

Problem A. 烦人的交通 (Annoying Traffic)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 4 秒
空间限制: 256 MB

“红灯行，绿灯停。”你已经听过太多次了，但你仍然决定遵守规则。

考虑一个 $2N \times 2M$ 节点网格上的道路网络，节点以 (i, j) ($1 \leq i \leq 2N$, $1 \leq j \leq 2M$) 编号。每个节点 (i, j) 通过长度为 1 的道路与 $(i-1, j)$ 和 $(i, j-1)$ 相连（当这些节点存在时）。你的速度为每单位时间 1，因此经过每条道路需要 1 单位时间。

对于每个满足 $1 \leq r \leq N$, $1 \leq c \leq M$ 的 (r, c) ，四个节点 $(2r-1, 2c-1)$ 、 $(2r-1, 2c)$ 、 $(2r, 2c-1)$ 和 $(2r, 2c)$ 构成一个路口。初始时每个路口均显示红灯，阻止所有通行。其余所有道路均可随时通行。

由 $(2r-1, 2c-1)$ 、 $(2r-1, 2c)$ 、 $(2r, 2c-1)$ 和 $(2r, 2c)$ 构成的路口，只能通过到达其控制节点 $(2r, 2c)$ 来激活。到达后，你按下下一个按钮：以概率 $\frac{p}{q}$ 进入状态 A，以概率 $1 - \frac{p}{q}$ 进入状态 B。

- **状态 A**：按下按钮后的第一个单位时间内，行方向移动（改变 i ）显示**绿灯**，列方向移动（改变 j ）显示**红灯**。此后两个方向每单位时间交替一次。
- **状态 B**：按下按钮后的第一个单位时间内，列方向移动（改变 j ）显示**绿灯**，行方向移动（改变 i ）显示**红灯**。此后两个方向每单位时间交替一次。

只有当你所途经的路口中对应方向当前显示**绿灯**时，你才能通行该方向的道路。不同路口的信号灯相互独立。

你从 $(2N, 2M)$ 出发，希望到达 $(1, 1)$ ，并以**最优策略**行走以最小化期望总行走时间。

给你 T 组询问。对于每组询问，给定 N 和 M ，求在最优策略下从 $(2N, 2M)$ 到 $(1, 1)$ 的期望行走时间。

可以证明，答案可以表示为有理数 $\frac{p}{q}$ ($p, q \in \mathbb{N}^*$, $\gcd(p, q) = 1$)，且满足 $\gcd(q, 998\,244\,353) = 1$ 。你只需输出一个整数 x ($0 \leq x < 998\,244\,353$)，使得 $q \times x \equiv p \pmod{998\,244\,353}$ 。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 100$)。接下来是每组测试用例的描述。

每组测试用例仅一行，包含四个整数 N, M, p, q ($1 \leq N, M \leq 10^6$, $1 \leq p < q < 998\,244\,353$) ——题目的参数。

输出格式

对于每组测试用例，输出一个整数，表示在最优策略下行走时间的期望值对 998 244 353 取模的结果。

测试样例

标准输入	标准输出
2	6
2 2 1 2	271128110
3 5 1 3	

Problem B. 按位最大化 (Bitwise Maximization)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 1 秒
空间限制: 256 MB

小羊有一个非负整数列表和两个空的多重集合。他需要将列表中的每个整数放入两个多重集合之一。注意，多重集合可以包含重复的值。

为了给小羊的工作评分，他的领导分别计算两个多重集合的按位异或 (XOR) 值，并将结果相加得到最终得分。小羊希望最大化得分，你能告诉他最高能得多少分吗？

一个多重集合的按位异或值为这个集合的异或和。空的多重集合的按位异或值视为 0。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 10^4$)。接下来是每组测试用例的描述。

每组测试用例的第一行包含一个整数 n ($1 \leq n \leq 5 \times 10^5$) —— 列表的长度。

每组测试用例的第二行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 2^{30}$) —— 列表中的元素。

