

第三届环球杯

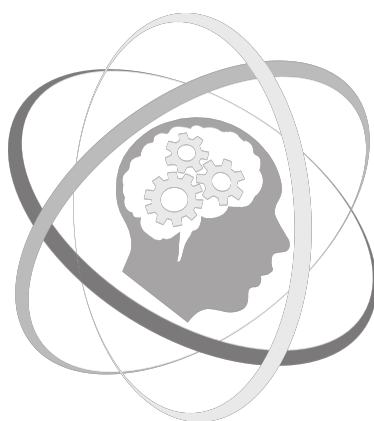


第 2 赛段：Zielona G'ora

2024 年 6 月 15-16 日

该问题集应包含 13 个问题，共 19 页。

根据



POTYCZKI ALGORYTMICZNE

Potyczki Algorytmiczne / 算法约定



问题 A. 有趣的路径

输入文件：标准输入
输出文件：标准输出
时间限制：4 秒 内存限制：
1024 兆字节

给我们一个有向图，图中有 n 个顶点和 m 条边。图中顶点的编号从 1 到 n ，且每条边都从数字较小的顶点通向数字较大的顶点。在以下情况下，我们称路径序列为有趣的路径序列：

- 每条路径都以顶点 1 为起点，以顶点 n 为终点、
- 每条路径都至少包含一条之前的路径中没有的边。

最长的有趣路径序列的长度是多少？

输入

输入的第一行包含两个整数 n 和 m ($2 \leq n \leq 10^6$ ， $0 \leq m \leq 10^6$)，表示所考虑图形的顶点和边的数量。

接下来的 m 行中，每一行都包含两个整数 a 和 b ($1 \leq a < b \leq n$)，描述从顶点 a 到顶点 b 的有向边。每对 (a, b) 在输入中最多出现一次。

输出

输出结果应包含一个整数，即有趣路径序列的最大长度。

实例

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



备注

在第一个例子中，长度为 4 的有趣路径序列如下：

- $1 \rightarrow 3 \rightarrow 5$ （首次使用边 $1 \rightarrow 3$ ）、
- $1 \rightarrow 2 \rightarrow 3 \rightarrow 5$ （首次使用边 $1 \rightarrow 2$ ）、
- $1 \rightarrow 3 \rightarrow 4 \rightarrow 5$ （首次使用边 $3 \rightarrow 4$ ）、
- $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$ （首次使用边 $2 \rightarrow 4$ ）。

在第二个例子中，没有从顶点 1 到顶点 5 的路径。



问题 B. 怒吼 III

输入文件：标准输入
输出文件：标准输出
时间限制 6 秒 内存限制
1024 兆字节

在 "百字节森林 "的中央，生长着一棵不寻常的树，树上栖息着咆哮的蜗牛。这棵树由 n 个顶点组成，这些顶点的编号从 1 到 n ，连接在一个有 $n - 1$ 条边的连通图中。开始时，树的每个顶点上最多有一只雄性咆哮蜗牛。

一会儿，一只雌鸟会出现在树的某个顶点，并开始吼叫。每当雌鸟吼叫一次，一只雄鸟就会沿着一条边移动到离雌鸟更近的相邻顶点。但是，如果目标顶点上已经有另一只雄性，或者雄性已经和雌性在同一个顶点上，那么雄性就不能移动。如果在下一次吼叫后没有雄性可以移动，雌性就会停止吼叫。

输入的信息是对咆哮蜗牛居住的树木的描述。对于每个顶点，您还会收到该顶点是否存在雄性蜗牛的信息。对于每个顶点，如果雌性蜗牛在该顶点，请确定它能吼叫的最大次数。我们假设雄性蜗牛的移动方式会使吼叫次数最大化。

输入

输入的第一行包含一个整数 n ($1 \leq n \leq 2 \cdot 10^5$)，代表树的顶点数。

输入的第二行包含一个由 n 个字符 0 和 1 组成的单词。如果该单词的第 i 个字符为 1，则树的第 i 个顶点上有一只雄性蜗牛；如果该单词的第 i 个字符为 0，则树的第 i 个顶点上没有雄性蜗牛。如果该字的第 i 个字符为 0，则树的第 i 个顶点没有雄性蜗牛。

接下来的 $n - 1$ 行包含两个整数 a_i 和 b_i ($1 \leq a_i, b_i \leq n; a_i \neq b_i$)，表示顶点 a_i 和 b_i 由一条边连接。

