

Problem 1001. cats 的快乐 CF 刷题

Input file:standard input

Output file:standard output

Time limit:10 seconds

Memory limit:1024 megabytes

cats 非常喜欢在 CF(CatForces) 上做好玩的题。今天 CF 更新了 n 个新的题，cats 希望通过一定的顺序来做这些题以获得尽可能更多的快乐值。

这 n 个题被从左到右排列在一个双端队列中，其中从左到右的第 i 个题具有难度 a_i 和趣味度 b_i 。cats 有一个能力值 d ，初始为 0。cats 可以不断的选择去做双端队列中最左侧或者最右侧的题，然后将其从双端队列中移除，直到做完全部的 n 个题为止。

在做题的过程中，cats 可以收获快乐。如果 cats 当前的能力值为 d ，当 cats 做一个难度为 A ，趣味度为 B 的题时，以下两件事情会按下面的顺序依次发生：

1. cats 的能力值增加 A 。即 $d := d + A$ 。
2. cats 获得 $d \cdot B$ 点快乐值。

现在，cats 想通过合理的安排做这 n 个题的顺序来获得尽可能多的快乐值。你要求出 cats 在做题过程中可以获得快乐值总和的最大值。

Input

第一行包含一个整数 T ($1 \leq T \leq 10^4$)，表示一共有 T 组测试数据。

每组测试数据的第一行包含一个整数 n ($1 \leq n \leq 2 \cdot 10^5$)，表示 CF 题的总数。

接下来 n 行，每行包含 2 个整数 a_i, b_i ($0 \leq a_i, b_i \leq 10^4$)，表示双端队列中从左到右第 i 个题的难度和趣味度。

保证所有测试数据的 n 之和不超过 10^6 。

Output

对于每组测试数据，输出一个整数，表示 cats 在做题过程中可以获得快乐值总和的最大值。

Example

standard input	standard output
3	49926
1	13
314 159	12
3	
1 1	
1 2	
2 1	
3	
1 1	
2 1	
1 2	

Problem 1002. cats 的随机原神

Input file:standard input

Output file:standard output

Time limit:5 seconds

Memory limit:512 megabytes

cats 十分喜欢随机，cats 也喜欢玩原神，所以 cats 决定用随机的策略玩原神。

现在是 4202 年，原神已经由原来的 7 种元素发展到 n 种元素。在玩了几千年原神后，cats 拥有了 n 种元素的神之眼各 m 个，也就是说 cats 一共有 $n \times m$ 个神之眼。cats 十分喜欢随机过程，决定通过随机的方式来向 Su_Zipei 展示自己拥有的神之眼元素种类多。具体而言，cats 每次等概率从神之眼中不放回地取出一个，直到某一种元素的神之眼被取完为止。cats 问 Su_Zipei 取出神之眼个数的数学期望是多少？因为期望可能是一个小数，Su_Zipei 只需要告诉 cats 期望对 998244353 取模后的结果。即使是这样，Su_Zipei 仍然不会这个题目，你能帮帮她吗？

题目保证对于输入的每一组数据，取出神之眼个数的数学期望都可以被表示为一个最简分数 $\frac{a}{b}$ ，其中 b 在模 998244353 下的逆元存在，你只需要输出 $a \cdot b^{-1} \bmod 998244353$ 的结果。

