

# 2026 年国际大学生程序设计竞赛全国邀请 赛（武汉）暨湖北省赛 正式赛

2026 年 5 月 17 日



## 题目列表

A	期望排序
B	序列操作
C	相信着你
D	质数博弈
E	棋盘
F	抽奖
G	我会/一直/记得你
H	矩形切割
I	奶龙大战暴暴龙 2
J	最厉害的卡牌
K	删除游戏
L	字符串匹配
M	彩叶和构造王国

本试题册共 13 题，22 页。

如果您的试题册缺少页面，请立即通知志愿者。

## Problem A. 期望排序

Input file: standard input  
Output file: standard output

给定一个长度为  $n$  的排列  $p$ 。

每次操作会等概率随机选择一个整数  $i$  ( $1 \leq i \leq n$ ), 并将排列  $p$  中的前缀区间  $[1, i]$  和后缀区间  $[i+1, n]$  分别进行升序排序。特别地, 当  $i = n$  时, 后缀区间为空, 即相当于对整个排列进行排序。

你要求出, 使得最终排列完全升序有序 (即满足  $p_j = j$ ) 所需要的期望操作次数。

答案对 998244353 取模。

### Input

本题包含多组测试数据。

第一行包含一个整数  $T$  ( $1 \leq T \leq 100$ ), 表示测试数据组数。

对于每组测试数据:

- 第一行包含一个整数  $n$  ( $1 \leq n \leq 500$ ), 表示排列的长度。
- 第二行包含  $n$  个整数  $p_1, p_2, \dots, p_n$  ( $1 \leq p_j \leq n$ ), 表示给定的排列  $p$ 。保证给出的序列是一个  $1 \sim n$  的排列。

保证所有测试数据中  $n$  的总和不超过 2000。

### Output

对于每组测试数据输出一行, 包含一个整数, 表示期望操作次数对 998244353 取模后的结果。

### Example

standard input	standard output
3	748683267
3	2
3 1 2	670695427
2	
2 1	
5	
3 2 5 1 4	

## Problem B. 序列操作

Input file: standard input  
Output file: standard output

给定一个长度为  $n$  的非负整数序列  $a_1, a_2, \dots, a_n$ , 以及两个长度为  $m$  的序列  $c_1, c_2, \dots, c_m$  和  $x_1, x_2, \dots, x_m$ 。保证  $x_i$  均为正整数。

你需要从 1 到  $m$  依次顺序执行  $m$  次操作。在第  $i$  次操作中, 你可以根据  $c_i$  的值对当前的序列  $a$  进行修改:

- 若  $c_i = 1$ , 那么你只能对所有的  $a_j$  执行  $a_j \leftarrow \text{mex}(a_j, x_i)$ 。
- 若  $c_i = 2$ , 那么你只能对所有的  $a_j$  执行  $a_j \leftarrow \text{gcd}(a_j, x_i)$ 。
- 若  $c_i = 0$ , 你可以自由选择执行上述两种操作中的任意一种。

在所有  $m$  次操作执行完毕后, 判断最后能不能让序列  $a$  中的所有数都相等。

关于运算  $\text{mex}$  和  $\text{gcd}$  的额外说明如下:

- 定义二元操作  $\text{mex}(u, v)$  表示不等于  $u$  且不等于  $v$  的最小非负整数。例如:  $\text{mex}(0, 1) = 2$ ,  $\text{mex}(2, 2) = 0$ 。
- 对于最大公约数  $\text{gcd}$ , 特别定义  $\text{gcd}(0, x) = x$ 。

### Input

输入包含多组测试数据。第一行包含一个整数  $T$  ( $1 \leq T \leq 10^4$ ), 表示测试数据组数。

对于每组测试数据:

- 第一行包含两个整数  $n, m$  ( $1 \leq n, m \leq 3 \times 10^5$ ), 分别表示序列  $a$  的长度和操作的次数。
- 第二行包含  $n$  个非负整数  $a_1, a_2, \dots, a_n$  ( $0 \leq a_j \leq 10^9$ ), 表示初始序列。
- 接下来  $m$  行, 第  $i$  行包含两个整数  $c_i, x_i$  ( $0 \leq c_i \leq 2, 1 \leq x_i \leq 10^9$ ), 表示第  $i$  次操作的类型参数和数值参数。

