

河南省第十五届 ICPC 大学生程序设计竞赛
The 2024 ICPC China Henan Provincial Programming
Contest

现场赛
Onsite Round



题目列表
Problem List

A	分糖果
B	表达式求导
C	置换操作
D	游戏扑克牌
E	公切线
F	Factorial
G	三元组
H	二十四点
I	字符串大师
J	$kx+b$ 数列
K	或的最大值
L	士兵列队
M	集合游戏

请勿在比赛开始前翻阅试题!
Do not open before the contest has started.

2024 年 6 月 3 日

Problem A. 分糖果

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 2048 megabytes

幼儿园准备了 n 包糖果，每包糖果里有 1、2 或 3 颗美味的糖果。现在需要将这些糖果平分给两个表现优异的小朋友以作奖励，为了公平公正，需要确保两位小朋友分得的糖果颗数相等，且每一包糖果都需要分给正好一位小朋友。幼儿园园长委托你帮忙判断下：是否存在可行的分配方法？

Input

输入共两行。第一行一个整数 $n(1 \leq n \leq 2 \times 10^5)$ ，表示糖果的包数。第二行 n 个整数，第 i 个整数 $a_i(1 \leq a_i \leq 3)$ 表示第 i 包中糖果的数量。

Output

如果存在可行的分配方案，则输出”YES”；否则输出”NO”。

Example

standard input	standard output
5 1 3 3 2 1	YES
4 3 1 3 3	NO

Problem B. 表达式求导

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 2048 megabytes

给定一个合法的函数表达式 $f(x)$ ，请你求出 $y = f(x)$ 在 a 处的导数值，输入保证此处导数值一定存在，答案四舍五入保留两位小数。

该函数表达式被称为合法，即满足以下要求：

- 1. x 一定合法。
- 2. 如果表达式 A 合法，表达式 B 合法，那么 $A + B$ ， $A - B$ 一定合法。
- 3. 如果表达式 A 合法，那么 $\ln(A)$ 也合法。

Input

本题有多组数据。第一行输入整数 $T(1 \leq T \leq 30)$ ，表示数据组数。接下来每组数据包含两行。其中第一行输入一个字符串 S ，表示 $f(x)$ 的函数表达式， $1 \leq |S| \leq 20$ 。其中第二行输入一个至多两位小数的浮点数 $a(0 \leq a \leq 20)$ ，含义如上所示。

Output

输出一个浮点数，表示 $y = f(x)$ 在 a 处的导数值，答案四舍五入保留两位小数。

Example

standard input	standard output
1 ln(x)+x 0.5	3.00

Problem C. 置换操作

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 2048 megabytes

给定一个长度为 n 的 01 字符串，你可以至多执行 2 次修改操作，每次可以将任意一个 0 替换成 1。
请问在所有可能的修改方案中，最多有多少个全 1 子串？
其中子串的定义为原串中任意个连续字符组成的子序列。

Input

输入共两行。
第一行输入一个整数 $n(1 \leq n \leq 10^5)$ ，表示 01 字符串的长度。
第二行输入一个长度为 n 且仅由 0 和 1 构成的字符串，含义如上所示。

Output

输出一个整数，表示修改后全 1 子串数量的最大值。

Example

standard input	standard output
9 110001110	18

Explanation

对于样例，一种可能的修改方案为 110011111，此时全 1 子串数量为 18 个，为最大值。

Problem D. 游戏扑克牌

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 2048 megabytes

Alice 和 Bob 做游戏, Alice 将 n 张扑克牌背面朝上依次排开。每一张牌的正面花色要么是黑色, 要么是红色, 而背面则全部一样。其中有 $0, 1, \dots, n$ 张黑色牌的概率相等, 均为 $\frac{1}{n+1}$ 。

现在 Alice 将前 $k_b + k_r$ 张牌翻转, 并得知其中有 k_b 张牌为黑色, k_r 张牌为红色。而 Bob 则需要根据以上信息, 来猜测黑色牌的数量区间 $[L, R]$ 。如果黑色牌的数量不在该区间内, 则 Bob 没有积分; 否则被视为猜中, 在这种情况下区间长度越小难度越高, 相应的得分也就越多。Bob 只有一次机会, 他想要兼顾得分和猜错的风险, 因此 Bob 采用这样一种策略进行游戏。他先确定一个自己能够接受的最差概率 p , 然后只考虑那些猜中概率大于等于 p 的区间, 并在所有满足条件的区间中选一个最短的 (即 $R - L$ 最小的)。如果存在多个最短的, 则选其中下标最小的一个 (即 L 最小的)。