输出

输出结果应在单行中包含 n 个整数；第 i 个整数应表示雌鼠在树的第 i 个顶点时能发出的最大吼声次数。

示例

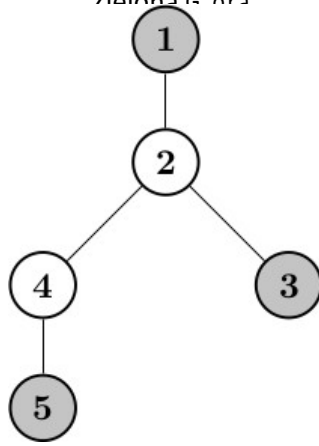
标准输入	标准输出
5 10101 1 2 2 3	2 2 2 3 3



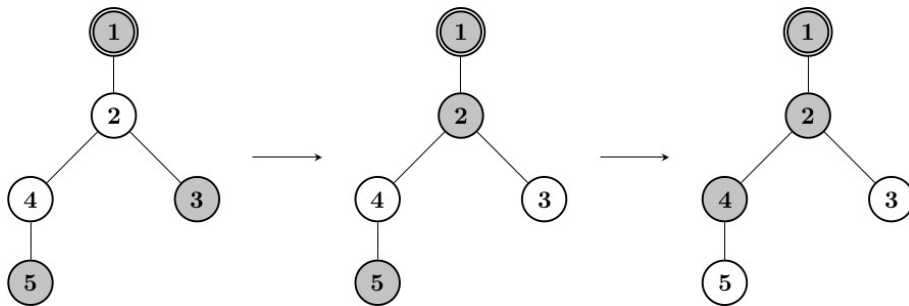
2 4	Zielona G´ora
4 5	

备注

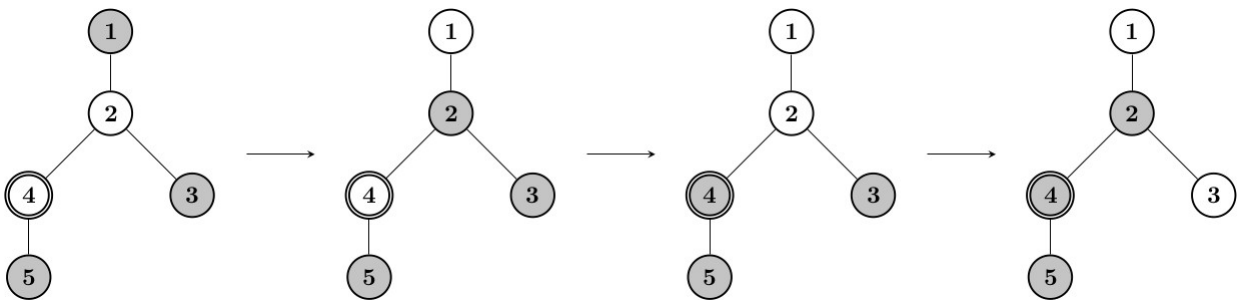
样本测试中的树如下所示。雄性位于灰色阴影的顶点。



如果雌性在第一个顶点，那么它最多只能吼叫两次，比如先把雄性从第三个顶点吸引到第二个顶点，然后再把雄性从第五个顶点吸引到第四个顶点：



如果雌鸟在第四顶点，只要第五顶点的雄鸟不动，雌鸟最多可以吼叫三次：





问题 c. 雷达

输入文件：标准输入
输出文件：标准输出
时间限制 2 秒 内存限制
1024 兆字节

给我们一个大小为 $n \times n$ 的棋盘，其中 n 为奇数。对于棋盘上的每一格，都给出了在其中建造雷达的成本。雷达覆盖一个正方形区域（边与棋盘边平行），边长为 n ，中心位于雷达所在的单元格。你的任务是尽量减少建造一定数量雷达的总成本，使电路板的每个区域都至少被覆盖一次。此外，您还必须解决多个测试案例。

输入

标准输入的第一行包含一个整数 t ($t \geq 1$)，表示测试用例的数量。然后，以 t 块为单位，依次对后面的测试用例进行描述。

测试用例说明的第一行包含一个整数 n ($1 \leq n \leq 499$; n 为奇数)，表示电路板的尺寸。

在接下来的 n 行中，将给出电路板说明。

在第 i 行中，有 n 个整数 $a_{i,1}, a_{i,2}, \dots, a_{i,n}$ ($1 \leq a_{i,j} \leq 10^9$)，其中 $a_{i,j}$ 表示在位于棋盘第 i 行第 j 列的单元中建造雷达的成本。

