的复杂度。

Examples

standard input	standard output
5 5 1 2 1 1 3 1 2 4 1 2 5 1 1 5 1	20
5 10 1 2 1 1 3 2 1 4 3 1 5 4 2 3 5 2 4 6 2 5 7 3 4 8 3 5 9 4 5 10	146

Problem D. compute 的树

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

众所周知，每一个优秀的程序猿跟树都是分不开的。作为一个程序猿，compute 也非常喜欢树。所以 compute 想让你画一棵树送给她。

你需要在二维平面上画出这棵树，使得任意两条树边除了端点以外都不相交。

Input

第一行一个整数 n ($1 \leq n \leq 1000$) 表示这棵树的点数。

接下来 $n - 1$ 行，每行两个数 u, v ($1 \leq u, v \leq n$)，表示这棵树的 $n - 1$ 条边。

Output

输出 n 行，每行输出两个整数 x_i, y_i 表示 i 号点的坐标。

你可以输出任何一种符合要求的答案，但要求 $0 \leq x_i, y_i \leq 10^9$ ，且 $\forall i \neq j, x_i \neq x_j \wedge y_i \neq y_j$ ，即每个点的坐标两两不同。

Examples

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

Problem E. lin88 的套娃

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

众所周知, lin88 前不久去了趟俄罗斯。lin88 想既然来了一定要找一点有趣的东西, 到了特产店, lin88 对俄罗斯套娃产生了极大的兴趣!

这里的套娃各式各样, 但奈何背包的容量有限, lin88 只能带一个, 而 lin88 想尽可能多带。众所周知套娃是可以套起来的, 这让 lin88 用同样的空间带了更多的套娃。

现在的难题是, 他想将一些娃套成一个带走, 在尽可能多带的前提下, 有多少种不同的带法, lin88 想请你帮帮他。

每个套娃都是独一无二的, 两种带法不一样的意思是有一个娃, 在一种带法中而不在另一种。

每个套娃都有一个外径 out 和一个内径 in , 且 $out > in$, 娃 a 可套在另一个娃 b 中当且仅当 $out_a \leq in_b$ 。

尽可能多带的意思是, 这种带法中再也不能加入任何一个其他的娃。

lin88 要你告诉他总共的带法数量, 但带法数量可能太多, 你只要输出答案对 998244353 取模后的结果即可。

Input

第一行一个整数 $n(1 \leq n \leq 2 \times 10^5)$ 代表套娃的个数。

接下来的 n 行每行两个整数 out_i 和 $in_i(1 \leq in_i < out_i \leq 10^9)$ 分别代表第 i 个套娃的外径和内径。

Output

输出一行一个整数, 代表带法数量对模 998244353 后的结果。

Examples

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

Note

第二个样例中, 所有的带法是: $\{1, 5\}, \{1, 6\}, \{2, 4, 5\}, \{2, 4, 6\}, \{3, 4, 5\}, \{3, 4, 6\}, \{4, 6, 7\}, \{4, 5, 7\}$, 数字代表编号。

Problem F. lin88 的高铁站

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

众所周知, compute 日常去散步, 最便捷的交通方式莫过于高铁了。compute 都是心血来潮就出发, 可是说走就走说得容易, 票要买得到才行。compute 时常买不到票, lin88 决定来帮他解决这个问题: 再开一条线, 但线路的运营需要管理啊, 可是 lin88 还得忙于训练, 抽不出足够的时间打理这条线路。现在他想用一个程序来完成, 但写程序也不是件易事, 所以他想求助于你。

简单来说是这样的, 这条线上有 n 个站, 编号 $1 \sim n$, 每天第 i 个站有 a_i 个乘客, 但每天的人数是会变的, 这条线路的特点是每天只有一个站的乘客数量会改变, lin88 要每天监测一个区间, l 站到 r 站之间的总乘客数。你需要每天在乘客数量改变之后输出 l 站到 r 站之间的总乘客数。

Input

第一行两个整数 $n(1 \leq n \leq 10^5), m(1 \leq m \leq 10^5)$, 分别代表站数和天数。

第二行有 n 个整数, 第 i 个整数 a_i 表示第 i 个站初始的乘客数。

接下来有 m 行, 每行 4 个整数 $x, v, l, r(1 \leq x \leq n, 1 \leq v \leq 10^9, 1 \leq l \leq r \leq n)$, 表示第 i 天时, 第 x 个站的乘客数变成了 v , 并监测 l 到 r 间的乘客数量。

Output

输出 m 行, 每行一个整数, 依次代表当天的监测结果

Examples

standard input	standard output
5 2 1 2 3 4 5 1 4 3 5 2 1 1 4	12 12
5 5 5 4 3 2 1 1 2 3 4 5 4 1 3 3 4 1 5 5 1 4 5 4 3 1 5	5 9 16 3 14

Problem G. lin88 的宇宙航行

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

众所周知, lin88 喜欢宇宙航行, 来往于不同星系, 前往未知的星系。

宇宙航行需要搭建星际航线。

在不断的飞行穿梭中, lin88 想实时知道自己的状态, 现有 n 个已知星系, 编号从 $1 \sim n$, 但没搭建航线, lin88 会不断地穿梭在其中搭建航线, 时不时会看一下至少还需要多条航线才能使任意两个星系间可来往。航线是双向的。