保证所有测试用例的 n 之和不超过 5×10^5 。

输出格式

对于每组测试用例，输出一个整数，表示最大得分。

测试样例

标准输入	标准输出
4	1
1	6
1	6
3	4
1 2 3	
4	
1 1 3 3	
4	
1 2 2 3	

Problem C. 比赛：连胜纪录 (Competition: Winning Streaks)

输入模式：标准输入
输出模式：标准输出
时间限制：1 秒
空间限制：256 MB

小羊正在观看一场特殊的羽毛球比赛，Alice 和 Bob 正在进行对决。若将两位选手的得分记为 x 和 y ，当以下两个条件同时满足时，比赛结束：

- $\max(x, y) \geq k$ ，其中 k 为给定参数。
- $|x - y| \geq 2$ 。

但不幸的是，小羊睡着了。过了相当长一段时间后，他醒来，发现比分已经大幅改变。

在他睡着之前，Alice 和 Bob 的比分分别为 x_1, y_1 ，而当他醒来时，比分变为 x_2, y_2 。在小羊睡眠期间，Alice 最长连胜长度的最小值和最大值分别是多少？

连胜是指一名选手连续赢得若干分的情况，连胜的长度为该期间赢得的分数数量。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 10^4$)。接下来是每组测试用例的描述。

每组测试用例仅一行，包含五个整数 k, x_1, y_1, x_2, y_2 ($1 \leq k \leq 10^9, 1 \leq x_1 < x_2 \leq 10^9, 1 \leq y_1 \leq y_2 \leq 10^9$) —— 比赛参数以及小羊睡前和醒来后 Alice 和 Bob 的比分。

保证 $\max(x_1, y_1) < k$ 或 $|x_1 - y_1| \leq 1$ 成立，且 $\max(x_2, y_2) \leq k$ 或 $|x_2 - y_2| \leq 2$ 成立。即，当小羊睡着时比赛尚未结束，但当他醒来时比赛可能已经结束。

输出格式

对于每组测试用例，输出两个整数，分别表示 Alice 最长连胜长度的最小值和最大值。

测试样例

标准输入	标准输出
4	2 9
21 1 1 10 5	2 2
21 19 20 22 21	5 5
21 16 19 21 19	2 13
21 9 12 22 20	

Problem D. 运送报纸 (Delivering Newspapers)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 8 秒
空间限制: 256 MB

小羊是牛客城的一名居民, 他每天都喜欢读报纸。

NowCity 由 n 个区 (分别称为第 1 区、第 2 区...第 n 区) 组成, 城市中共有 $n-1$ 条双向街道。每条街道连接两个区, 且不经其他街道。保证从任意一个区出发, 都能通过这些街道到达其他任意一个区。

两家最大的报社分别位于第 1 区和第 2 区, 小羊同时订阅了这两家报社的报纸。每天, 两家报社各派一名送报员将报纸送到小羊家。小羊不住在第 1 区或第 2 区。

第 i 条街道连接第 u_i 区和第 v_i 区, 送报员从一端走到另一端需要 1 单位时间。 $dis(u, v)$ 表示从 u 到 v 的最短时间。

小羊希望尽快读到报纸, 因此他将不便值定义为 P 的获取两份报纸的总时间次幂 (即第一家报社的配送时间加上第二家报社的配送时间)。即若小羊住在第 x 区, 不便值为 $P^{dis(1,x)+dis(2,x)}$, 其中 P 为给定常数。

但小羊对城市街道的构型完全不熟悉。对于每种构型, 他希望了解关于 $B = [P^{dis(1,3)+dis(2,3)}, \dots, P^{dis(1,n)+dis(2,n)}]$ 的一些信息。

小羊想要确认的目标值为 $\sum_{x=0}^{2n} [(freq(P^x, B) \oplus P) \bmod P] \times P^x$, 其中 $freq(P^x, B)$ 表示 P^x 在 B 中出现的次数, $x \oplus y$ 为 x 与 y 的按位异或。