保证所有测试数据中  $\sum n \leq 3 \times 10^5$  且  $\sum m \leq 3 \times 10^5$ 。

### Output

对于每组测试数据输出一行。如果在所有操作结束后能让所有数都相等, 则输出 “Yes”; 否则输出 “No”。

## Example

standard input	standard output
7	Yes
8 6	No
3 1 4 1 5 9 2 6	Yes
0 1	No
0 1	No
0 4	No
0 5	Yes
0 1	
0 4	
3 1	
0 1 2	
1 1	
3 1	
0 1 2	
2 1	
4 2	
3 3 5 0	
1 1	
2 3	
2 2	
3 0	
2 4	
2 2	
4 2	
3 0 2 0	
1 1	
1 3	
3 4	
1 4 0	
1 1	
1 1	
0 4	
2 2	

## Note

对于第一组数据：在第一次操作时，选择执行 gcd 操作，序列将变为  $[1, 1, 1, 1, 1, 1, 1]$ 。在后续的 5 次操作中，你无论选择什么操作，序列元素始终会保持一致，因此输出 “Yes”。

对于第二组数据：唯一的一次操作是  $c_1 = 1$ ,  $x_1 = 1$ ，你只能选择 mex 操作。

- $a_1 \leftarrow \text{mex}(0, 1) = 2$
- $a_2 \leftarrow \text{mex}(1, 1) = 0$
- $a_3 \leftarrow \text{mex}(2, 1) = 0$

最终序列变为  $[2, 0, 0]$ ，无法让所有数字相等，输出 “No”。

## Problem C. 相信着你

Input file:        standard input  
Output file:       standard output

《杀戮尖塔 2》是一款风靡全球的游戏。游戏中，你需要扮演攀爬高塔的角色，并用卡牌击败沿途的怪物，直到获得胜利。

现在你不得不面对一个强大的敌人，很不幸的是，你剩下的生命值无法抵挡任何一次未被格挡的攻击。幸运的是，你知道了未来  $n$  回合内你能抽到的牌和怪物的行动，你希望快速判断能否在  $n$  回合内击败怪物，且自己不会死亡。

游戏的卡牌分为两种：

- 打击 (Strike): 打出会造成  $S$  点伤害。
- 防御 (Defend): 打出会为自己提供  $D$  点格挡。

击败敌人的条件是你造成的总伤害大于等于敌人的生命值  $HP$ 。

在第  $i$  个回合，玩家的行动定义为：

- 清空当前手上的牌以及上一回合剩余的格挡（如果有）。
- 获得  $a_i$  张打击， $5 - a_i$  张防御。
- 选择不超过 3 张卡牌打出，每张卡牌只能被打出一次。
- 结束行动。

在第  $i$  个回合，怪物的行动定义为：

- 对玩家造成  $k_i$  点伤害。一旦  $k_i$  大于玩家当前的格挡值，你就会死亡，游戏立即结束。
- 结束行动。

在每个回合，双方的行动顺序是：

1. 玩家行动。
2. 若怪物死亡，则玩家胜利，游戏结束。
3. 怪物行动。
4. 若玩家死亡，则游戏结束。

你需要判断在  $n$  个回合内你能否击败怪物。若可以，输出 “Yes”，并输出最快在哪个回合能击败怪物；若不能在  $n$  回合内击败怪物或者中途玩家死亡，则输出 “No”。

### Input

第一行包含一个整数  $n$  ( $1 \leq n \leq 100$ )，表示预测的游戏回合数。

第二行包含三个正整数  $S, D, HP$  ( $1 \leq S, D \leq 10, 1 \leq HP \leq 100$ )，分别表示打击造成的伤害、防御提供的格挡和怪物的血量。

接下来  $n$  行，每行包含两个非负整数  $a_i, k_i$  ( $0 \leq a_i \leq 5, 0 \leq k_i \leq 50$ )，描述第  $i$  轮的抽牌情况（抽到  $a_i$  张打击）和怪物的攻击伤害。

## Output

若能在  $n$  回合内击败怪物，输出一行 “Yes”，并在下一行输出一个整数，表示最快在哪个回合能击败怪物。

若不能在  $n$  回合内击败怪物或者中途玩家死亡，则输出一行 “No”。

## Examples

standard input	standard output
4 5 5 25 4 0 1 10 2 12 2 12	Yes 3
5 1 5 4 1 4 2 0 5 3 0 4 0 1	Yes 3
4 1 5 5 0 1 3 3 0 5 2 1	No
5 1 4 4 5 4 1 5 3 1 4 4 5 4	No
4 4 5 4 1 2 1 0 4 5 1 1	Yes 1