Input

第一行一个整数 T ，表示测试组数 ($1 \leq T \leq 10$)。

对于每一组测试数据，只有一行，包含有两个整数 n, m ($1 \leq n, m < 998244353$)。

Output

对于每组测试数据的每次询问，输出一行一个整数，表示答案。

Example

standard input	standard output
2	5
1 5	665496238
2 2	

Problem 1003. cats 的飞机坠毁

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

飞机是一种非常危险的交通工具，是所有交通工具中事故存活率最低的交通工具之一。cats 对乘坐飞机感到非常恐惧。

在一个数轴上有 n 个编号从 1 到 n 的飞机，其中编号为 i 的飞机初始位于数轴上 $x = i$ 的位置。然后，每个飞机将会相互独立的随机选择自己的飞行方向。其中第 i 个飞机有 $\frac{a_i}{b_i}$ 的概率选择沿 x 增大方向飞行，有 $1 - \frac{a_i}{b_i}$ 的概率选择沿 x 减小方向飞行。所有飞机飞行的速率均为每秒 1 个单位长度，且都在第 0 秒开始时开始飞行。在飞行过程中，如果在任意时刻的任意坐标位置（包括非整数坐标）存在超过 1 个飞机，就会发生一次飞机坠毁。所有处在这一坐标上的飞机都会坠毁并消失(不再参与后续的飞机坠毁)。

现在，cats 想知道最后一次飞机坠毁发生的时间(按秒计算)的数学期望对 $10^9 + 7$ 取模的结果。如果没有任何飞机坠毁发生，则认为最后一次飞机坠毁发生的时间为 0。

可以证明答案一定可以被表示为一个有理数 $\frac{x}{y}$ 。其中 x 与 y 互质。你需要输出 $x \cdot y^{-1} \bmod (10^9 + 7)$ 。可以证明 $(10^9 + 7) \nmid y$ 。

Input

第一行包含一个整数 T ($1 \leq T \leq 1000$)，表示一共有 T 组测试数据。

每组测试数据的第一行包含一个整数 n ($1 \leq n \leq 400$)，表示飞机的总数。

接下来 n 行，每行包含 2 个整数 a_i, b_i ($2 \leq b_i \leq 10^9, 1 \leq a_i < b_i$)，表示第 i 个飞机选择沿 x 增大方向飞行的概率的分子和分母。保证 a_i 与 b_i 互质。

保证所有测试数据的 n^2 之和不超过 $8 \cdot 10^5$ 。

Output

对于每组测试数据，输出一个整数，表示最后一次飞机坠毁发生的时间(按秒计算)的数学期望对 $10^9 + 7$ 取模的结果。

Example

standard input	standard output
4	0
1	125000001
13 37	222222224
2	352222809
1 2	
1 2	
3	
1 2	
1 3	
2 3	
7	
405082616 465273293	
75486143 274603790	
335370224 859624599	
367950901 497839321	
185066160 809056603	
277437177 572935355	
274625651 316202992	

Problem 1004. cats 的重力拼图

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

cats 有一个有 n 行，每行有 m 个方格的重力拼图。其中第 i 行第 j 个方格坐标为 (i, j) 。重力拼图中有一个物块，初始位于坐标 (a, b) 的方格。若当前物块位于 (r, c) ，在一次操作中，cats 可以选择以下四种操作之一：

- 将重力切换为向上：将物块从当前位置移动到 $(1, c)$ 。这个过程中物块将经过所有坐标为 (i, c) ($1 \leq i \leq r$) 的方格。
- 将重力切换为向下：将物块从当前位置移动到 (n, c) 。这个过程中物块将经过所有坐标为 (i, c) ($r \leq i \leq n$) 的方格。
- 将重力切换为向左：将物块从当前位置移动到 $(r, 1)$ 。这个过程中物块将经过所有坐标为 (r, i) ($1 \leq i \leq c$) 的方格。
- 将重力切换为向右：将物块从当前位置移动到 (r, m) 。这个过程中物块将经过所有坐标为 (r, i) ($c \leq i \leq m$) 的方格。

cats 可以最多进行 142857 次操作。现在 cats 希望最大化被拼图块经过至少一次（包括初始位置和最终位置）的方格的总数。你需要告诉 cats 这个总数的最大值。

Input

第一行包含一个整数 T ($1 \leq T \leq 10^5$)，表示一共有 T 组测试数据。
每组测试数据包含一行四个整数 n, m, a, b ($1 \leq n, m \leq 10^8, 1 \leq a \leq n, 1 \leq b \leq m$)，表示重力拼图的大小和物块的初始位置。