因此他想知道, 对于每个 $i = 1, 2, \dots, n-1$, 在所有满足 $dis(1, 2) = i$ 的街道构型中, 目标值的平均值是多少? 给定整数 N , 对于每个 $n = 3, \dots, N$, 求解上述问题并输出 $n-1$ 个整数作为结果。

当且仅当存在某对 (u, v) ($1 \leq u, v \leq n, u \neq v$), 使得两种构型中恰好一种在 u 与 v 之间有街道, 则认为这两种构型不同。

将结果对给定素数 mod ($9.9 \times 10^8 < mod < 1.01 \times 10^9$) 取模输出。可以证明, 答案可以表示为有理数 $\frac{p}{q}$ ($p, q \in \mathbb{N}^*, \gcd(p, q) = 1$), 且满足 $\gcd(q, mod) = 1$ 。你只需输出一个整数 x ($0 \leq x < mod$), 使得 $q \times x \equiv p \pmod{mod}$ 。

输入格式

输入的唯一一行包含三个整数 N, P, mod ($3 \leq N, P \leq 500, 9.9 \times 10^8 < mod < 1.01 \times 10^9$) ——城市的最大区数、给定的参数与模数。

保证 mod 为素数。

输出格式

输出 $N-2$ 行。第 n ($1 \leq n \leq N-2$) 行输出 $n+1$ 个整数, 分别表示在共有 $n+2$ 个区且 $dis(1, 2) = 1, 2, \dots, n+1$ 的条件下的答案。

测试样例

标准输入	标准输出
5 3 1000000007	54 18 500000287 180 27 680001466 200000716 513 0

Problem E. 简单谜题 (Easy Puzzle)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 2 秒
空间限制: 256 MB

小羊发明了这个谜题:

想象将 $(n+1) \times (m+1)$ 个点按 $n+1$ 行、 $m+1$ 列排成一张网格，编号以下规则给定：位于第 r 行、第 c 列 ($0 \leq r \leq n$, $0 \leq c \leq m$, 均从 0 开始计数) 的点，编号为 $v = r \times (m+1) + c + 1$ 。每个点都与它在这张网格上左右、上下相邻的点相连。

你需要从这张网格的边中标记出一个子集 S ，要求每个点连出的被标记的边的数量只能是 0 或 2。

对于每个网格中的 1×1 的方格，我们关心的是它的上、下、左、右四条边中有多少条被标记。这道题要求：除了至多 5 个小方格之外，其余所有小方格都恰好有 2 条边界边被标记。

形式化地：

- 给定一张无向图，共有 $(n+1) \times (m+1)$ 个顶点。顶点编号为 $1, 2, \dots, (n+1) \times (m+1)$ 。
 - 你可以在图上标记一个边的子集。顶点 u 与 v 之间存在一条边，当且仅当以下条件至少有一个成立：
 - $|u - v| = m + 1$ 。
 - $\left\lfloor \frac{u-1}{m+1} \right\rfloor = \left\lfloor \frac{v-1}{m+1} \right\rfloor \wedge |u - v| = 1$ 。
 - 标记完边之后，设 S 为被标记的边的集合。以下条件成立：
 - 每个顶点所关联的被标记边数只能是 0 或 2。
 - 考虑所有满足 $i \in \{1, 2, \dots, n\}$, $j \in \{1, 2, \dots, m\}$ 的数对 (i, j) ，定义如下的边：
 - * $e_1(i, j)$ 是连接 $i \times (m+1) + j$ 与 $i \times (m+1) + j + 1$ 的边。
 - * $e_2(i, j)$ 是连接 $(i-1) \times (m+1) + j$ 与 $(i-1) \times (m+1) + j + 1$ 的边。
 - * $e_3(i, j)$ 是连接 $i \times (m+1) + j$ 与 $(i-1) \times (m+1) + j$ 的边。
 - * $e_4(i, j)$ 是连接 $i \times (m+1) + j + 1$ 与 $(i-1) \times (m+1) + j + 1$ 的边。
- 满足 $1 \leq i \leq n$, $1 \leq j \leq m$ 的数对 (i, j) 中，至多有 5 个不同的 (i, j) 使得

$$|\{e_1(i, j), e_2(i, j), e_3(i, j), e_4(i, j)\} \cap S| \neq 2$$

你能解决这道题吗？

输入格式

每个测试包包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 100$)。接下来是每组测试用例的描述。

每组测试用例仅一行，包含两个整数 n, m ($1 \leq n, m \leq 5 \times 10^5$, $1 \leq n \times m \leq 5 \times 10^5$) —— 给定图的两个参数。

