

Problem A. 排列问题

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

给定一个长度为 n 的排列* p 。你可以执行以下操作任意次（包括零次）：

- 选择一个整数 i 满足 $2 \leq i \leq n - 1$ 且 $p_i > \max(p_{i-1}, p_{i+1})$ 。然后交换 p_{i-1} 和 p_{i+1} 。

一个排列 q 从 p 是可达的，当且仅当存在一个操作序列，使得对 p 执行这些操作后可以得到 q 。
求从 p 可达的排列数量。由于该数字可能非常大，请输出其对 998 244 353 取模后的余数。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 5 \cdot 10^4$)。接下来是测试用例的描述。
每个测试用例的第一行包含一个整数 n ($3 \leq n \leq 10^6$)，表示 p 的长度。
第二行包含 n 个不同的整数 $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq n$)，表示 p 的元素。
保证所有测试用例中 n 的总和不超过 10^6 。

Output

对于每个测试用例，输出一个整数，表示从 p 可达的排列数量对 998 244 353 取模后的结果。

Example

standard input	standard output
5	1
3	1
1 2 3	2
3	12
2 1 3	4
3	
2 3 1	
8	
4 5 3 8 1 6 7 2	
5	
2 5 4 3 1	

*长度为 n 的排列是一个数组，它由 n 个从 1 到 n 的不同整数以任意顺序组成。例如， $[2, 3, 1, 5, 4]$ 是一个排列，但 $[1, 2, 2]$ 不是排列（2 在数组中出现了两次）， $[1, 3, 4]$ 也不是排列（ $n = 3$ 但数组中存在 4）。

Problem B. CF2071B

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 1024 megabytes

一个数组 a 被认为是丑陋的，当且仅当其元素之和是一个完全平方数。
对于任意数组 a ，定义 $f(a)$ 为 a 的丑陋非空子数组[†]的数量。
给定一个正整数 n 。令 $M(n)$ 为长度为 n 的所有排列[‡] q 中 $f(q)$ 的最小值。
请构造一个长度为 n 的排列 p ，使得 $f(p) = M(n)$ 。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 10^3$)。接下来是测试用例的描述。
每个测试用例的唯一一行包含一个整数 n ($1 \leq n \leq 5000$) —— 排列的长度。
保证所有测试用例中 n^2 的总和不超过 10^8 。

Output

对于每个测试用例，输出 n 个互不相同的整数 $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq n$)，表示你构造的排列 p 。
如果有多个答案，你可以输出其中任意一个。

Example

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

[†]数组 a 是数组 b 的子数组，当且仅当可以通过删除 b 开头若干个（可能是零个或全部）元素以及结尾若干个（可能是零个或全部）元素得到 a 。
[‡]长度为 n 的排列是一个数组，它由 n 个从 1 到 n 的不同整数以任意顺序组成。例如， $[2, 3, 1, 5, 4]$ 是一个排列，但 $[1, 2, 2]$ 不是排列（2 在数组中出现了两次）， $[1, 3, 4]$ 也不是排列（ $n = 3$ 但数组中存在 4）。

Problem C. 切树

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 1024 megabytes

给定一棵由 n 个顶点组成的无向树，顶点编号从 1 到 n 。树的根节点是顶点 1。我们定义树的切割序列及其对应的得分如下：

- 初始化一个空数组 a 和一个整数 $s = 0$ 。
- 执行以下操作，直到树为空：选择一个当前树中仍然存在的顶点。然后移除其子树[§]中的所有顶点。将所选顶点的编号追加到 a 的末尾，并将 s 增加本次操作后树中剩余顶点的数量。
- 最终的 a 被认为是树的一个切割序列， s 是其对应的得分。

如果两个切割序列的长度不同，或者至少在一个位置上不同，则认为它们是不同的。
求所有不同切割序列的得分之和。由于答案可能非常大，请输出其对 998 244 353 取模后的余数。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 10^4$)。接下来是测试用例的描述。
第一行包含一个整数 n ($2 \leq n \leq 3000$) — 树中顶点的数量。
接下来的 $n - 1$ 行，每行包含两个整数 u 和 v ($1 \leq u, v \leq n, u \neq v$)，表示顶点 u 和 v 之间的一条边。
保证输入构成一棵有效的树。
保证所有测试用例中 n^2 的总和不超过 $9 \cdot 10^6$ 。