## Note

对于第一组样例，最优的打牌策略如下：

- 第一回合：玩家抽到 4 张打击，1 张防御。怪物攻击力为 0。玩家打出 3 张打击，造成 15 点伤害。回合结束时怪物剩余 15 点血量。玩家受到 0 点伤害，存活。

- 第二回合：玩家抽到 1 张打击，4 张防御。怪物攻击力为 10。为了存活，玩家必须打出 2 张防御获得 10 点格挡，剩余 1 次出牌机会打出 1 张打击，造成 5 点伤害。回合结束时怪物剩余 10 点血量。玩家受到 10 点攻击，被格挡抵消，存活。
- 第三回合：玩家抽到 2 张打击，3 张防御。由于只要怪物死亡就不会进行攻击阶段，玩家可以直接打出 2 张打击，造成 10 点伤害。此时累计总伤害达到 30，怪物死亡，游戏在第 3 回合结束。

## Problem D. 质数博弈

Input file:        standard input  
Output file:        standard output

彩叶和辉夜最近迷上了一个基于质数的博弈游戏。

起初，她们共有一个正整数  $n$ 。两人将轮流进行操作，彩叶与辉夜各完成一次操作视为一个回合。在每一个回合中：

1. 彩叶先手：她需要选择一个非负整数  $x$  ( $0 \leq x < n$ )，并将当前的  $n$  替换为  $n + x$ 。
2. 辉夜后手：她需要选择一个能整除当前  $n$  的质数  $p$  (即  $p \mid n$ )，并将当前的  $n$  替换为  $\frac{n}{p}$ 。

以上操作会不断重复，直到某个回合结束后，若  $n = 1$ ，则游戏立即结束。

在游戏过程中，彩叶希望最大化游戏进行的总回合数，而辉夜希望最小化总回合数。假设双方都足够聪明并采取最优策略，请你求出游戏最终会进行多少个回合。

### Input

第一行包含一个整数  $n$  ( $1 \leq n \leq 10^{18}$ )，表示游戏初始的数字。

### Output

输出一行包含一个整数，表示在双方都采取最优策略的情况下，游戏最终进行的总回合数。

### Examples

standard input	standard output
1	0
9	4
16	5

## Problem E. 棋盘

Input file:        standard input  
Output file:       standard output

有一个  $N \times M$  的棋盘。

你需要在这个棋盘上放置  $K$  个互不相同的“车”（即不同的棋子）。放置方案必须满足以下所有条件：

1. 每个格子至多放置一个“车”。
2. 棋盘的每一行都至少有一个“车”。
3. 棋盘的每一列都至少有一个“车”。

请你求出满足条件的放置方案总数。由于答案可能很大，你只需要输出方案数对 998244353 取模后的结果。

### Input

输入仅包含一行。

这一行依次包含三个整数  $N, M, K$  ( $1 \leq N, M, K \leq 10^5$ )，分别表示棋盘的行数、列数以及需要放置的“车”的数量。

### Output

输出一行包含一个整数，表示合法的放置方案总数对 998244353 取模后的结果。

### Examples

standard input	standard output
2 2 3	24
5 3 4	0
353 442 899	623104742

## Problem F. 抽奖

Input file:        standard input  
Output file:       standard output

月读空间里正在举办一场抽奖活动。

活动奖池中共有  $n$  个商品，它们的价值分别为  $A_1, A_2, \dots, A_n$ 。

辉夜可以进行若干轮抽奖，但至少要进行一轮。在每一轮抽奖中，过程如下：

1. 辉夜需要支付当前的抽奖代价  $c$ ，随后在  $[1, n]$  中等概率随机抽取一个整数  $x$ ，并得知所抽中的商品  $x$ 。
2. 得知结果后，辉夜面临两个选择：
  - 留下第  $x$  个商品并结束整个抽奖活动，此时她将获得价值为  $A_x$  的商品。
  - 放弃当前抽中的商品，进入下一轮抽奖。但是，抽奖的代价会随之提升，下一轮的抽奖代价将会增加  $k$ ，即更新  $c \leftarrow c + k$ 。