保证一个文件中 n^2 的值之和不超过 500 000。

输出

输出结果应包含 t 行。第 i 行应包含一个整数--在第 i 个测试用例中建造雷达的最低成本。

示例

标准输入	标准输出
2	1
3	5
1 1 1	
1 1 1	
1 1 1	
5	
8 5 2 8 3	
5 6 9 7 3	
7 8 9 1 4	
8 9 4 5 5	
2 8 6 9 3	



备注

在第一个抽样测试中，只需要建造一个雷达。在第二个样本测试中，值得建造三个雷达。雷达的最佳位置及其覆盖区域如下图所示：

Zielona G'ora

1	1	1
1	1	1
1	1	1

8	5	2	8	3
5	6	9	7	3
7	8	9	1	4
8	9	4	5	5
2	8	6	9	3



问题 D. Xor 分区

输入文件：标准输入
输出文件：标准输出
时间限制 2 秒 内存限制
1024 兆字节

给出一个整数序列 a_1, a_2, \dots, a_n 。序列 a_1, a_2, \dots 的分区定义为其连续区间的集合，使得每个元素都恰好属于一个区间。 a_n 的分区定义为其连续区间的集合，使得每个元素恰好属于一个区间。

该序列中连续区间的 xor 定义为该区间中数字的比特 xor。序列分区的值定义为区间 xor 的乘积。

计算序列 a_1, a_2, \dots 的所有可能分区的值之和。 a_n 。由于这个数字可能非常大，因此只需提供它除以 10 的余数⁹ + 7 即可。

输入

输入的第一行包含一个整数 n ($1 \leq n \leq 3 \cdot 10^5$)，代表序列的长度。

第二行包含 n 个整数 a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^{18}$) 代表序列。

输出

输出结果应包含一个整数，即给定序列所有可能分区的值之和除以 10 后的余数⁹ + 7。

示例

标准输入	标准输出
4 7 3 1 2	170

备注

序列的可能分区是

- [7, 3, 1, 2] - 值为 7、
- [7, 3], [1, 2] - 值为 $4 \cdot 3 = 12$ 、
- [7]、[3, 1, 2] - 值为 $7 \cdot 0 = 0$ 、
- [7, 3, 1], [2] - 值为 $5 \cdot 2 = 10$ 、
- [7, 3], [1], [2] - 值为 $4 \cdot 1 \cdot 2 = 8$ 、
- [7]、[3]、[1, 2] - 值为 $7 \cdot 3 \cdot 3 = 63$ 、



- [7]、[3, 1]、[2] - 值为 $7 - 2 - 2 = 28$ 、 Zielona G'ora
- [7]、[3]、[1]、[2] - 值为 $7 - 3 - 1 - 2 = 42$ 。

分区值的总和是 $7 + 12 + 0 + 10 + 8 + 63 + 28 + 42 = 170$ 。



问题 E. 模式搜索 II

输入文件：	标准输入
输出文件：	标准输出
时间限制	4 秒 内存限制
	1024 兆字节

让我们定义斐波那契单词的无穷序列 $S_0, S_1, S_2, S_3 \dots$ 如下：

- $S_0 = b$
- $S_1 = a$
- $S_i = S S_{i-1} S_{i-2}$ for $i \geq 2$

该序列的前几个字如下：

- $S_0 = b$
- $S_1 = a$
- $S_2 = ab$
- $S_3 = aba$
- $S_4 = abaab$
- $S_5 = abaababa$
- $S_6 = abaababaabaab$

不难发现，每个字（ S_0 除外）都是下一个字的前缀。因此，我们也可以定义一个无限斐波那契单词 S ，其中第 i 个字符是这些单词无限序列（ S_0 除外）中至少有 i 个字符的第 i 个字符。