Output

对于每个测试用例，输出一个整数，表示所有不同切割序列的得分之和对 998 244 353 取模后的结果。

Example

standard input	standard output
4	6
3	10
1 2	37
2 3	48
3	
1 2	
1 3	
4	
1 2	
2 3	
2 4	
4	
1 2	
1 3	
3 4	

[§]顶点 v 的子树是 v 及其所有后代以及它们之间所有边组成的子图。

Problem D. 整除

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

给定一个由 n 个顶点和 m 条边组成的无向图 G 。同时给定一个正整数 k 。注意, G 可能包含自环和重边。

对于顶点 u , 定义 $f(u)$ 为满足以下条件的最小非负整数 d : d 可以被 k 整除, 且存在一条从顶点 1 到顶点 u 的长度为 d 的路径。如果不存在这样的 d , 则 $f(u)$ 为 -1 。注意, 路径可以多次访问同一个顶点或同一条边。

求所有 $1 \leq u \leq n$ 的 $f(u)$ 的值。

Input

每个测试包含多个测试用例。 第一行包含测试用例的数量 t ($1 \leq t \leq 5 \cdot 10^4$)。 接下来是测试用例的描述。

每个测试用例的第一行包含三个整数 n 、 m 和 k ($2 \leq n \leq 5 \cdot 10^5, 0 \leq m \leq 5 \cdot 10^5, 1 \leq k \leq 10^9$), 分别表示 G 中的顶点数、 G 中的边数以及给定的常数。

接下来的 m 行, 每行包含两个整数 u 和 v ($1 \leq u, v \leq n$), 表示连接顶点 u 和 v 的一条无向边。

保证所有测试用例中 n 的总和与 m 的总和均不超过 $5 \cdot 10^5$ 。

Output

输出 n 个整数, 其中第 i 个整数表示 $f(i)$ 的值。

Example

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

Problem E. 猫猫

Input file: **standard input**
Output file: **standard output**
Time limit: 6 seconds
Memory limit: 1024 megabytes

给定一个长度为 n 的数组 a ，由非负整数组成。同时给定两个整数 k 和 m 。你必须恰好执行以下操作 k 次：

- 从 a 中选择一个元素，记为 x ，并将其删除。剩余部分拼接在一起。然后，将 $\min(x + 1, m)$ 插入到 a 的任意位置。

在不同的操作中，你可以多次选择同一个元素。求恰好执行 k 次操作后所能得到的字典序最小的数组。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 5 \cdot 10^4$)。接下来是测试用例的描述。

每个测试用例的第一行包含三个整数 n 、 k 和 m ($1 \leq n \leq 5 \cdot 10^4$, $0 \leq k \leq n$, $0 \leq m \leq 2 \cdot n$)，分别表示 a 的长度、需要执行的操作次数以及操作的参数。

第二行包含 n 个整数 $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq m$)，表示 a 中的元素。

保证所有测试用例中 n 的总和不超过 $5 \cdot 10^4$ 。

Output

对于每个测试用例，输出 n 个整数，表示恰好执行 k 次操作后所能得到的字典序最小的数组。

Example

standard input	standard output
4	1 1 2 3 3
5 2 3	1 1 2
1 2 3 1 2	0 0 0 0 0 2
3 1 6	0 1 1 3 4 5 6 7 7 8 9 10 14
1 1 1	
6 2 2	
1 0 0 0 0 0	
13 12 14	
6 5 4 3 2 0 1 1 7 8 9 10 11	

Problem F. 字母表

Input file: **standard input**
Output file: **standard output**
Time limit: 10 seconds
Memory limit: 1024 megabytes

在本题中, 字母表被定义为小写英文字母从 **a** 到 **z** 的一个排列。在字母表 A 下, 字母 x 的字典序小于字母 y 当且仅当在 A 中 x 出现在 y 之前。

在字母表 A 下, 字符串 s 的字典序小于字符串 t 当且仅当满足以下两个条件之一:

- s 是 t 的真前缀。
- 存在一个整数 i ($1 \leq i \leq \min(|s|, |t|)$), 使得对于所有的 $1 \leq j < i$ 都有 $s_j = t_j$, 并且在字母表 A 下 s_i 的字典序小于 t_i 。