Output

对于每组测试数据，输出一个整数，表示被拼图块经过至少一次（包括初始位置和最终位置）的方格的总数的最大值。

Example

standard input	standard output
7	4
4 1 2 1	1
1 1 1 1	4
2 2 2 2	8
3 3 1 1	14
4 4 2 3	19
5 5 3 3	399999996
100000000 100000000 100000000 100000000	

Problem 1005. cats 的二分答案

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

cats 刚刚开始学习二分答案，写出了下面这段代码（以下给出伪代码，在题目的末尾会提供 C/C++，Python，Java 语言的具体代码）

Algorithm 1 Binary Search

```
function BINARYSEARCH( $l, r, a$ )  
  while  $l \leq r$  do  
     $mid \leftarrow \lfloor \frac{l+r}{2} \rfloor$   
     $val \leftarrow a_{mid}$   
    if  $val = 2$  then  
      return  $mid$   
    end if  
    if  $val = 1$  then  
       $l \leftarrow mid + 1$   
    else  
       $r \leftarrow mid - 1$   
    end if  
  end while  
  return -1  
end function
```

给出的数组 a 下标范围为 $[1, n]$ ，且

$$a_i = \begin{cases} 1 & (i \in [1, n)) \\ 2 & (i = n) \end{cases}$$

cats 知道 $n \in [l, r]$ ，他希望通过以上这段代码找出 n 的值。可以发现这段代码有访问越界下标的可能，访问越界下标 $i > n$ 时，会得到 $a_i = 0$ 。但是在一次程序运行过程中，如果访问越界下标超过 k 次，程序就会崩溃。现在你要求出有多少不同的 n ($n \in [l, r]$) 可以让程序在不崩溃的情况下得到正确结果。

Input

第一行一个整数 T ($1 \leq T \leq 10^3$)，表示测试数据的组数。
接下来 T 行，每行三个整数 l, r, k ($1 \leq l \leq r \leq 10^{18}, 0 \leq k \leq 10^{18}$)，表示 n 的范围和程序在不崩溃的情况下访问越界下标次数的上限。

Output

对于每组测试数据，输出一个整数，表示满足条件的不同的 n 的个数。

Example

standard input	standard output
3	7
1 100 0	28
1 100 1	61
1 100 2	

Problem 1006. cats 的最小生成树

Input file:standard input

Output file:standard output

Time limit:5 seconds

Memory limit:1024 megabytes

cats 有一个有 n 个点， m 个边的可能有重边的无向图。边有边权，其中第 i 条边的边权为 i 。

在一次操作中，cats 会找出这个图的最小生成树，然后将图中所有属于最小生成树的边移除。cats 会不断重复以上操作直到图不连通为止。

现在，你需要告诉 cats 图中的每一条边是在第几次操作中被移除的。若一条边在操作结束时未被移除，你需要输出 -1 。

可以证明在以上条件下，cats 每次选择的最小生成树一定是唯一的。

注：一个有 n 个点的图的最小生成树即为这个图的所有的包含全部 n 个点和刚好 $n - 1$ 条边的连通子图中使子图所有边的边权总和最小的子图。

Input

第一行一个整数 T ($1 \leq T \leq 1000$)，表示测试数据的组数。

每组测试数据的第一行包含两个整数 n, m ($2 \leq n \leq 10^5, 1 \leq m \leq 3 \cdot 10^5$)，表示图中点和边的个数。

接下来 m 行，每行两个整数 u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$)，表示第 i 条边连接的两个点。第 i 条边的边权为 i 。

保证所有测试数据的 n 之和不超过 $5 \cdot 10^5$ ，保证所有测试数据的 m 之和不超过 $1.5 \cdot 10^6$ 。

Output

对于每组测试数据，输出一行 m 个整数。对于第 i 个数，若第 i 个边在第 x 次操作中被移除，你需要输出 x 。若第 i 个边在操作结束时未被移除，你需要输出 -1 。

Example

standard input	standard output
3	1 1 -1
3 3	-1 -1 -1
1 2	1 2 1 2 1 2
2 3	
3 1	
3 3	
1 2	
2 1	
2 1	
4 6	
1 2	
1 2	
1 3	
2 3	
1 4	
3 4	