保证所有测试用例的 $n \times m$ 之和不超过 5×10^5 。

输出格式

对于每组测试用例，若不存在解则输出 -1 。否则，按如下格式输出你标记的边：

- 第一行输出一个整数，表示你标记的边数，记为 M ($0 \leq M \leq n \times (m+1) + m \times (n+1)$)。
- 接下来 M 行，第 i 行输出两个整数，表示第 i 条标记边的两个端点。

如果有多种可行的构造方案，输出任何一种即可。

测试样例

标准输入	标准输出
2	0
1 1	8
2 3	1 2
	2 3
	3 4
	5 6
	6 7
	7 8
	1 5
	4 8

Problem F. 神奇的树 (Fabulous Tree)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 6 秒
空间限制: 512 MB

大小为 n 的树是一个有 $n - 1$ 条双向边的连通图。带标签的树是指每个顶点和每条边都被赋予一个非负整数值值的树。

顶点 u 被赋予的值为 a_u ， u 与 v 之间的边被赋予的值为 $w_{u,v}$ 。若对于每对存在边的 (u, v) ，均满足 $w_{u,v} = |a_u - a_v|$ ，则称该带标签的树为**神奇的树**。

对于一棵树，其**复杂度**定义为所有顶点被赋予的值中最大值与最小值之差。

给定一棵以 1 为根的带标签树，但各顶点的值缺失。对于从 1 到 n 的每个 i ，若将顶点 i 对应的子树的顶点赋值，使其成为一棵神奇的树，该子树的最小复杂度是多少？注意各子树的问题需独立求解。即某子树复杂度的最小化并不意味着其他子树的复杂度也一定达到最小。

顶点 u 的子树包含且仅包含所有满足“从 1 到 v 的最短路径经过顶点 u ”的顶点 v ，并保留原树的连接关系。

输入格式

每个测试包包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 10^4$)。接下来是每组测试用例的描述。

每组测试用例的第一行包含一个整数 n ($3 \leq n \leq 10^5$) ——树的顶点数。

接下来 $n - 1$ 行，第 i 行包含三个整数 u_i, v_i 和 w_{u_i, v_i} ($1 \leq u_i, v_i \leq n, u_i \neq v_i, 0 \leq w_{u_i, v_i} \leq 5000$) ——树中第 i 条边的两个端点及其被赋予的值。保证给定的边构成一棵树。

记 W 为同一测试中所有测试用例的 w_{u_i, v_i} 最大值。保证 n 的总和不超过 $\min\left(\frac{3 \times 10^7}{\max(1, W)}, 10^5\right)$ 。

输出格式

对于每组测试用例，输出 n 个整数，分别表示顶点 $1, 2, \dots, n$ 各自子树的最小复杂度。

测试样例

标准输入	标准输出
3	1 1 0
3	3 2 2 0
1 2 1	4 2 0 0 0
2 3 1	
4	
1 2 2	
2 3 1	
3 4 2	
5	
1 2 4	
1 3 1	
2 4 2	
2 5 1	

Problem G. 最大公约数图 (GCD Graph)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 2 秒
空间限制: 256 MB

小羊得到了一个有向图，其顶点由正整数 $1, 2, \dots$ 命名，共有无穷多个顶点。

对于每对满足 $1 \leq i < j$ 的整数 (i, j) ，存在一条从 i 到 j 、权值为 $w_{i,j} = \gcd(i, j)$ 的有向边，即若你决定从 i 走到 j ，需要支付代价 $w_{i,j}$ 。图中不存在其他边。