在字母表 A 下, 字符串 s 被称为一个 Lyndon 串, 当且仅当在 A 下它的字典序严格小于其所有的非空真后缀。

给定一个字符串 s 。请计算有多少种不同的字母表 A 使得 s 在 A 下是一个 Lyndon 串。由于答案可能很大, 请输出其对 2^{32} 取模后的余数。

Input

每个测试点仅包含一行, 给出一个字符串 s ($2 \leq |s| \leq 2 \cdot 10^7$)。

Output

输出一个整数, 表示使得 s 在字母表 A 下是 Lyndon 串的不同字母表 A 的数量对 2^{32} 取模的结果。

Examples

standard input	standard output
abcdefghijklmnopqrstuvwxyz	2076180480
abaabb	0
sserxhsfan	2452619264
arcabcagc	3598712832

Problem G. 图上游戏

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

给定一个简单连通无向图 G , 包含 n 个顶点和 m 条边, 以及一个由 k 个特殊顶点 $a_1, a_2, \dots, a_k$ 组成的集合。 G 有一个有趣的性质: 所有顶点的度数不超过三。

考虑在 G 上进行的以下游戏:

- 游戏由两名玩家 Alice 和 Bob 参与。初始时, Alice 站在一个非特殊顶点 s 上。然后, 两名玩家轮流操作, 由 Alice 先手:
 - 如果轮到 Alice 操作, 假设她当前在顶点 u 。如果 u 是特殊顶点, 游戏结束且 Alice 获胜。否则, 她选择一个与 u 有边相连的顶点 v 并移动到 v 。如果不存在这样的顶点, 游戏结束且 Bob 获胜。
 - 如果轮到 Bob 操作, 他会选择 G 中剩余的任意一条边并将其删除。如果 G 中没有边了, 他什么也不做。
- 可以证明游戏会在有限回合内结束。在整个游戏过程中, Alice 的位置对两名玩家都是已知的。

一个顶点 u 被认为是必胜的, 当且仅当:

- u 不是特殊顶点;
- 如果初始时 Alice 站在顶点 u 上, 在双方都采取最优策略的情况下, 她能赢得游戏。

找出所有必胜顶点。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 2 \cdot 10^4$)。接下来是测试用例的描述。

每个测试用例的第一行包含三个整数 n 、 m 和 k ($1 \leq k < n \leq 2 \cdot 10^5$, $n - 1 \leq m \leq \min(\frac{3n}{2}, 2 \cdot 10^5)$), 分别表示 G 中的顶点数、 G 中的边数和特殊顶点的数量。

接下来的 m 行, 每行包含两个整数 u 和 v ($1 \leq u, v \leq n$, $u \neq v$), 表示连接顶点 u 和 v 的一条边。保证 G 不包含自环或重边。此外, 保证所有顶点的度数不超过三。

下一行包含 k 个互不相同的整数 $a_1, a_2, \dots, a_k$ ($1 \leq a_i \leq n$), 表示特殊顶点。

保证所有测试用例中 n 的总和与 m 的总和均不超过 $2 \cdot 10^5$ 。

Output

对于每个测试用例, 首先输出一个整数 c ($1 \leq c \leq n - k$), 表示必胜顶点的数量。然后, 输出 c 个互不相同的整数 $w_1, w_2, \dots, w_c$ ($1 \leq w_i \leq n$), 表示必胜顶点。你可以以任意顺序输出这些顶点。

Example

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

Problem H. 困难问题

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

给定一个整数 n 。请找出长度为 n 的字典序最小的[¶]排列[‡] p ，使得对于任意 $1 \leq i \leq n$ ， $|p_i - p_{(i \bmod n)+1}|$ 都不是质数。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 10^4$)。接下来是测试用例的描述。
每个测试用例的唯一一行包含一个整数 n ($2 \leq n \leq 2 \cdot 10^5$)。
保证所有测试用例中 n 的总和不超过 $2 \cdot 10^5$ 。

Output

对于每个测试用例，如果不存在符合条件的长度为 n 的排列，请输出 -1 。
否则，输出 n 个互不相同的整数 $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq n$)，表示你找到的排列 p 。