辉夜将这场抽奖活动的“利润”定义为：最终获得商品的值  $A_x$  减去她在所有轮次中支付的抽奖代价之和。

假设辉夜足够聪明，并且总是采取最优策略来最大化她的期望利润。请你求出，在最优策略下，辉夜能获得的期望利润是多少？

### Input

第一行包含三个整数  $n, c, k$  ( $1 \leq n \leq 4 \times 10^5$ ,  $1 \leq c \leq 4 \times 10^5$ ,  $0 \leq k \leq 4 \times 10^5$ )，分别表示奖池中的商品总数、第一轮抽奖的初始代价，以及每重抽一轮代价的增加量。

第二行包含  $n$  个整数  $A_1, A_2, \dots, A_n$  ( $1 \leq A_i \leq 4 \times 10^5$ )，依次表示每个商品的值。

### Output

输出一行包含一个实数，表示辉夜在最优策略下的期望利润。你的答案被认为是正确的，当且仅当你的答案和标准答案的绝对或相对误差不超过  $10^{-6}$ 。

### Examples

standard input	standard output
3 2 0 4 1 6	2.000000000
9 8 2 53 71 5 6 80 37 50 33 90	51.557786840

## Problem G. 我会/一直/记得你

Input file:        standard input  
Output file:       standard output

给定一个  $n$  个点  $m$  条边的有向无环图  $G$ ，以及一个长度为  $n$  的颜色序列  $a_1, a_2, \dots, a_n$ ，其中  $a_i$  表示点  $i$  的初始颜色。

记  $S_i$  表示在图  $G$  上从点  $i$  出发能够到达的所有点构成的集合（包括点  $i$  自身）。

现在需要依次处理  $q$  次操作，操作分为以下两种：

- 给定  $x$  和  $y$ ，将点  $x$  的颜色  $a_x$  修改为  $y$ 。
- 给定  $x$ ，查询从点  $x$  出发能够到达的所有点中，一共有多少种不同的颜色。即求集合  $\{a_j \mid j \in S_x\}$  的大小。

### Input

第一行包含两个整数  $n, m$  ( $1 \leq n \leq 1.5 \times 10^5$ ,  $0 \leq m \leq 3 \times 10^5$ )，分别表示图的节点数和边数。

第二行包含  $n$  个整数  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq n$ )，表示每个节点的初始颜色。

接下来  $m$  行，每行包含两个整数  $u, v$  ( $1 \leq u, v \leq n$ )，表示图  $G$  中存在一条从  $u$  指向  $v$  的有向边。保证图  $G$  是一个有向无环图，且不存在重边。

接下来一行包含一个整数  $q$  ( $1 \leq q \leq 1.5 \times 10^5$ )，表示操作的总次数。

接下来  $q$  行，每行描述一次操作，格式为以下两种之一：

- $1 \ x \ y$ : 表示一次修改操作，将  $a_x$  的颜色修改为  $y$  ( $1 \leq x, y \leq n$ )。
- $2 \ x$ : 表示一次查询操作，查询从点  $x$  ( $1 \leq x \leq n$ ) 出发能到达的颜色种类数。

### Output

对于每个第二种操作，输出一行包含一个整数，表示对应查询的答案。

## Example

standard input	standard output
9 14	6
8 4 2 3 5 2 2 4 7	5
3 5	2
8 9	5
4 5	1
5 8	
6 8	
3 8	
1 4	
2 9	
2 4	
2 8	
1 3	
7 9	
1 8	
5 7	
10	
1 6 4	
1 3 3	
2 1	
2 3	
1 4 3	
1 8 5	
1 8 7	
2 7	
2 2	
2 9	

## Problem H. 矩形切割

Input file:        standard input  
Output file:       standard output

八千代在平面直角坐标系的第一象限内有一个矩形。该矩形的左下角位于坐标原点  $(0,0)$ ，右上角位于  $(n,m)$ 。换言之，矩形的两边分别与  $x$  轴、 $y$  轴重合，其在  $x$  轴方向的长度为  $n$ ，在  $y$  轴方向的长度为  $m$ 。八千代决定把这个矩形切  $q$  刀。具体地，她有以下两种切割方式：

- 横向切割：给定一个正整数  $k$ ，沿着直线  $y = k$  进行切割。
- 纵向切割：给定一个正整数  $k$ ，沿着直线  $x = k$  进行切割。