给你一个仅由字符 "a" 和 "b" 组成的单词 t 。您的任务是找出 S 的**子串**中包含 t 的最短单词 s ，并输出其长度。

输入

标准输入包含一行单词 t ，仅由字母 "a" 和 "b" 组成。

输出

输出结果应包含一个整数，表示描述字可能的最小长度 s 。

示例

标准输入	标准输出
------	------



aabbaab

Zielona G'ora

备注

S 中不包含长度为 7 或更短的子串 t 。然而，它确实包含子串 aababaab，我们可以从中删除第四个字符，从而得到 t 。



问题 F. 瀑布矩阵

输入文件：标准输入
输出文件：标准输出
时间限制 8 秒 内存限制
1024 兆字节

Radewoosh 最近发现了一种特殊类型的矩阵。如果每个单元格中的数字都大于或等于正下方和右边的数字（如果存在的话），我们就说包含整数的正方形矩阵是 **瀑布矩阵**。换句话说，大小为 $n \times n$ 的瀑布矩阵 M 是这样的一个矩阵：对于满足 $1 \leq i \leq n$ 和 $1 \leq j \leq n-1$ 的所有数对 (i, j) ，我们有 $M_{i,j} \geq M_{i,j+1}$ 和 $M_{j,i} \geq M_{j+1,i}$

Radewoosh 想要创建一个大小为 $n \times n$ 的瀑布矩阵。遗憾的是，他不可能在每个选定的单元格中准确输入他想要的值。因此，他决定创建一个矩阵，使他想输入的数值与实际输入的数值之间的绝对差值之和最小。

形式上，Radewoosh 有一个三元组数字列表 (a_i, b_i, c_i) 并希望选择一个瀑布矩阵 M 使 $\sum_i |Ma_{ii} - c_i|$ 值最小。帮助他，并输出所提及的最小值。

如果 Radewoosh 最佳地选择了矩阵，那么就能达到总和。

输入

标准输入的第一行包含一个整数 n ($1 \leq n \leq 200\,000$)，表示 Radewoosh 想要绘制的矩阵大小和他选择的字段数。

在接下来的 n 行中，每行都有三个整数 a_i, b_i 和 c_i ($1 \leq a_i, b_i \leq n; 1 \leq c_i \leq 10^9$)，如任务所述。

可以保证，对于 $i \neq j$ ， $(a_i, b_i) \neq (a_j, b_j)$

输出

输出结果应包含任务中描述的一个整数。

示例

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

备注



Radewoosh 可以选择的最佳矩阵之一如下：

9	7	5	5	5
6	6	5	5	3
5	3	3	3	3
3	2	2	1	1
2	1	1	1	1

对于上述矩阵，我们可以计算出如下结果：

$$|m_{1,3} - 5| + |m_{3,2} - 1| + |m_{3,3} - 3| + |m_{4,4} - 1| + |m_{3,5} - 4| = |5 - 5| + |3 - 1| + |3 - 3| + |1 - 1| + |3 - 4| = 0 + 2 + 0 + 0 + 1 = 3$$



问题 G. 拼图 II

输入文件：	标准输入
输出文件：	标准输出
时间限制	3 秒 内存限制
	1024 兆字节

拉德沃什收到了父母寄来的新谜题。这次的谜题包含两个循环的球序列。每个序列包含 n 个球，每个球要么是白色的，要么是黑色的。两个序列总共包含 n 个白球和 n 个黑球。为了描述这个谜题，我们将使用字符串 a 和 b -- 字符串 a 中的连续字符表示第一个序列中连续球的颜色，字符串 b 中的连续字符表示第二个序列中连续球的颜色。两个序列中的位置编号从 1 到 n 。

这道谜题还涉及到一个重要的数字 k 。在一次移动中，拉德沃什可以从第一个序列中选择一个由整整 k 个球组成的循环间隔，并从第二个序列中选择一个由整整 k 个球组成的循环间隔，然后将它们互换。目的是使两个序列都是单色的，即第一个序列中的所有球都是同一种颜色（全黑或全白），而第二个序列中的所有球都是同一种颜色。

帮助拉德沃什在最多 n 步内解开谜题。可以证明这总是可能的。

输入

标准输入的第一行包含两个整数 n 和 k ($2 \leq n \leq 300\,000$; $1 \leq k \leq n - 1$)，如任务所述。

下一行包含一个字符串 a ，正好由 n 个字符组成。如果第 i 个字符是 "B"，那么第一个序列中的第 i 个球是白色的（波兰语中 "白色" 的意思是 "biały"）。否则，如果字符是 "C"，则球是黑色的（波兰语中 "black" 的意思是 "czarny"）。