Problem 1007. cats 的 k-xor

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

定义两个数 x, y 的 k-xor 为 x, y 在正整数 k ($k \geq 2$) 进制意义下的不进位加法。现在 cats 有两个整数 a, b , cats 算出了它们在某个进制 k 下的 k-xor 为 c 。但在 cats 计算出 c 后, cats 忘记了 k 的值。你能帮 cats 算出所有大于等于 2 的可能是 k 的不同正整数个数吗? 如果有无穷多个满足条件的 k , 输出 -1 。

注: 两个数在 k 进制下的不进位加法为, 将两个数分别写出它们的 k 进制表示, 并将两个数对应的位分别相加, 然后将每一位相加得到的结果分别对 k 取模, 将结果看做一个新的 k 进制数, 这个结果即为两个数 k 进制下不进位加法的结果。例如 $16 = (121)_3$ 和 $8 = (022)_3$ 在 3 进制下的不进位加法的结果即为 $(110)_3 = 12$ 。

Input

第一行包含一个整数 T ($1 \leq T \leq 100$), 表示一共有 T 组测试数据。
每组测试数据包含一行三个整数 a, b, c ($0 \leq a, b, c \leq 10^9$), 表示参与 k-xor 运算的两个数和运算结果。

Output

对于每组测试数据, 输出一个整数, 表示所有大于等于 2 的可能是 k 的不同正整数个数。如果有无穷多个满足条件的 k , 输出 -1 。

Example

standard input	standard output
5	1
3 5 6	2
16 8 12	-1
0 0 0	3
21 21 0	0
123 456 789	

Problem 1008. cats 的数据结构

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

cats 刚刚学习了堆这个数据结构，不过他觉得二叉树的情况太简单了，想把它推广到任意结构的有根树上。

现在 cats 有一个以 1 为根节点的树，树上的每个节点 i 都有两个整数权值 a_i, b_i 。现在你需要为每个节点确定权值 a_i, b_i ，使得这个树满足以下四个条件。

- 1. 对于任意的 i ($1 \leq i \leq n$)，都有 $1 \leq a_i, b_i \leq 10^9$ 。
- 2. 对于任意的一对 u, v ($1 \leq u, v \leq n, u \neq v$)，都有 $a_u \neq a_v$ 且 $b_u \neq b_v$ 。
- 3. 对于任意的一对 u, v ($1 \leq u, v \leq n, u \neq v$)，若 u 是 v 的祖先，则 $a_u > a_v$ 且 $b_u > b_v$ 。
- 4. 对于任意的一对 u, v ($1 \leq u, v \leq n, u \neq v$)，若 $a_u > a_v$ 且 $b_u > b_v$ ，则 u 是 v 的祖先。

可以证明存在至少一种合法的构造方式，你需要给出所有合法构造方式中使序列 $a_1, b_1, a_2, b_2, \dots, a_n, b_n$ 字典序最小的解。

注 1: 在一个有根树上，节点 u 为节点 v 的祖先，当且仅当所有的以 v 为起点，以根节点为终点的树上路径都经过 u 。

注 2: 两个相同长度的序列 a, b (a 与 b 至少存在一个位置不相同) 的字典序比较方式为，对于最小的满足 $a_i \neq b_i$ 的 i ，若 $a_i < b_i$ 则序列 a 的字典序更小，若 $a_i > b_i$ 则序列 b 的字典序更小。

Input

第一行一个整数 T ($1 \leq T \leq 2000$)，表示测试数据的组数。

接下来包含 T 组测试数据。

每组数据第一行一个整数 n ($2 \leq n \leq 2 \cdot 10^5$)，表示 cats 给出的树的节点个数。

接下来一行包含 $n - 1$ 个整数 p_2, p_3, \dots, p_n ($1 \leq p_i \leq n$)，依次表示编号为 2 到 n 的节点的父亲节点编号。保证输入的会构成一颗合法的以编号为 1 的节点作为根节点的有根树。