每次切割都会贯穿整个区域，将现有的矩形进一步切割成更多的小矩形。

八千代想知道，在每一次切割完成之后，当前所有被切分出来的小矩形中，面积的最大值是多少？

### Input

第一行包含三个整数  $n, m, q$  ( $1 \leq n, m \leq 10^9$ ,  $1 \leq q \leq 5 \times 10^5$ )，分别表示初始矩形的水平长度、垂直长度以及切割的次数。

接下来  $q$  行，每行包含两个整数  $op, k$ ，描述一次切割操作：

- 若  $op = 1$ ，表示进行一次纵向切割，给定整数  $k$  ( $1 \leq k < n$ )，沿着直线  $x = k$  切割。
- 若  $op = 2$ ，表示进行一次横向切割，给定整数  $k$  ( $1 \leq k < m$ )，沿着直线  $y = k$  切割。

保证八千代绝对不会在同一个位置切割两次（即所有的切割直线均互不相同）。

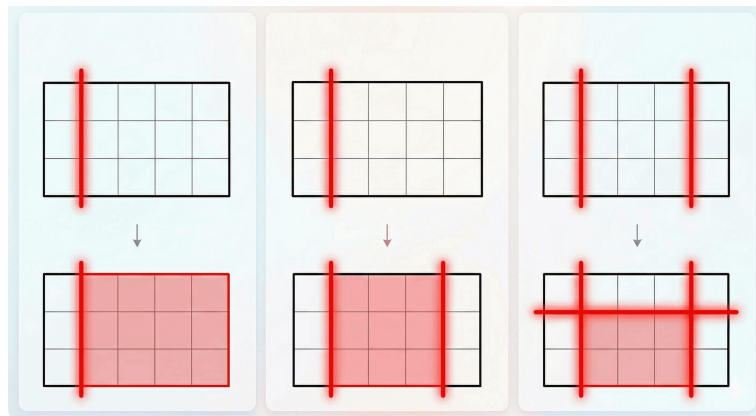
### Output

输出共  $q$  行，每行包含一个整数，第  $i$  行的整数表示在第  $i$  次切割之后，所有小矩形中面积的最大值。

## Example

standard input	standard output
5 3 3	12
1 1	9
1 4	6
2 2	

## Note



. 1: 样例解释

## Problem I. 奶龙大战暴暴龙 2

Input file: standard input  
Output file: standard output

这是一道通信题。在本题中，你的程序将运行两次。两次运行之间，内存中存储的所有变量都将丢失，但第一次运行中获取的信息可能对第二次运行中正确解决问题非常重要。

本题中有“奶龙”和“暴暴龙”两个角色。

奶龙拥有一棵包含  $n$  个节点的树的完整结构信息，而暴暴龙只知道树的节点数  $n$ 。由于暴暴龙被邪恶的小豹子关起来了，所以奶龙只能通过一种特殊的单向通信方式，来帮助暴暴龙还原出一棵与原树同构的树。

通信的规则如下：

- 初始时，树上的所有节点都是白色的。
- 奶龙可以进行  $m \leq n - 3$  次操作（ $m$  的值由奶龙自行决定）。
- 在第  $i$  次操作中，奶龙需要选择树上的一个节点  $x_i$ ，并反转该节点的颜色（白变黑，黑变白）。
- 每次反转后，定义  $a_i$  为当前树中两端点颜色不同的边的数量。
- 经过  $m$  次操作后，奶龙得到了一个长度为  $m$  的序列  $a_1, a_2, \dots, a_m$ 。

奶龙无法直接将节点编号发给暴暴龙，她只能将操作总数  $m$  以及序列  $a$  发送给暴暴龙。暴暴龙在收到  $n$ 、 $m$  以及序列  $a$  后，需要构造并输出一棵与奶龙的树同构的树。

### Communication Protocol

每个测试点中，选手程序将被运行两次。在下发文件中，提供了一份测试工具供选手本地调试使用。

#### First Run

在第一次运行中，你将扮演“奶龙”角色。

#### Input

输入的第一行包含一个整数 1，其作用是让你的程序能够识别这是第一次运行。

第二行包含一个整数  $n$  ( $4 \leq n \leq 3 \times 10^5$ )，表示树的节点数。

接下来  $n - 1$  行，每行包含两个整数  $u, v$  ( $1 \leq u, v \leq n$ )，表示树上的一条边。

#### Output

输出的第一行应该包含一个整数  $m$  ( $0 \leq m \leq n - 3$ )，表示操作次数。

第二行包含  $m$  个整数  $x_1, x_2, \dots, x_m$  ( $1 \leq x_i \leq n$ )，依次表示奶龙每次操作反转的节点编号。