下面一行包含一个字符串 b ，正好由 n 个字符组成，同样描述了第二个球序列。

字符串 a 和 b 总共包含 n 个 "B" 字符和 n 个 "C" 字符。

输出

输出的第一行应包含一个数字 r ($0 \leq r \leq n$)，表示您要走的步数。

接下来，输出应包含 r 行。每行包含两个整数。这些行中第 i 行的数字 c_i 和 d_i ($1 \leq c_i, d_i \leq n$) 应表明，您正在从位置 c_i 开始的第一个序列中选择一个循环间隔，并从位置 d_i 开始的第二个序列中选择一个循环间隔。

如果 $c_i + k - 1 \leq n$ ，则表示位置 $c_i, c_i + 1, \dots, c_i + k - 2, c_i + k - 1$ 在第一个序列中。如果 $c_i + k - 1 \geq n + 1$ ，则表示位置 $c_i, c_i + 1, \dots, n - 1, n, 1, 2, \dots, c_i + k - 2 - n, c_i + k - 1 - n$ 。 d_i 的值也有类似的含义。

在走完您所描述的所有棋步后，第一个序列中的所有小球应该具有相同的颜色，而第二个序列中



的所有小球应该具有相同的颜色。请注意，您不必尽量减少移动次数，只要最多移动 n 次即可。

示例

标准输入	标准输出
6 3	2
BCCBCC	1 3
英国广播公司BBC	5 1



备注

最初的球序如下（第一球序绘制于上，第二球序绘制于下）：



第一步选择这两个区间：



然后交换：



第二步选择这两个区间：



然后交换：



请注意，将所有黑球移至第一序列，将所有白球移至第二序列也是正确的。



问题 H. 天气预报

输入文件：标准输入
输出文件：标准输出
时间限制 3 秒 内存限制 1024 兆字节

Radewoosh 是一位气象学家。通过他的研究站，他知道了未来 n 天的预测气温。每天的温度（以拜腾海特度表示）总是一个正整数。

Radewoosh 现在需要在电视上播报天气预报。为了让厌倦了凉爽气候的人们高兴起来，他决定营造出一种天气将相当温暖的印象。他决定把未来的 n 天分成 k 个非空区间，这样每天正好属于一个区间。然后，在天气预报中，Radewoosh 将只提供 k 个数字，即属于各自区间的各天气温的算术平均值。

Radewoosh 知道，收看天气预报的观众非常害怕低温。因此，他想做一个除法，最大限度地提高他提供的最冷时段的平均气温（如果其余时段的平均气温都不低，我们就称该时段为最冷时段）。请帮助他做出这样的除法！

输入

输入的第一行包含两个整数 n 和 k ($1 \leq n \leq 2 \cdot 10^5$, $1 \leq k \leq n$)，分别代表 Radewoosh 获得温度预报的天数和他希望将温度预报分成的时段数。

输入的第二行包含 n 个整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq 1000$) 表示连续几天的温度，单位为拜腾海特度。

输出

输出结果应包含一个实数，代表最冷时段的最高平均温度。如果绝对误差不超过 10^{-4} ，答案将被接受。

示例

标准输入	标准输出
7 3 1 3 1 2 2 2 1	1.6666666666667

备注

天数可分为以下几个时段： $[1, 3, 1]$, $[2]$, $[2, 2, 1]$. 连续时段的平均气温分别为³、2 和³，因此最低气温为³。这
除法使单个时段的最小平均气温达到最大值。



问题 1. 雇佣军

输入文件：	标准输入
输出文件：	标准输出
时间限制	5 秒 内存限制
	1024 兆字节

在神奇的拜特大陆上，有 n 座城市排成一行，从左到右编号为 1 到 n ，它们之间有 $n - 1$ 条道路。每条道路都连接着两个相邻的城市。由于地形多山，每条路只能从数字较小的城市通往数字较大的城市。换句话说，这些通道都是单向的。

每个城市都生活着一名雇佣兵。这些雇佣兵可以用两个数字来描述，分别代表他们的力量和魔法知识。生活在第 i 座城市的雇佣兵由 s_i 和 m_i 描述。

每条路都有一家商店。每家商店都提供一定的物品清单，每种物品都有其对力量的加成和对魔法知识的加成。这些加成显示了相应英雄数据的加成。更准确地说，对于第 i 家商店（即连接第 i 个城市和第 $(i + 1)$ -st 个城市），我们就知 $(s_{i,1}^r, m_{i,1}^r), (s_{i,2}^r, m_{i,2}^r), \dots, (s_{i,r_i}^r, m_{i,r_i}^r)$ ，代表