对于每对满足 $1 \leq u < v$ 的整数 (u, v) ，定义 $cost(u, v)$ 为从 u 到 v 的最小总代价。

小羊想知道对于某些 (l, r, n) ， $\sum_{i=l}^r cost(i, n)$ 的值是多少，你能告诉他答案吗？

注意， $\gcd(x, y)$ 表示 x 与 y 的最大公因数。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 100$)。接下来是每组测试用例的描述。

每组测试用例仅一行，包含三个整数 l, r, n ($1 \leq l \leq r < n \leq 10^7$) —— 小羊询问的参数。

注意对 $\sum n$ 没有限制。

输出格式

对于每组测试用例，输出一个整数，表示 $\sum_{i=l}^r cost(i, n)$ 的值。

测试样例

标准输入	标准输出
3	8
1 5 6	54
2 33 36	158
1 99 100	

Problem H. 超空间配对 (Hyperspace Pairing)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 2 秒
空间限制: 512 MB

n 维超空间立方体是一个有 2^n 个顶点的图，每个顶点对应一个不同的 n 位二进制串。当且仅当两个顶点的二进制串恰好在**两个位置**不同时，它们相邻。每个顶点以其二进制串对应的整数值为标签，因此顶点标签为 $0, 1, \dots, 2^n - 1$ 。

太空管理局正在变换超立方体网络上进行同步量子传输。每个节点都必须被匹配成一对，当且仅当两个节点在变换超立方体中相邻时，它们才可以被配对。

不幸的是，一次空间异常永久损坏了恰好两个节点 a 和 b 。请判断剩余的 $2^n - 2$ 个节点能否被划分为若干对，使得每个节点恰好出现在一对中，且每对中的两个节点相邻。若可以，请输出一种合法方案。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 10^4$)。接下来是每组测试用例的描述。

每组测试用例的第一行包含三个整数 n, a, b ($2 \leq n \leq 22, 0 \leq a, b \leq 2^n - 1, a \neq b$) —— 变换超立方体的维数以及两个损坏节点的标签。

保证所有测试用例的 2^n 之和不超过 2^{22} 。

输出格式

对于每组测试用例，若存在合法方案，在第一行输出 **'Yes'**，随后输出 $2^{n-1} - 1$ 行，每行包含两个整数 $u\ v$ ，表示一对配对。否则输出 **'No'**。'Yes' 或 'No' 的大小写可任意选择。

测试样例

标准输入	标准输出
3	Yes
2 0 3	1 2
2 0 1	No
3 2 7	Yes
	0 5
	1 4
	3 6

Problem I. 不完美点积与叉积之和 (Imperfect Dot Sums and Cross Sums)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 4 秒
空间限制: 256 MB

给你三维空间中的 n 个向量 $V_1, V_2, \dots, V_n$, 其中 $V_i = (x_i, y_i, z_i)$ 。

你需要回答 q 个询问。每个询问给定五个整数 a, b, c, l, r 。设 $T = (a, b, c)$ (使用输入描述中所述的随机方法生成)。该询问的答案为

$$\sum_{i=l}^r (|V_i \cdot T| + \|V_i \times T\|)$$

其中 $V_i \cdot T = x_i a + y_i b + z_i c$ 为点积, $V_i \times T = (y_i c - z_i b, z_i a - x_i c, x_i b - y_i a)$ 为叉积, 其模长为 $\|V_i \times T\| = \sqrt{(y_i c - z_i b)^2 + (z_i a - x_i c)^2 + (x_i b - y_i a)^2}$ 。

由于精确答案可能难以高效计算, 对于每个询问, 你输出的答案 A' 只要满足

$$\frac{|A - A'|}{\max(A, 1)} \leq 0.1$$

即可被接受, 其中 A 为真实答案。

本题恰好有 30 个测试数据。

输入格式

第一行包含一个整数 n ($1 \leq n \leq 10^5$)。

接下来 n 行, 每行包含三个整数 x_i, y_i, z_i ($-10^6 \leq x_i, y_i, z_i \leq 10^6$), 且 x_i, y_i, z_i 中至少有一个不为零——向量的参数。