Example

standard input	standard output
2	1 2
2	-1
3	

[¶]数组 a 字典序小于数组 b ，当且仅当满足以下条件之一：

- a 是 b 的前缀，且 $a \neq b$ ；或者
- 在 a 和 b 首次出现不同的位置上，数组 a 中的元素小于 b 中对应的元素。

[‡]长度为 n 的排列是一个数组，它由 n 个从 1 到 n 的不同整数以任意顺序组成。例如， $[2, 3, 1, 5, 4]$ 是一个排列，但 $[1, 2, 2]$ 不是排列（2 在数组中出现了两次）， $[1, 3, 4]$ 也不是排列（ $n = 3$ 但数组中存在 4）。

Problem I. 整数函数

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

对于非负整数 x , 令 $f(x)$ 表示 x 的二进制表示中 1 的个数。
给定两个整数 n 和 d 。计算:

$$\left(\sum_{i=0}^n f(i) \cdot f(i+d)\right) \bmod 998\,244\,353$$

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 10^4$)。接下来是测试用例的描述。
每个测试用例的唯一一行包含两个整数 n 和 d ($0 \leq n, d < 2^{60}$)。

Output

对于每个测试用例，输出一个整数，表示该表达式的值。

Example

standard input	standard output
5	0
0 0	1
1 1	14
5 7	6022
314 159	11512513
114514 1919810	

Problem J. 加除

Input file: **standard input**
Output file: **standard output**
Time limit: 5 seconds
Memory limit: 1024 megabytes

给定两个长度为 n 且由正整数组成的数组 a 和 b 。你需要回答 q 个以下三种类型的询问：

- $1\ i\ x\ (1 \leq i \leq n, 1 \leq x \leq 10^9)$: 赋值 $a_i \leftarrow x$;
- $2\ i\ x\ (1 \leq i \leq n, 1 \leq x \leq 10^9)$: 赋值 $b_i \leftarrow x$;
- $3\ l\ r\ (1 \leq l < r \leq n)$: 计算:

$$\max_{l \leq i < j \leq r} \frac{a_i + a_j}{b_i + b_j}$$

所有的询问将按给出的顺序依次处理。每次赋值操作都会永久修改对应的数组元素，并影响所有后续的询问。

Input

每个测试点的第一行包含两个整数 n 和 $q\ (2 \leq n \leq 10^5, 1 \leq q \leq 10^5)$ ，分别表示数组 a 和 b 的长度以及询问的数量。

第二行包含 n 个整数 $a_1, a_2, \dots, a_n\ (1 \leq a_i \leq 10^9)$ ，表示数组 a 的元素。

第三行包含 n 个整数 $b_1, b_2, \dots, b_n\ (1 \leq b_i \leq 10^9)$ ，表示数组 b 的元素。

接下来的 q 行，每行包含一个询问。

Output

对于每个第三种类型的询问，可以证明其答案能被唯一表示为最简分数 $\frac{x}{y}$ 。你需要输出两个整数 x 和 y 。

Example

standard input	standard output
6 7	40 3
10 20 30 100 1000000000 1000000000	65 6
1 5 2 10 1 1000000000	10 1
3 1 4	26 3
3 2 4	2000000000 1000000001
1 3 4	
3 1 4	
2 1 100	
3 1 4	
3 5 6	

Problem K. 构造

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

在本题中, 一个图灵机 M 被定义为一个元组 $(Q, q_0, q_h, \delta, p_0)$, 其中:

- Q : 所有可能状态的有限集合。为简便起见, Q 始终为 $\{0, 1, 2, \dots, N-1\}$, 其中 N 为某个正整数。
- $q_0 \in Q$: 初始状态。
- $q_h \in Q$: 停机状态。
- $\delta: (Q \setminus \{q_h\}) \times \{0, 1\} \rightarrow Q \times \{0, 1\} \times \{-1, 0, 1\}$: 转移函数。
- $p_0 \in \mathbb{Z}_{\geq 0}$: 指针的初始位置。

图灵机在一个由非负整数索引的半无限纸带 c 上运行。每个单元格包含 0 或 1。我们用 c_i 表示写在单元格 $i \in \mathbb{Z}_{\geq 0}$ 上的值。

执行过程遵循以下规则:

- M 的运行配置是一个二元组 (q, p) , 其中 q 是当前状态, p 是当前指针位置。初始时, $(q, p) = (q_0, p_0)$ 。
- 在每一步中, 严格按顺序发生以下事件:
 - 如果 $p < 0$, 机器崩溃。
 - 如果 $q = q_h$, 机器成功停机。
 - 否则, 令 $(q_{\text{next}}, w, d) = \delta(q, c_p)$ 。机器将更新当前单元格 $c_p \leftarrow w$, 状态转移至 $q \leftarrow q_{\text{next}}$, 并将指针移动 $p \leftarrow p + d$ 。

给定一个长度为 n 的数组 a 和一个整数 k 。保证:

- $1 \leq n < 2^{16}$ 。
- $0 \leq k < 2^{16}$ 。
- 对于所有 $0 \leq i < n$, $0 \leq a_i < 2^{16}$ 。

所有输入都预先写在纸带 c 上, 以 16 位小端序二进制编码。具体来说:

- n 存储在单元格 16 到 31。
- k 存储在单元格 32 到 47。
- 对于每个 $0 \leq i < n$, 值 a_i 存储在单元格 $48 + 16 \cdot i$ 到 $63 + 16 \cdot i$ 。
- 纸带上的所有其他单元格初始均为 0。

当 x 存储在单元格 l 到 r 时, 这意味着 x 的第 j 个最低有效位存储在 c_{l+j} 中 ($0 \leq j \leq r-l$)。例如, 如果 10 存储在单元格 1 到 7 中, 那么 $c_{1..7}$ 将是 0101000。

你的任务是构造一个图灵机 M , 用于计算 a 中严格大于 k 的元素的数量。

你的机器最终必须到达状态 q_h 并停机。停机时, 答案必须存储在单元格 0 到 15 中。一旦机器停机, 指针 p 的最终位置以及写在其他单元格中的值都不重要。

你的机器包含的状态数不应超过 11451 个, 并且在不超过 $4 \cdot 10^8$ 步内停机。

Input

本题没有输入。

Output

首先, 输出一个整数 N ($1 \leq N \leq 11451$), 表示 $|Q|$ 。

然后, 输出 N 行, 每行包含 6 个整数。第 i 行的前三个整数表示 $\delta(i, 0)$, 后三个整数表示 $\delta(i, 1)$ 。

最后, 输出 3 个整数 q_0 、 q_h 和 p_0 ($0 \leq q_0, q_h < N, 0 \leq p_0$)。

Example

standard input	standard output
No Input	4 1 1 1 3 1 1 1 1 -1 2 0 1 2 1 -1 0 1 -1 3 0 1 3 1 1 0 3 100

Note

样例输入和输出仅用于演示目的, 并且是不正确的。实际的测试用例将没有任何输入。

有趣的事实: 样例输出是用于计算 BB(3) 的图灵机。

Problem L. 最长回文子串询问

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 512 megabytes

给定一个长度为 n 的字符串 s 。请回答 q 个如下类型的询问：

- 给定两个整数 l 和 r ($1 \leq l \leq r \leq n$)。求字符串 $s[1..l-1] + s[r+1..n]$ 的最长回文子串的长度，其中 $+$ 表示字符串拼接。这里， $s[x..y]$ 表示 s 从第 x 个字符到第 y 个字符的子串。特别地，如果 $x > y$ ， $s[x..y]$ 为空串。

Input

每个测试包含多个测试用例。第一行包含测试用例的数量 t ($1 \leq t \leq 10^4$)。接下来是测试用例的描述。每个测试用例的第一行包含两个整数 n 和 q ($1 \leq n \leq 5 \cdot 10^5$, $1 \leq q \leq 5 \cdot 10^5$)，分别表示 s 的长度和询问的数量。
第二行包含字符串 s 。保证 s 仅由小写英文字母组成。
接下来的 q 行，每行包含两个整数 l 和 r ($1 \leq l \leq r \leq n$)，表示一个询问。
保证所有测试用例中 n 的总和以及 q 的总和均不超过 $5 \cdot 10^5$ 。

Output

对于每个测试用例，输出 q 个整数，其中第 i 个整数表示第 i 个询问的答案。

Example

standard input	standard output
1	5
6 21	3
abcacb	1
1 1	1
1 2	1
1 3	0
1 4	3
1 5	2
1 6	1
2 2	1
2 3	1
2 4	3
2 5	3
2 6	2
3 3	1
3 4	4
3 5	3
3 6	1
4 4	1
4 5	1
4 6	3
5 5	
5 6	
6 6	