该商店提供的物品的奖励。沿路旅行的佣兵每经过一家商店，最多只能购买一件物品。佣兵可以同时使用的物品数量不受限制，所有物品的奖励都会加到佣兵自身的统计数据中并累积起来。

怪物可以攻击城市。在第 i 个攻击场景中，怪物由其攻击城市的编号（表示为 v_i ）和值 a_i 、 b_i 和 c_i 描述。它们表明，如果 $a_i - S_j + b_i - M_j \geq c_i$ ，来自第 j 个城市的雇佣兵就能打败怪物，其中 S_j 和 M_j 的值是雇佣兵的力量和魔法知识，并考虑了他从自己的城市前往怪物攻击的城市途中购买物品获得的奖励。具体而言，这意味着 $j \leq v_i$ 必须成立。在这里，我们允许 $j = v_i$ --那么雇佣兵将没有机会购买任何物品。否则，他将在从第 j 个城市到第 v_i -th 个城市的途中，在经过的每个商店最多购买一件物品。

作为拜特兰国王的顾问，你的任务是为所有怪物攻击方案制定行动计划。更确切地说，对于每个攻击方案 i ，您必须找到最大的城市编号 j ，使得 $j \leq v_i$ ，并且来自第 j 个城市的雇佣兵能够选择购买的物品，从而打败怪物。请注意，所考虑的攻击情况只是理论上的，佣兵不会移动或获取任何物品。

输入

标准输入的第一行包含一个整数 n ($2 \leq n \leq 200\,000$)，表示城市数量。接下来的 $2n - 1$ 行描述城市和城市间道路上的商店。

对于满足 $1 \leq i \leq n$ 的每个 i ，其中第 $(2i - 1)$ 行包含两个整数 s_i 和 m_i ($0 \leq s_i, m_i \leq 10^9$)，表示居住在第 i 个城市的雇佣兵的实力和魔法知识。



对于满足 $1 \leq i \leq n-1$ 的每个 i ，这些行中的第 $(2i)$ - 行以一个整数 r_i ($1 \leq r_i \leq 500\,000$) 开始，表示位于连接第 i 行和第 $(i+1)$ - 行的道路上的商店提供的商品数量。

st个城市。然后，在同一行中，有一个由 $2r_i$ 个整数组成的序列 $s_{i,1}^r, m_{i,1}^r, s_{i,2}^r, m_{i,2}^r, \dots, s_{i,r_i}^r, m_{i,r_i}^r$ ($0 \leq s_{i,j}^r, m_{i,j}^r \leq 5\,000$)，表明连续的

该商店提供的商品。 r_i 的所有数值总和不超过 $500\,000$ 。

下一行包含一个整数 q ($1 \leq q \leq 200\,000$)，表示怪物攻击场景的数量。

随后的 q 行中每一行都包含一个攻击场景的描述。其中第 i 行包含四个整数 v_i, a_i, b_i 和 c_i ($1 \leq v_i \leq n; 0 \leq a_i, b_i \leq 10^9; a_i + b_i \geq 1; 1 \leq c_i \leq 10^{18}$)，如问题陈述所述。



输出

输出结果应由 q 行组成：其中第 i 行应包含一个整数，表示雇佣兵在第 i 个攻击场景中能够击败怪物的最大城市编号，如果没有雇佣兵能够击败怪物，则输出结果为 -1。

示例

标准输入	标准输出
3	1
1 1	2
2 1 2 1 2	3
3 2	3
5 1 5 4 3 3 4 5 1 1 2	2
4 5	2
1 2	1
1 1 1 1	-1
2 1 1 1	1
3 1 1 1	-1
3 1 1 9	2
3 2 2 20	2
3 1 2 18	
3 1 2 19	
3 1 2 20	
3 0 1 8	
2 1 0 4	
2 1 0 3	
2 1 0 2	

备注

第一家商店出售两件商品，这两件商品由相同的配对 $(1, 2)$ 和 $(1, 2)$ 描述。第二家商店出售五件商品，描述的商品对分别是 $(1, 5)$ 、 $(4, 3)$ 、 $(3, 4)$ 、 $(5, 1)$ 和 $(1, 2)$ 。