下一行包含一个整数 q ($1 \leq q \leq 10^5$)。

接下来 q 行, 每行包含五个整数 a, b, c, l, r ($-10^6 \leq a, b, c \leq 10^6$; $1 \leq l \leq r \leq n$) ——询问的参数。对于每个询问, a, b, c 的值独立且均匀地从 $[-10^6, 10^6]$ 中的整数随机抽取 (给定样例除外)。

输出格式

对于每组询问, 输出一个数作为答案。

测试样例

标准输入	标准输出
2	12.000000
3 4 0	50.000000
0 0 5	
2	
1 0 0 1 2	
3 4 0 1 2	

Problem J. 那就四舍五入吧! (Just Round It)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 8 秒
空间限制: 256 MB

小羊得到了一个正整数 V 作为礼物。小羊很讨厌数码 9，幸运的是， V 也没有任何数码 9。

但这个数太大了，小羊记不住。所以他决定使用一个“四舍五入机”来对这个数进行四舍五入。“四舍五入机”是这么运作的。

- 如果 $V = 0$ ，“四舍五入机”就什么都不做。否则就按照如下流程操作。
- 找到最小的使得 $10^v > V$ 的 v 。
- 从 $1, 2, \dots, v$ 中独立等可能地选择一个正整数 i 。
- 找到满足 $x \times 10^i - 5 \times 10^{i-1} \leq V < x \times 10^i + 5 \times 10^{i-1}$ 的 x 。可以证明 x 存在且唯一。
- 将 V 赋值为 $x \times 10^i$ 。

注意，尽管 V 开头是正数，但可能经过几次操作后就变成 0 了。

小羊恰好使用“四舍五入机” k 次。经过这些操作后， V 的期望是多少呢？

可以证明答案可以表示为有理数 $\frac{p}{q}$ ($p, q \in \mathbb{N}^*$, $\gcd(p, q) = 1$)，且满足 $\gcd(q, 998\,244\,353) = 1$ 。你只需输出一个整数 x ($0 \leq x < \text{mod}$) 使得 $q \times x \equiv p \pmod{\text{mod}}$ 。

输入格式

每个输入包含多个测试样例。输入的第一行包含测试样例的数量 T ($1 \leq T \leq 10^4$)。每个测试样例的输入格式如下。

第一行包含两个整数 n, k ($1 \leq n \leq 10^6, 1 \leq k \leq 10^9$) — V 的位数和操作次数。

第二行包含一个整数 V ($10^{n-1} \leq V < 10^n$) — V 一开始的数值。保证 V 中不存在数码 9。

保证所有测试样例的 n 的和不超过 10^6 。

输出格式

对于每个测试样例，输出一个整数，表示 V 在 k 次操作后的期望关于 998 244 353 取模的结果。

测试样例

标准输入	标准输出
3	436801049
1 100	75
5	360538984
2 1	
50	
6 3	
114514	

Problem K. 幼儿园 (Kindergarten)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 3 秒
空间限制: 256 MB

牛客城的天气渐渐转好，牛客幼儿园的园长准备带小朋友们去主题公园玩。

主题公园有 n 个游乐设施，有 m 条游乐设施之间的无向的通道。第 i 个设施被称为设施 i ，而第 i 条通道连结了设施 u_i 和设施 v_i 。你没办法不使用通道从一个设施到另一个设施，同时任意两个设施是相互可达的。

一般而言，每条通道需要 T 单位的时间通过。但现在在一些通道上正在进行 k 个活动，使得这条路的通行时间变得不一致。第 i 个活动发生在第 w_i 条通道上，而当小朋友到达公园时，需要 c_i 单位的时间去通过这条通道。