Input

第一行两个整数 $n, m (1 \leq n, m \leq 10^5)$, 分别代表星系数, 和询问次数。

接下来 n 行, 每行一个整数 $q (q \in \{0, 1\})$ 开头。

若 $q = 1$, 后边有两个整数 $u, v (1 \leq u, v \leq n)$, 代表搭建 u, v 之间的航线。

若 $q = 0$, 为询问至少还需要多条航线才能任意两个星系间可来往。

Output

对于每个 $q = 0$ 的询问, 输出一行一个整数代表至少还需要多条航线能任意两个星系间可来往。

Examples

standard input	standard output
4 5 0 0 1 4 1 0 0	3 3 2 2
5 6 1 5 2 1 1 3 0 0 1 4 2 0	2 2 1

Problem H. lin88 找规律

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

众所周知, lin88 看到数学题就只会打表找规律, 找不到就束手无策了。

但有时, 打出了表也找不到规律, 所以他希望你可以帮帮他, 怎么把打出的表中的规律找出来。

具体来说是这样的, lin88 想知道打出来的表的循环节是多长。

打出的表是一个整数序列, 序列开头可能是乱序, 随后可能进入循环节。

设有两个整数序列 A, B , 长度分别为 $|A|, |B|$, 则打出来的表可表示为 $S = A + k \cdot B$ ($1 < k, 0 \leq |A|, 0 \leq |B|$), 其中 k 为整数, 即序列 A 后面接了 k 遍的序列 B 。

要求求出, 在 $k \cdot |B|$ 最大, 即总循环最长的情况下, 最小的 $|B|$, 即循环节长度。

Input

第一行一个整数 n ($1 \leq n \leq 10^6$) 代表序列的长度。

接下来的一行为 n 个整数 a_i ($1 \leq a_i < 10^6$) 代表整个序列 S 。

Output

输出一行一个整数, 表示循环节长度, 若不存在循环节, 输出 -1 。

Examples

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

Problem I. compute 的期末作业

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

众所周知, 计算机科学与技术有一门必修课是计算机组成原理, 而 compute 一直沉迷于训练, 没有怎么学习这门课。

现在到期末了, compute 十分苦恼, 因为今天老师布置了期末大作业, 而 compute 甚至听不懂要干什么。作业是这样的, 要求画出一个由若干个 RAM 芯片和若干个 ROM 芯片组成的电路, 而老师为了让每个组做的题目不太一样, 让每组同学画的两种芯片的个数不一样。

老师按组号从小到大分别要求画 $1, 2, 4, \dots, 2^{n-1}$ 个 RAM 芯片, 按组号从大到小分别要求画 $1, 2, 4, \dots, 2^{n-1}$ 个 ROM 芯片, 即第 i 组的同学要画 2^{i-1} 个 RAM 芯片和 2^{n-i} 个 ROM 芯片。

由于 compute 真的没有听过课, 所以她甚至不知道自己是哪一组的, 但她还是要完成作业交差的。

虽然要交差, 可是 compute 更想训练, 所以经过一番调查, 她发现老师并没有空去数她到底画了几个芯片, 只会确认每组画的总的芯片个数除以某个奇数 p 的余数是否达到要求。

现在机智的 compute 想要完成任务, 可她只知道班上有 n 组, 在她不知道她是哪一组的前提下, 她想知道她至少画多少芯片才能安全萌混过关。

简单来说, compute 想要知道 $\max_{1 \leq i \leq n} \text{cnt}_i \pmod p$, cnt_i 表示第 i 组需要画的芯片数量。

Input

第一行一个整数 $T (1 \leq T \leq 100)$, 表示测试组数。

接下来有 T 行, 每行有两个整数, $n (1 \leq n \leq 10^{18}), p (1 \leq p \leq 1 \times 10^4)$, 分别表示班上的小组数量和老师检查作业时所用的模数, 保证模数 p 为奇数。

Output

输出共 T 行。

每行输出一个数表示在当前情况下最少要画的芯片总数。

Examples

standard input	standard output
1 5 9	8
2 1 3 3 5	2 4

Note

在第一个样例中, 班级中共有 5 组同学。

如果 compute 是第 1 组的, 她被要求画 $2^0 + 2^4 = 17$ 个芯片, 而 compute 只要画 8 个就可以萌混过关。

如果 compute 是第 2 组的, 她被要求画 $2^1 + 2^3 = 10$ 个芯片, 而 compute 只要画 1 个就可以萌混过关。

如果 compute 是第 3 组的, 她被要求画 $2^2 + 2^2 = 8$ 个芯片, 而 compute 只要画 8 个就可以萌混过关。

如果 compute 是第 4 组的, 她被要求画 $2^3 + 2^1 = 10$ 个芯片, 而 compute 只要画 1 个就可以萌混过关。

如果 compute 是第 5 组的, 她被要求画 $2^4 + 2^0 = 17$ 个芯片, 而 compute 只要画 8 个就可以萌混过关。

为了确保绝对安全, compute 只要画 8 个芯片就无论如何都能过关了。