保证所有测试数据的 n 之和不超过 10^6 。

Output

对于每一组测试数据输出一行 $2 \cdot n$ 个由空格隔开的整数 $a_1, b_1, a_2, b_2, \dots, a_n, b_n$ ，表示满足条件的字典序最小的序列。

Example

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

Problem 1009. cats 的凸包计算

Input file: standard input
Output file: standard output
Time limit: 12 seconds
Memory limit: 512 megabytes

定义一个排列 p_1, p_2, \dots, p_n 的面积为：将排列中的每个数 p_i 看做二维坐标平面上的一个点 (i, p_i) ，这 n 个点的凸包的面积。

现在 cats 有一个长为 n 的数组 a_1, a_2, \dots, a_n ，其中每个数均为一个 1 到 n 之间的正整数或 -1 。你要求出将所有 -1 替换成 1 到 n 之间的正整数后，形成的所有排列的面积之和对 $10^9 + 7$ 取模的结果。

可以证明答案一定可以被表示为一个有理数 $\frac{x}{y}$ 。其中 x 与 y 互质。你需要输出 $x \cdot y^{-1} \bmod (10^9 + 7)$ 。
可以证明 $(10^9 + 7) \nmid y$ 。

注1：一个长为 n 的数组 p_1, p_2, \dots, p_n 是一个排列，当且仅当 1 到 n 之间的每一个正整数都在 p_1, p_2, \dots, p_n 中出现且仅出现一次。

注2： n 个点 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ 的凸包为：所有满足存在 n 个范围在 $[0, 1]$ 之间的实数 w_1, w_2, \dots, w_n ，使得 $\sum_{i=1}^n w_i = 1$ ， $\sum_{i=1}^n w_i \cdot x_i = x$ ， $\sum_{i=1}^n w_i \cdot y_i = y$ 的点 (x, y) 在二维坐标平面上组成的图形。

Input

第一行包含一个整数 T ($1 \leq T \leq 1000$)，表示一共有 T 组测试数据。
每组测试数据的第一行包含一个整数 n ($1 \leq n \leq 50$)，表示数组 a 的长度。
每组测试数据的第二行包含 n 个整数 a_1, a_2, \dots, a_n ($-1 \leq a_i \leq n, a_i \neq 0$)，表示数组 a 。
保证对于任意的 $1 \leq i < j \leq n$ ，若 a_i 和 a_j 均不为 -1 ，则 $a_i \neq a_j$ 。
保证所有测试数据的 n^3 之和不超过 $5 \cdot 10^5$ 。

Output

对于每组测试数据，输出一个整数，表示形成的所有排列的面积之和对 $10^9 + 7$ 取模的结果。

Example

standard input	standard output
5	0
3	9
1 2 3	500000006
4	0
2 -1 -1 3	99546
4	
1 2 -1 3	
1	
1	
7	
-1 -1 -1 -1 -1 -1 -1	

Problem 1010. cats 的集合 1

Input file: standard input
Output file: standard output
Time limit: 12 seconds
Memory limit: 1024 megabytes

cats 需要维护一个可重集 S 。开始时，可重集中存在 n 个元素 p_1, p_2, \dots, p_n 。然后，cats 对集合进行了 m 次操作。操作分为以下 5 种：

- 1 a: 在可重集中加入元素 a 。
- 2 a: 将可重集中的每个元素都变为其与 a 按位或运算的结果。
- 3 a: 将可重集中的每个元素都变为其与 a 按位与运算的结果。
- 4 a: 将可重集中的每个元素都变为其与 a 按位异或运算的结果。
- 5 a: 查询可重集中每一个元素与 a 按位异或运算结果中的最大值。