在活动的进行过程中，通过通道的时间也可能发生改变。总共有 q 次更新，第 i 次中，第 x_i 条通道的通行时间变为了 y_i 。这里保证， x_i 是曾经发生过活动的街道。

更新是累积且持久的，即前面更新所做的修改将在所有后续的更新和询问中持续生效。

在每次更新后，小朋友会提出 L_i 个问题：从 a 到 b 需要多少时间呢？你能帮忙回答这些问题吗？

输入格式

测试样例由多行组成。

第一行包含三个整数 n, m, T ($3 \leq n \leq 2 \times 10^3, n-1 \leq m \leq 10^5, 1 \leq T \leq 10^6$) — 设施个数、通道个数以及一般而言通道需要的通行时间。

接下来 m 行中的第 i 行包含两个整数 u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$) — 第 i 条通道的两个端点。可能有重边。

接下来的一行包含一个整数 k ($1 \leq k \leq 50$) — 在进行活动的通道数量。

接下来 k 行中的第 i 行包含两个整数 w_i, c_i ($1 \leq w_i \leq m, 1 \leq c_i \leq 10^9$) — 有活动的通道的下标及其通行时间。保证 $\forall 1 \leq i < j \leq k, w_i \neq w_j$ 。

接下来的一行包含一个整数 q ($1 \leq q \leq 2 \times 10^4$) — 更新的次数。

每次更新由多行组成。以下是更新的格式。

第一行包含两个整数 x_i, y_i ($x_i \in \{w_1, w_2, \dots, w_k\}, 1 \leq y_i \leq 10^9$) — 更新的通道的下标及其新的通行时间。

接下来的一行包含一个整数 L_i ($1 \leq L_i \leq 10^4$) — 小朋友提出的问题的个数。

接下来 L_i 行中的第 j 行包含两个整数 a_j, b_j ($1 \leq a_j, b_j \leq n$) — 小朋友提出的问题对应的两个端点。

保证 L_i 的和不超过 2×10^4 。

输出格式

对于每个小朋友提出的问题，输出一个正整数，表示从 a_j 走到 b_j 所需的最小时间。

测试样例

标准输入	标准输出
4 4 1	7
1 2	2
2 3	2
1 3	
1 4	
3	
1 1	
2 2	
3 100	
3	
2 5	
1	
3 4	
3 1	
1	
3 4	
1 5	
1	
3 4	

Problem L. 懒得打乱 (Lazy Shuffling)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 2 秒
空间限制: 256 MB

小羊被分配了一个打乱长度为 n 的排列的任务。一个排列就是一个长度为 n 的包含 $1, 2, \dots, n$ 中的数各一次的序列。

小羊的领导会通过比较打乱前后数组的逆序对数量来评价小羊的工作。

也就是说，我们定义一个排列 $A = [A_1, A_2, \dots, A_n]$ 的逆序对数量为 $f(A) = \sum_{1 \leq i < j \leq n} 1_{A_i > A_j}$ ，而打乱前后的排列分别是 A 和 A' 。那么，小羊的领导会计算 $g(A, A') = |f(A) - f(A')|$ 来检查小羊的工作。领导希望这个数值越大越好。

但小羊是个懒人。他不想真的好好打乱数组，所以他决定使用一个长度为 n 的排列 $p = [p_1, p_2, \dots, p_n]$ ，这样，当他工作中遇到一个排列 $A = [A_1, A_2, \dots, A_n]$ 的时候，他就可以提交 $f_p(A) = [A_{p_1}, A_{p_2}, \dots, A_{p_n}]$ 作为他的工作成果。可以证明 $f_p(A)$ 也是一个排列。

小羊认为一个排列是幸运的，当且仅当如果他使用上述的“打乱方法”的话，这个排列 A 能最大化 $g(A, f_p(A))$ 。也就是说，对于任意一个长度为 n 的排列 A_0 ，都有 $g(A, f_p(A)) \geq g(A_0, f_p(A_0))$ 。