在前三个攻击场景中，怪物分别攻击第一个、第二个和第三个城市。在所有这些情况下，已在该城市的雇佣兵都能打败怪物。

让我们来看看第六、第七和第八个攻击方案。在每种情况下，怪物都以相同的参数 a_i 和 b_i 攻击第三座城市。在第六种情况下，当 $c_i = 18$ 时，如果第二座城市的雇佣兵选择了一对 $(1, 5)$ 所描述的物品，那么怪物就会被击败。那么他的力量为 $3 + 1 = 4$ ，魔法知识为 $2 + 5 = 7$ ，结果为 $1 - 4 + 2 - 7 = 18 \geq 18$ 。在第七种情况下，当 $c_i = 19$ 时，只有第一个城市的佣兵才能打败怪物。通过购买 $(1, 2)$ 和 $(1, 5)$ 这两对描述的物品，他的力量为 $1 + 1 + 1 = 3$ ，魔法知识为 $1 + 2 + 5 = 8$ ，结果为 $1 - 3 + 2 - 8 = 19 \geq 19$ 。在第八种情况下，当 $c_i = 20$ 时，没有雇佣兵能打败怪物。请注意，在第六种情况下 ($c_i = 18$)，第一个雇佣兵也能打败怪物，但我们感兴趣的是最大的城市数字，所以答案是 2。



问题 J. 多边形 II

输入文件：标准输入
输出文件：标准输出
时间限制：3 秒 内存限制：1024 兆字节

科斯特卡想为自己建造一个大多边形。他随机订购了 n 条线段。第 i 个线段的长度将是范围 $(0, 2^a i)$ 的均匀分布中随机抽取的**实数**。各段的长度是独立抽取的。科斯特卡感兴趣的是用所有这些线段构造一个非退化 n 边多边形的概率。请帮他计算一下。

可以证明，对于下面给出的约束条件，结果可以表示为一个有理数 $\frac{l}{m}$ 使分母 m 不能被 $10^9 + 7$ 整除⁹。在给定参数 a_i

应输出该分数除以 $10^9 + 7$ 的余数⁹，即一个数字 x ，使得 $m - x \equiv l$ modulo $10^9 + 7$ 。

注意：请记住，线段的长度不是取自区间 $(0, a_i)$ ，而是取自 $(0, 2^a i)$ 。

输入

输入的第一行包含一个正整数 n ($3 \leq n \leq 1000$)，表示有序线段的数量。第二行包含一个由 n 个整数组成的序列 a_1, \dots, a_n ($0 \leq a_i \leq 50$)，表示用于绘制单个线段长度的参数。

输出

输出结果应包含一个整数--概率除以 $10^9 + 7$ ，即有可能从科斯特卡排序的随机线段中构造出一个非退化的 n 边多边形。

示例

标准输入	标准输出
3 0 2 0	166666668

备注

所求概率恰好是¹。 6



问题 K. 权力划分

输入文件：标准输入
输出文件：标准输出
时间限制 5 秒 内存限制
1024 兆字节

我们得到一个 n 个非负整数序列 a_1, a_2, \dots, a_n 。根据这个序列，我们创建了一个整数序列 b_1, b_2, \dots, b_n ，其中对于每个 i ， $b_i = 2^{a_i}$ 。

序列 b_1, b_2, \dots 的分部称为其连续区间的集合。每个元素恰好属于一个区间。如果每个区间中的数字之和是 2 的幂次（指数为整数），我们就称这种除法为好除法。

你的任务是计算序列 b_1, b_2, \dots 的好除数。由于这个数可能非常大，因此只需提供它除以 $10^9 + 7$ 的余数。

输入

输入第一行包含一个整数 n ($1 \leq n \leq 3 \cdot 10^5$)，代表序列 a_1, a_2, \dots 的长度（因此也代表序列 b_1, b_2, \dots 的长度）。

输入的第二行包含 n 个整数 a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^6$)。

输出

输出结果应包含一个整数，表示除以 $10^9 + 7$ 后的余数，即 b_1, b_2, \dots, b_n 。

示例

标准输入	标准输出
5 2 0 0 1 1	6

备注

序列 b_1, b_2, \dots, b_n 的抽样检验结果是 4, 1, 1, 2, 2。它的好除法是

- [4], [1], [1], [2], [2],
- [4], [1, 1], [2], [2],
- [4], [1], [1], [2, 2],
- [4], [1, 1], [2, 2],
- [4], [1, 1, 2], [2],



- [4, 1, 1, 2], [2].