但是 Bob 的概率与统计学得并不是很好, 因此他将这个任务交给了你。

Input

本题采用多组数据。

第一行输入一个整数 $T (1 \leq T \leq 100)$, 代表数据组数。

接下来 T 行, 每行输入四个非负整数 $n (1 \leq n \leq 50), k_b (0 \leq k_b \leq n), k_r (0 \leq k_r \leq n - k_b), p (0 \leq p \leq 100)$, 分别代表扑克牌的总数, 前 $k_b + k_r$ 张牌中黑色牌的数量, 前 $k_b + k_r$ 张牌中红色牌的数量, 以及以百分比形式给出的 Bob 所能接受的最差概率 (实际概率为 $\frac{p}{100}$)。

Output

对于每一组数据, 输出一行一个整数, 表示 Bob 所要求的区间端点 L 和 R , 需要满足 $0 \leq L \leq R \leq n$ 。

Example

standard input	standard output
2	1 1
50 1 49 100	1 15
50 1 10 95	

Problem E. 公切线

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

切线是计算几何中至关重要的一个定义。一条直线 L 是正方形 S 的切线，当且仅当以下两个条件成立：

- 1. 正方形 S 至少有一个点在直线 L 上。
- 2. 在直线 L 所划分的两个半平面（即两侧）中，有一个半平面不存在来自正方形 S 的点。

公切线则是多个图形共有的切线。在一般的几何题中，总会涉及到两个圆的公切线，但却对其它图形的公切线很少提及。本题给定了两个边长相同的正方形，保证两者交集的面积为 0。请问两个正方形共有多少个公切线？

Input

本题采用多组数据。
第一行输入一个整数 $T(1 \leq T \leq 40)$ ，代表数据组数。

接下来 T 行，每行输入八个整数， $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ 。所有坐标的范围为 $[-5 \times 10^3, 5 \times 10^3]$ 。其中点 (x_1, y_1) 和点 (x_2, y_2) 是第一个正方形中一条对角线的两端点，点 (x_3, y_3) 和点 (x_4, y_4) 是第二个正方形中一条对角线的两端点。

数据保证两个正方形的边长相等，且两者交集的面积为 0。

Output

对于每一组数据输出一行。如果两个正方形存在无数条公切线，输出 “Infinity”（不带引号）；否则输出公切线的数量。

Example

standard input	standard output
2	3
0 0 2 2 1 2 3 4	3
0 0 10 0 8 2 16 8	

Problem F. Factorial

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 2048 megabytes

给你一个正整数 M ，请你求出一个最小的正整数 N 满足: $N! \equiv 0(mod \quad M)$ 。

由于 M 实在是太大了，为了方便你读入，善良的出题人给你 K 个整数 $e_1, e_2, ..., e_K$ ， $M = \prod_{i=1}^K p_i^{e_i}$ 。其中 p_i 是第 i 小的素数，保证答案不超过 10^{18} 。

Input

第一行一个整数 $T, (1 \leq T \leq 10^4)$ ，表示数据组数。

对于每组数据，第一行一个整数 $K (1 \leq K \leq 100)$ ，表示使用前 K 个素数即可表示 M 。其后一行 K 个整数 $e_i (0 \leq e_i \leq 10^{18})$ ，表示 M 分解后第 i 个素数的次数。

Output

对于每组数据，输出一行一个整数，表示最小的满足 $N! \equiv 0(mod \quad M)$ 的正整数 N 。

Example

standard input	standard output
1 5 1 1 1 1 1	11

Explanation

$M = 2^1 \times 3^1 \times 5^1 \times 7^1 \times 11^1 = 2310, 11! = 2310 \times 17280$

Problem G. 三元组

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

对于一个长度为 n 的序列 A , 定义其价值为 (i, j, k) 的个数, 满足 $1 \leq i < j < k \leq n$ 且 $A_i + A_j = A_k$ 。

现在请你求出, 对于一个序列, 序列中都是正整数, 且序列的所有元素的和为 X 的情况下, 三元组最多能有多少个?