#### Second Run

在第一次运行中，你将扮演“暴暴龙”角色。

#### Input

输入的第一行包含一个整数 2，其作用是让你的程序能够识别这是第二次运行。

第二行包含两个整数  $n, m$  ( $4 \leq n \leq 3 \times 10^5$ ,  $0 \leq m \leq n - 3$ )，分别表示节点数和奶龙的操作次数。

若  $m > 0$ ，第三行包含  $m$  个整数  $a_1, a_2, \dots, a_m$  ( $0 \leq a_i \leq n$ )，表示评测机根据奶龙的操作生成的序列；若  $m = 0$ ，则没有第三行。

## Output

输出  $n - 1$  行，每行两个整数  $u, v$ ，表示你还原出的树的一条边 ( $1 \leq u, v \leq n$ )。你需要保证输出的树和第一次运行中输入的树同构。

## Examples

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

## Note

两个样例演示了同一测试点中的两次运行。

奶龙翻转了节点 2，此时两端点颜色不同的边的数目为 2，故  $a_1 = 2$ 。

暴暴龙获得了序列  $a = [2]$ ，由于暴暴龙和奶龙心有灵犀，所以她一下子就猜到了这棵树的结构。虽然暴暴龙得到的树和奶龙手上的树关于点的编号不同，但他们是同构的，因此仍然算作正确。

## Problem J. 最厉害的卡牌

Input file:       standard input  
Output file:       standard output

蛇咬是游戏《杀戮尖塔 2》中最强力的卡牌，Yuki 十分喜欢用这张卡打败敌人。

游戏一共会进行  $10^{1000}$  个回合。在游戏开始时，敌人的中毒层数为 0。

Yuki 共拥有  $n$  张蛇咬，第  $i$  张蛇咬的效果是：在打出时使敌人的中毒层数增加  $v_i$ 。Yuki 会在第  $t_i$  个回合开始时打出第  $i$  张蛇咬。

在每个回合结束时，设敌人的中毒层数为  $x$ ：

- 若  $x = 0$ ，则无事发生；
- 若  $x \neq 0$ ，则敌人受到  $x$  点伤害，且中毒层数减少 1，即  $x \leftarrow x - 1$ 。

为了使敌人受到尽可能多的伤害，在游戏开始前，Yuki 可以进行  $k$  次附魔。每次附魔，Yuki 需要选择一个不大于  $n$  的正整数  $i$ ，将  $v_i$  的值增加 1。同一个正整数  $i$  可以被选择多次。

你需要求出，在 Yuki 进行  $k$  次附魔后，敌人最多会受到多少点伤害。

### Input

本题包含多组测试数据。

第一行包含一个整数  $t$  ( $1 \leq t \leq 10^4$ )，表示测试数据组数。

对于每组测试数据：

- 第一行包含两个整数  $n, k$  ( $1 \leq n \leq 3 \cdot 10^5$ ,  $0 \leq k \leq 10^9$ )。
- 第二行包含  $n$  个整数  $v_1, v_2, \dots, v_n$  ( $1 \leq v_i \leq 10^9$ ,  $1 \leq \sum v_i \leq 10^9$ )。
- 第三行包含  $n$  个整数  $t_1, t_2, \dots, t_n$  ( $1 \leq t_i \leq 10^9$ )。

保证所有测试数据中  $n$  的总和不超过  $3 \cdot 10^5$ 。

### Output

对于每组测试数据，输出一行，包含一个整数，表示在 Yuki 进行  $k$  次附魔后，敌人最多会受到的伤害点数。

## Example

standard input	standard output
5	13
2 1	37
2 3	63
1 4	3335
2 2	126
3 4	
1 3	
4 1	
3 3 1 4	
4 4 1 5	
3 0	
16 27 62	
9 88 11	
3 5	
9 3 1	
8 3 3	