Input

第一行一个整数 T ($1 \leq T \leq 1000$)，表示测试数据的组数。
接下来包含 T 组测试数据。
每组数据第一行为两个整数 n, m ($1 \leq n, m \leq 2 \times 10^5$)，分别表示开始时 S 中元素的个数，cats 需要进行的操作次数。
接下来一行包含 n 个整数 p_1, p_2, \dots, p_n ($0 \leq p_i < 2^{31}$)，表示开始时 S 中的元素。
接下来 m 行每行两个整数 opt, a ($1 \leq opt \leq 5, 0 \leq a < 2^{31}$)，表示一次操作。保证每组测试数据中至少包含一次操作 5。
保证所有测试数据的 n 之和不超过 $1.5 \cdot 10^6$ ，保证所有测试数据的 m 之和不超过 $1.5 \cdot 10^6$ 。

Output

对于每个 5 操作输出一行一个正整数，表示可重集中元素所有元素与 a 按位异或运算结果的最大值。

Example

standard input	standard output
1	5
5 10	8
1 2 3 4 5	8
5 1	12
1 10	10
5 2	14
2 3	
5 3	
3 12	
5 4	
4 7	
5 5	
5 13	

Problem 1011. cats 的集合 2

Input file: standard input
Output file: standard output
Time limit: 6 seconds
Memory limit: 512 megabytes

cats 有一个大小为 n 的可重集合 S ，但是 cats 却不知道 S 中的每个数分别是多少，只知道其中的第 i 个数在 $[0, a_i]$ 中。定义 S 的价值为集合中所有数的异或和，cats 认为一个集合有意义，当且仅当这个集合的价值在 $[L, R]$ 中。cats 想知道在 S 的所有可能方案中，有多少方案满足 S 是有意义的，由于 cats 并不确定 L, R 的具体值，所以 cats 会进行多次询问。因为方案数可能会很大，你只需要输出答案对 998244353 取模后得到的结果。

Input

第一行一个整数 T ，表示测试组数 ($1 \leq T \leq 10$)。

对于每一组测试数据，第一行有两个整数 n, m ($1 \leq n, m \leq 10^6$)。

第二行 n 个整数，第 i 个整数表示 a_i ($0 \leq a_i \leq 10^{18}$)。

接下来有 m 行，每行两个整数 L, R ($0 \leq L \leq R \leq 10^{18}$)。

保证 $\sum n \leq 2 \times 10^6, \sum m \leq 2 \times 10^6$ 。

Output

对于每组测试数据的每次询问，输出一行一个整数，表示答案。

Example

standard input	standard output
2	2
2 2	3
1 2	432
0 0	
1 2	
4 1	
2 3 5 7	
2 7	

Problem 1012. cats 的电脑中毒

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

cats 的电脑被电脑病毒感染了。cats 的电脑可视为所有 2^n 个长度为 n 的，只包含 0,1 两种字符的字符串的集合。每一个字符串为一个节点。如果两个节点的字符串之间有且仅有一个位置的字符不同，则认为这两个节点是相邻的。例如节点 1111 和节点 1110 是相邻的，但节点 101 和节点 000 不是相邻的。

在第 0 秒时，电脑病毒感染了三个节点 S_1, S_2, S_3 。从第 1 秒开始，每一秒每个被电脑病毒感染的节点都会感染与它相邻的所有未被感染的节点。当所有节点都被感染时，cats 的电脑就损坏了。现在，你要求出 cats 的电脑损坏的时间（按秒计算）。

Input

第一行包含一个整数 T ($1 \leq T \leq 10^4$)，表示一共有 T 组测试数据。

每组测试数据的第一行为一个整数 n ($2 \leq n \leq 2 \cdot 10^5$)，表示 cats 的电脑的节点的字符串长度。

接下来 3 行，每行包含一个长为 n 的，只包含 0,1 两种字符的字符串 S_i ，表示初始被电脑病毒感染的节点。保证对于任意的 $1 \leq i < j \leq 3$ ， S_i 与 S_j 不完全相同。

保证所有测试数据的 n 之和不超过 10^6 。

Output

对于每组测试数据，输出一个整数，表示 cats 的电脑损坏的时间（按秒计算）。

Example

standard input	standard output
4	1
2	2
01	4
10	7
11	
3	
011	
000	
110	
7	
1011001	
0101110	
0110100	
10	
0110110100	
1100100010	
0001110111	