由于该问题太简单, 出题人希望你能多次回答这个问题。

Input

第一行一个正整数 $T(1 \leq T \leq 100000)$, 表示数据组数。

对于每组数据, 输入一行一个正整数 $X(0 \leq X \leq 5 \times 10^6)$ 。代表对于序列中元素的和。

Output

对于每组数据, 输出一行一个整数, 代表答案。

Example

standard input	standard output
9	0
1	0
2	0
3	1
4	3
5	6
6	10
7	15
8	21
9	

Problem H. 二十四点

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

你手里有无数个整数 k (保证 $1 \leq k \leq 9$)。还有无数个左右小括号 $(,)$ 和加乘号 $+, *$ 。现在希望你拼出来一个合法的表达式, 使得式子的值为 n 。

允许你连续多次使用 k , 比如 k 为 1 时, 可以 $1+1+1+1+1+1+1+1+1+1+1+1$ 来拼出 11, 也可以用 $(1+1)*(1+1)*(1+1)+1+1+1$ 来拼出 11, 但是使用最少的 k 的方案是使用 11 来拼出 11。

现在希望你找到所有能拼出来值为 n 的表达式中, 使用 k 最少的个数。如果不能拼出来, 请输出 -1 。

Input

第一行一个正整数 t 表示数据组数 ($1 \leq t \leq 10^3$)。
接下来 t 行, 每行两个正整数分别表示 k 和 n ($1 \leq k \leq 9, 1 \leq n \leq 5 * 10^3$),

Output

输出 t 行, 每行一个整数表示本次询问的答案: 如果能拼出来, 请输出最少 k 的个数; 如果不能拼出来, 请输出 -1 。

Example

standard input	standard output
6	1
1 1	2
1 2	3
1 3	4
1 4	5
1 5	14
1 5000	

对于第六个询问, 你可以使用 $111+1+(1+11*111)*(1+1+1+1)$ 来得到 5000, 需要 14 个 1。

Example

standard input	standard output
4	-1
3 2	1
3 3	-1
3 10	4
3 66	

对于第四个询问, 你可以使用 $33+33$ 来得到 66, 需要 4 个 3。

Problem I. 字符串大师

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

在成为字符串大师的路上，你遇到了一个棘手的问题。

你有一个字符串 s 和一个长为 2 的 t ，令 $f(s,t)$ 表示有多少个 s 的子序列等于 t 。

子序列的定义：子序列就是在原来序列中找出一部分组成的序列（子序列不一定连续）。

为了最大化这个数量，你可以多次修改 s 的任意位置的字符。

现在希望你输出 s 经过多次修改后 $f(s,t)$ 的最大值。同时也想让你输出为了达成最大值，最少需要多少次修改。

Input

输入三行。

第一行一个正整数 n 表示字符串 s 的长度， $1 \leq n \leq 10^6$ 。

第二行一个长为 n 的字符串 s ，保证 s 中只包含小写字母。

第三行一个长为 2 的字符串 t ，保证 t 中只包含小写字母。

Output

输出一行两个整数表示多次修改后 $f(s,t)$ 的最大值、为了达成最大值最少需要多少次修改。

Example

standard input	standard output
4 aabb ab	4 0

不需要修改即可获得最大值为 4

Example

standard input	standard output
3 abc aa	3 2

$f(s,t)$ 最大值为 3，需要两次修改将字符串 s 修改为 "aaa"。

Problem J. $kx + b$ 数列

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

现在有一个整数数列 A , $(-10^{18} \leq A_i \leq 10^{18})$, 它虽然简简单单, 但是却有着奇妙的性质: 有一组 k, b , 满足对于任意 $2 \leq i, A_i = A_{i-1} * k + b$ 。其中 k 不等于 0。

现在小 Q 只记得这个数列的前 n 项, 请你帮助他回忆 k 和 b 的值, 有如下规则:

- k 和 b 都是 $[-10^{18}, 10^{18}]$ 范围内的整数, 其中 k 不等于 0, 如果存在解的话按如下规则确定 k 和 b 的值, 并输出一行用空格隔开的两个整数, 代表 k, b 。
- 如果 k 和 b 有唯一解, 则直接输出这个唯一解。
- 如果有多解, 输出 b 的绝对值最小的解, 如果有两个 b 绝对值最小的解, 输出 b 大于 0 的那个。
- 如果确定 b 后 k 有多解, 那么输出绝对值最小的 k , 如果有两个 k 绝对值最小的解, 输出大于 0 的那个。
- 如果无解, 输出 -1。

Input

第一行输入一个正整数 t 表示数据组数 $1 \leq t \leq 10^5$ 。

接下来 t 组输入, 每组两行: 第一行一个正整数 n 表示数列长度, 第二行 n 个整数表示数列 A 里的每个数字。

数据保证 $\sum n \leq 10^6, 1 \leq n \leq 10^6, -10^{18} \leq A_i \leq 10^{18}$ 。

Output

对于每组询问, 输出一行两个整数分别表示 k 和 b 。或者输出 ‘-1’代表无解

Example

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

可以发现 $k = 1, b = 1$ 。

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

Problem K. 或的最大值

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

现在给你 n 个数 $a_i(2 \leq n \leq 10^5, 0 \leq a_i \leq 10^8 \text{ 且 } 0 \leq \sum a_i \leq 10^8)$ ，请你找两个互不相等且满足 $1 \leq i, j \leq n$ 的两个下标 i, j ，输出 $a_i \mid a_j$ 的最大值。