## Problem K. 删除游戏

Input file: standard input  
Output file: standard output

八千代有一个长度为  $n$  的整数序列  $S$ ，序列的每个位置（下标从 1 开始）都有一个权值  $a_i$ 。她可以将这个序列连同权值一起复制并首尾拼接若干次。具体来说，若她选择拼接使得总份数为  $m$  ( $m \geq 1$ )，则会得到一个长度为  $m \times n$  的新序列  $S'$  和对应的新权值序列  $a'$ 。对于任意的  $0 \leq c < m$  和  $1 \leq i \leq n$ ，满足  $S'_{cn+i} = S_i$  且  $a'_{cn+i} = a_i$ 。

八千代现在想对得到的新序列  $S'$  进行若干次（可以为 0 次）消除操作。每次操作的规则如下：

- 选定两个下标  $i, j$  满足  $1 \leq i < j \leq |S'|$ ，且  $S'_i = S'_j$ 。
- 将  $S'$  和  $a'$  中的第  $(i+1)$  个到第  $j$  个元素删去。
- 操作完成后，序列剩余的元素会按照原顺序重新紧凑排列，序列的总长度相应减少。

八千代想知道，在经过恰当份数的拼接和任意次消除操作后，整个剩余序列的权值和最小是多少？以及在保证权值和最小的前提下，最少需要多少份原序列拼接（即求最小的  $m$ ）？

### Input

输入包含三行。

第一行包含一个整数  $n$  ( $1 \leq n \leq 3 \times 10^5$ )，表示初始序列  $S$  的长度。

第二行包含  $n$  个整数  $S_1, S_2, \dots, S_n$  ( $1 \leq S_i \leq 3 \times 10^5$ )，表示初始序列  $S$  的元素。

第三行包含  $n$  个整数  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 3 \times 10^5$ )，表示每个位置对应的权值。

### Output

输出一行包含两个整数，用空格隔开：

- 第一个整数表示经过操作后，剩余序列的最小权值和。
- 第二个整数表示为了达到该最小权值和，最少需要的原序列总份数。

### Example

standard input	standard output
6	2 1
1 1 4 5 1 4	
1 9 1 9 8 10	

### Note

在本样例中，最优策略是仅使用 1 份原序列（即  $m = 1$ ，不进行额外复制）。

初始序列  $S'$  为  $[1, 1, 4, 5, 1, 4]$ ，对应的权值  $a'$  为  $[1, 9, 1, 9, 8, 10]$ 。

八千代可以进行以下两次操作：

1. 选择  $i = 1, j = 2$ （此时  $S'_1 = S'_2 = 1$ ），删去第  $i+1 \sim j$  个元素（即删去第 2 个元素）。操作后序列  $S'$  变为  $[1, 4, 5, 1, 4]$ ，权值  $a'$  变为  $[1, 1, 9, 8, 10]$ 。

2. 在新序列中选择  $i = 2, j = 5$  (此时  $S'_2 = S'_5 = 4$ ), 删去第  $i + 1 \sim j$  个元素 (即删去第 3, 4, 5 个元素)。操作后序列  $S'$  剩余  $[1, 4]$ , 权值  $a'$  剩余  $[1, 1]$ 。

此时无法继续消除, 剩余序列的权值和为  $1 + 1 = 2$ 。可以证明在任何拼接份数和操作下, 权值和不可能小于 2。

## Problem L. 字符串匹配

Input file: standard input  
Output file: standard output

给定  $n$  个由小写字母组成的字符串  $s_1, s_2, \dots, s_n$ 。保证这些字符串已按照长度非严格递增的顺序排列（即  $|s_1| \leq |s_2| \leq \dots \leq |s_n|$ ）。

对于任意两个不同的字符串  $s_i$  和  $s_j$ （满足  $1 \leq i < j \leq n$ ），定义  $t_{i,j} = s_i + s_j$ ，表示将字符串  $s_i$  和  $s_j$  按顺序拼接得到的新字符串（其中  $s_i$  在前， $s_j$  在后）。

我们定义一个函数  $f(t)$ ，其中  $t$  是一个长度为  $m$  的字符串。 $f(t)$  的值为满足以下条件的正整数  $x$ （ $0 < x \leq m$ ）的个数：