问题 L和弦

输入文件：	标准输入
输出文件：	标准输出
时间限制	3 秒 内存限制
	1024 兆字节

圆上标有 $2n$ 个点，按顺时针方向等间隔排列。这些点以**随机方式**（更多内容请参见"输入"部分）连接成 n 对，形成 n 个和弦。你的任务是找出没有两个点相交的最大可能的弦集，并输出这个弦集的大小。

输入

标准输入的第一行包含一个整数 n ($1 \leq n \leq 100\,000$)，表示和弦的数量。

在接下来的 n 行中，各有两个整数 a_i 和 b_i ($1 \leq a_i < b_i \leq 2n$)，表示弦连接 a_i -th 点和 b_i -th 点。 a_i 和 b_i 的所有 $2n$ 个值都是成对不同的。

注：出于评委的懒惰，为了增加任务的趣味性，本任务的所有测试（样本测试除外）都是随机生成的。为每个测试选择一个 n 值和随机数生成器的种子。然后随机产生一个从 1 到 $2n$ 的数字排列，并将其分为 n 对。然后，在每对数字中，可以交换数字，以确保 $a_i < b_i$ 这一条件。

示例测试是手动创建的，但要使解决方案被接受，还必须正确求解。

输出

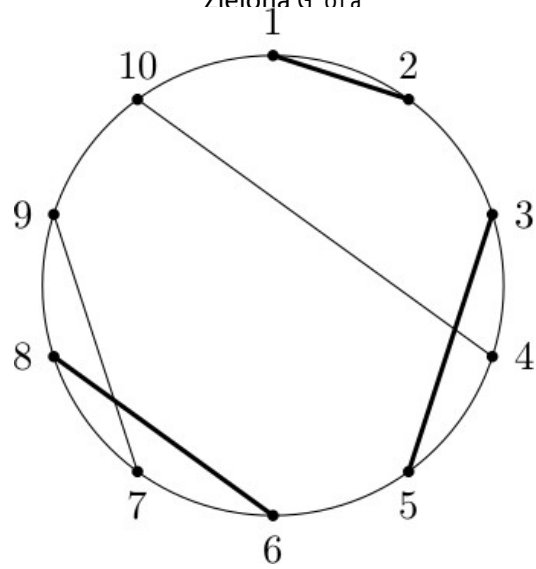
输出结果应包含一个整数，表示不相交和弦最大集合的大小。

示例

标准输入	标准输出
5 1 2 4 10 7 9 3 5 6 8	3

备注

样本测试中的和弦如下：



可能最大的一组正确和弦包含连接点 1 和 2、3 和 5 以及 6 和 8 的和弦。



问题 M. 排列平衡

输入文件：标准输入
输出文件：标准输出
时间限制 8 秒 内存限制
1024 兆字节

$|p_i - i|$ 。您的任务是找到

对于从 1 到 n 的数字排列 p ，其平衡定义为 $\sum_{i=1}^n$
余数等于 b 的长度为 n 的第 k 个最小词典排列。

输入

如问题陈述所述，标准输入包含一行三个整数 n 、 b 和 k ($1 \leq n \leq 30$; $0 \leq b$; $1 \leq k$)。
可以保证至少有 k 种长度为 n 的排列组合的余额等于 b 。
请注意，参数 k 的值可能不适合 64 位数据类型。在 C++ 中，我们建议使用 int128 类型来存储它。

输出

输出结果应在一行中包含 n 个整数，即所求的排列组合。

实例

标准输入	标准输出
6 6 6	1 2 6 3 4 5
30 300 3030303030303030303030303030	1 2 3 4 9 23 20 28 24 16 21 17 27 29 8 26 25 30 19 18 22 12 7 13 6 10 5 15 14 11

备注

考虑第一次抽样检验。有 46 种长度为 6 的排列，余数等于 6。其中，按词典顺序排列最小的第六个排列是排列 [1, 2, 6, 3, 4, 5]。
第二个示例中的换行符是为适应 PDF 语句的 "输出" 部分而添加的。您不需要在实际输出中添加换行符。