那么有多少幸运的排列呢？鉴于答案可能很大，请将结果关于 998 244 353 取模后输出。

注意，当 $condition$ 成立时， $1_{condition}$ 等于 1，否则等于 0。

输入格式

输入包含两行。

第一行包含一个整数 n ($1 \leq n \leq 22$) — 排列的长度。

第二行包含 n 个整数 $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq n$) — 小羊用于打乱 A 的排列 p 。保证 $\forall 1 \leq i < j \leq n, p_i \neq p_j$ 。

输出格式

输出一个整数，表示幸运的排列的数量关于 998 244 353 取模的结果。

测试样例

标准输入	标准输出
3 3 1 2	4
3 3 2 1	2

Problem M. 可能连通罢 (Maybe Connected)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 1 秒
空间限制: 256 MB

小羊对于一个图中，两个点可能不相连但是连通这件事感到很讶异。两个点相连指的是两个点之间有边，而两个点连通指的是存在一条路径从一点出发到达另一点。

小羊给你一个 n 个结点的没有边的图，并让你给这个图上新增 m 条无向边。他希望你能最大化 (u, v) ($u \neq v$) 的对数，使得 u, v 不相连但是连通。图可能很大，所以你不需要完整的输出这个图的各项边，只需输出这个对数的最大值即可。而这里 $(u, v), (v, u)$ 被认为是相同的对。

注意图中不允许出现自环和重边。

输入格式

每个输入包含多个测试样例。输入的第一行包含测试样例的数量 T ($1 \leq T \leq 10^4$)。每个测试样例的输入格式如下。

每个测试样例有且仅有一行，包含两个整数 n, m ($1 \leq n \leq 10^5, 0 \leq m \leq \min\left(\frac{n(n-1)}{2}, 10^9\right)$) — 结点的个数以及需要新增的边的数量。

输出格式

对于每个测试样例，输出一个整数，表示连通但不相连的点对 (u, v) ($u \neq v$) 数量的最大值。

测试样例

标准输入	标准输出
3	1
3 2	0
4 6	3
5 3	

Problem N. 收拢至中位数 (Narrow to Median)

输入模式: 标准输入
输出模式: 标准输出
时间限制: 1 秒
空间限制: 256 MB

给定一个长度为 n 的数组 a ，每次操作你可以从数组中选取任意 k 个元素，将它们收拢至其中位数——即将这 k 个元素中的每一个都替换为这 k 个元素的中位数。

你必须恰好执行一次这样的收拢操作。求操作后数组所有元素之和的最大可能值。

中位数的定义：设选取的 k 个整数按非降序排列为 $b_1 \leq b_2 \leq \dots \leq b_k$ 。

- 若 k 为奇数，中位数为唯一的中间元素： $b_{(k+1)/2}$ 。
- 若 k 为偶数，中位数为两个中间元素的平均值： $\frac{b_{k/2} + b_{k/2+1}}{2}$ 。

输入格式

每个测试包含多组测试用例。第一行包含测试用例数 T ($1 \leq T \leq 10^4$)。接下来是每组测试用例的描述。

每组测试用例的第一行包含两个整数 n 和 k ($1 \leq k \leq n \leq 2 \times 10^5$) ——数组 a 的元素个数以及选取子集的大小。

每组测试用例的第二行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$) ——数组 a 的元素。

保证所有测试用例的 n 之和不超过 2×10^5 。

输出格式

对于每组测试用例，输出一个整数，表示操作后最大的数组和。

测试样例

标准输入	标准输出
2	19
6 3	20
1 1 4 5 1 4	
4 2	
1 3 6 10	

样例解释

对于第一组测试用例，我们可以将 $\{1, 4, 4\}$ 收拢至其中位数 4。数组变为 $\{4, 1, 4, 5, 1, 4\}$ ，元素之和为 19。可以证明没有任何收拢操作能得到更大的结果。

对于第二组测试用例，我们可以将 $\{1, 10\}$ 收拢至其中位数 5.5。数组变为 $\{5.5, 3, 6, 5.5\}$ ，元素之和为 20。可以证明没有任何收拢操作能得到更大的结果。