- 字符串  $t$  的前  $x$  个字符与后  $x$  个字符完全相同。即  $t[1 : x] = t[m - x + 1 : m]$ 。

你的任务是计算出所有合法对  $(i, j)$  的  $f(t_{i,j})$  的总和，即求：

$$\sum_{1 \leq i < j \leq n} f(t_{i,j}).$$

### Input

输入包含多行。

第一行包含一个整数  $n$ （ $1 \leq n \leq 10^6$ ），表示给定的字符串总数。

接下来  $n$  行，第  $i$  行包含一个由小写字母组成的字符串  $s_i$ （ $1 \leq |s_i| \leq 10^6$ ）。

保证所有测试数据中，所有字符串的长度之和  $\sum |s_i| \leq 10^6$ 。

### Output

输出一行包含一个整数，表示所有  $f(t_{i,j})$  的总和。

### Examples

standard input	standard output
2 ak hbcpc	1
3 aa abaa ababaa	9

### Note

在第二组样例中，共有  $n = 3$  个字符串，我们可以组合出 3 种不同的  $t_{i,j}$ ：

- 当  $i = 1, j = 2$  时： $t_{1,2} = aaabaa$ 。满足前缀等于后缀的长度  $x$  有：1, 2, 6。
- 当  $i = 1, j = 3$  时： $t_{1,3} = aaababaa$ 。满足前缀等于后缀的长度  $x$  有：1, 2, 8。
- 当  $i = 2, j = 3$  时： $t_{2,3} = abaaababaa$ 。满足前缀等于后缀的长度  $x$  有：1, 4, 10。

最终答案为  $3 + 3 + 3 = 9$ 。

## Problem M. 彩叶和构造王国

Input file: standard input  
Output file: standard output

兜兜转转，彩叶来到了月读空间里的一个神秘地界 — 构造王国。

听闻你曾经来过此处，彩叶非常开心，并向你分享了她在构造王国中一段有趣的记忆。

构造王国中有一副神奇的纸牌，它有恰好  $n$  种不同点数（依次用  $1$  到  $n$  之间的整数表示），每种点数恰好有两张。经过均匀洗牌，这  $2n$  张纸牌构成了一幅牌堆。换句话说：输入数据保证，牌堆是由所有包含  $n$  种点数、每种恰好两张牌的合法牌堆中等概率随机生成的。

在这个空间中，彩叶被赋予了一个很强大的能力：生成栈。由于这项能力十分消耗体力，彩叶只能生成 260 个栈。

现在，彩叶需要按照从上到下的顺序依次取出初始牌堆中的牌。对于每张被取出的牌，彩叶可以将其放入任意一个栈中。

当所有的  $2n$  张牌都放置完毕后，彩叶的栈需要满足以下条件：可以进行若干次如下的“消除操作”，直到所有牌均被消除完毕：

- 每次操作选择两个不同的栈，若这两个栈当前的栈顶卡牌点数相同，则将这两张牌同时弹出（删除）。

很可惜，彩叶并没有解决这个问题，所以她把问题交给了聪明又有经验的你，希望你来帮她解决这个问题。

为了方便你解答这个问题，你不需要给出具体的消除过程，只需要输出按原牌堆顺序每张牌分别被放入了哪个栈中。彩叶会帮助你判断你的构造是否正确。

### Input

第一行包含一个正整数  $n$  ( $1 \leq n \leq 5 \times 10^5$ )，表示卡牌的种类数。

第二行包含  $2n$  个正整数  $a_1, a_2, \dots, a_{2n}$  ( $1 \leq a_i \leq n$ )，依次表示初始牌堆从上到下每张牌的点数。保证  $\{1, 2, \dots, n\}$  中的每种数字恰好出现两次。

保证数据恰好有 50 组（包括样例），每组数据中的数组  $a$  都是从所有的长度为  $2n$ ，且  $\{1, 2, \dots, n\}$  中每种数字各出现两次的序列中等概率选择。

### Output

输出一行，包含  $2n$  个正整数  $c_1, c_2, \dots, c_{2n}$ ，其中  $1 \leq c_i \leq 260$ ，表示初始牌堆中第  $i$  张取出的牌被放入了第  $c_i$  个栈中。

如果有多种满足条件的方案，输出任意一种即可。

### Example

standard input	standard output
3	1 1 1 2 2 3
1 2 3 1 3 2	

### Note

在处理完你的指令后，后 257 个栈都是空的，而前 3 个栈分别形如（元素按照从栈底到栈顶的顺序列举）：

1.  $[1, 2, 3]$ ;

2.  $[1, 2]$ ;

3.  $[3]$ 。

不难证明存在一种办法消除所有的数。