\mid 代表按位或运算，运算规则如下：

对于两个数的第 i 位，设为 x, y ，那么有如下运算规则：

x	y	$x \mid y$
0	1	1
1	0	1
0	0	0
1	1	1

例如： $13 \mid 23 = 31$

$13 =$	$0...001101$
$23 =$	$0...010111$
$13 \mid 23 =$	$0...011111$

Input

输入两行。
第一行一个正整数 n 表示数字数量。
接下来一行 n 个非负整数表示每个数字。

Output

输出一行一个整数表示答案。

Example

standard input	standard output
4 1 2 3 4	7

令 $i = 3, j = 4$ ，则有 $a_3 \mid a_4 = 3 \mid 4 = 7$ 。
也有别的方案，但是都小于等于 7。

Example

standard input	standard output
10 5 18 8 7 15 2 16 15 16 10	31

Problem L. 士兵列队

Input file: standard input
Output file: standard output
Time limit: 1 seconds
Memory limit: 2048 megabytes

有 n 个士兵正在准备出发，但是长官很喜欢二进制，他决定将所有士兵站成四列，每一列都需要是一个正二进制数，例如 1, 2, 4, 8, 16...

但是不一定能这样列队，例如 63。于是长官决定从别的连队调来若干个士兵。他想问问你他至少需要再加多少个士兵，使得能够站成四列，恰好满足每一份都是正二进制数。

Input

输入共 $t + 1$ 行。
第一行一个整数 $t(1 \leq t \leq 2 \times 10^4)$ ，表示询问的数量。
接下来 t 行，第 i 行一个正整数 $n(1 \leq n \leq 10^9)$ 表示本次询问的士兵数量。

Output

输出 t 行，每行一个整数表示这次询问的答案。

Example

standard input	standard output
3	0
10	1
63	4
236	

10 可以分成 1 1 4 4，不需要额外士兵。
63 不可以划分，但是 64 可以划分为 4 个 16。
236 ~ 239 都不可以划分，但是 240 可以划分成 128, 64, 32, 16。

Problem M. 集合游戏

Input file: standard input
Output file: standard output
Time limit: 2 second
Memory limit: 2048 megabytes

有一种游戏，每轮游戏给定两个集合 S, T ，其中 S 内元素个数为 n ， T 内元素个数为 m ，然后进行以下操作：

- 1. 等概率在两个集合中随机选择一个集合，并在选中的那个集合中等概率的删除其中的一个元素。
- 2. 如果进行上操作后有一个集合变为空集，游戏结束，否则返回操作 1。

现在有 Q 轮游戏，每轮游戏给定两个集合的大小 n, m ，问游戏结束时 S 为空集的概率。答案对 $10^9 + 7$ 取模。

Input

第一行输入一个整数 $Q(1 \leq Q \leq 10^5)$ ，表示游戏轮数。

接下来 Q 行，每行两个整数 $n, m(1 \leq n, m \leq 10^5)$ ，其中第 $i + 1$ 行的两个整数表示第 i 轮游戏中两个集合的大小。

Output

输出 Q 行，每行一个整数，其中第 i 行表示第 i 轮游戏中游戏结束时 S 为空集的概率对 $10^9 + 7$ 取模后的结果。

Example

standard input	standard output
2	500000004
1 1	750000006
1 2	
6	187500002
2 3	718750006
1 5	742034918
7 9	508259082
114 514	685612627
2025 1025	421871959
99824 43531	

Explanation

对于样例 1：对于第一轮游戏，在第一次操作中有 $\frac{1}{2}$ 的概率选择 S 集合，此时游戏结束时 S 集合为空，否则游戏结束时 T 集合为空，故答案为 $\frac{1}{2}$ ，对 $10^9 + 7$ 取模后答案为 500000004。

对于第二轮游戏：

- 有 $\frac{1}{2}$ 的概率进行一次删除操作后游戏结束且结束时 S 为空集，即第一次删除一个 S 中的元素。
- 有 $\frac{1}{4}$ 的概率进行两次删除操作后游戏结束且结束时 S 为空集，即第一次删除一个 T 中元素，第二次删除一个 S 中元素。

故答案为 $\frac{1}{2} + \frac{1}{4} = \frac{3}{4}$ ，对 $10^9 + 7$ 取模后答案为 750000006。