

上海大学 ACM 集训队练习赛

2018 年 1 月 28 日



Problems

Problem A. 抽彩球	1
Problem B. 阶乘幂	2
Problem C. 异或	3
Problem D. 统计难题	4
Problem E. 生成树计数	6
Problem F. 喵神的股票	7
Problem G. CSL 的字符诱惑	8
Problem H. TSL 序列	10
Problem I. OneDay 的小球	11
Problem J. 利普希茨序列	13

Problem A. 抽彩球

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

在一个盒子中有若干黑球和一个红球。CSL 和 OneDay 轮流从盒子中抽一个球(不放回)直到有人抽到了红球。抽到红球的人获胜。CSL 是个大方的人, 他让 OneDay 学长选择谁先手。现在请你来帮 OneDay 选择一下。

Input

有多组测试数据 (数据组数 ≤ 50)。对于每组数据, 有一个整数 k 表示黑球的数量。

Output

对于每组测试数据, 如果先手获胜概率较大则输出1, 后手获胜概率较大则输出2, 如果先后手获胜概率相同则输出0。

Limits

- $1 \leq k \leq 10^5$

Sample input and output

Sample Input	Sample Output
1	0
2	1

Problem B. 阶乘幂

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

TSL 喜欢大数 (如果你不喜欢, 你可以换题了), 他知道很多构造大数的方式, 例如:

- 幂运算: $28^{2018} = 28 \cdot 28 \cdot \dots \cdot 28$
- 阶乘: $2018! = 2018 \cdot 2017 \cdot \dots \cdot 2 \cdot 1$

今天, TSL 要搞点幺蛾子, 他要用阶乘幂来搞出一个大数。

$$f(n) = n^{(n-1)(n-2)\dots 2^1}$$

例如: $f(1) = 1$, $f(5) = 5^{4 \cdot 3 \cdot 2^1} \approx 6.206 \times 10^{183230}$

由于结果很大, 所以 TSL 要将它对 m 取模。

Input

有多组测试数据 (数据组数 ≤ 50)。对于每组数据, 有一个两个整数 n, m 如题目所述。

Output

对于每组测试数据, 输出 $f(n) \bmod m$ 的值。

Limits

- $1 \leq n \leq 10^9$
- $1 \leq m \leq 10^9$

Sample input and output

Sample Input	Sample Output
2 42	2
5 123456789	16317634
94 265	39

Problem C. 异或

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

给定整数 m 以及 n 个整数 a_1, a_2, \dots, a_n , 将这 n 个数两两异或共能得到 $\frac{n(n-1)}{2}$ 个结果, 问这些结果中大于 m 的有多少个?

Input

有多组测试数据 (数据组数 ≤ 10)。对于每组数据, 第一行有两个整数 n, m 。第二行有 n 个整数, 中间以空格分割。

Output

对于每组测试数据, 输出一个整数表示所求得的答案。

Limits

- $1 \leq n \leq 10^5$
- $1 \leq m \leq 10^5$
- $1 \leq a_i \leq 10^5$

Sample input and output

Sample Input	Sample Output
3 10 6 5 10	2

Note

本题的输入数据量较大, 建议使用高效的读写方式: C++使用 `scanf/printf` 代替 `cin/cout`, Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

Problem D. 统计难题

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

给定 n 个整数 a_1, a_2, \dots, a_n 。定义一个子区间 $[l, r]$ 的价值为子区间内所有的数按位与的结果。即 $f(l, r) = a_l \& a_{l+1} \& a_{l+2} \& \dots \& a_r$ 。请问所有子区间的价值之和是多少？即计算

$$\sum_{l=1}^n \sum_{r=l}^n f(l, r)$$

Input

有多组测试数据（数据组数 ≤ 20 ）。对于每组数据，第一行有一个整数 n 。第二行有 n 个整数，中间以空格分割。

Output

对于每组测试数据，输出一个整数表示所求得的答案。

Limits

- $1 \leq n \leq 10^5$
- $1 \leq a_i \leq 10^6$

Sample input and output

Sample Input	Sample Output
3	40
7 11 9	48
4	
11 9 6 11	

Note

本题的输入数据量较大，建议使用高效的读写方式：C++使用 `scanf/printf` 代替 `cin/cout`，Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

按位与运算可以在两个长度相同的二进制数上进行，即每一位做逻辑与运算。在 C/C++ 和 Java 中的运算符为 &。

对于第一个样例，答案的计算方式如下：

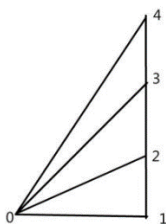
$$\begin{aligned} & f(1, 1) + f(1, 2) + f(1, 3) + f(2, 2) + f(2, 3) + f(3, 3) \\ &= (7) + (7 \& 11) + (7 \& 11 \& 9) + (11) + (11 \& 9) + (9) = 40 \end{aligned}$$

Problem E. 生成树计数

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

OneDay 最近对生成树（包含所有顶点的联通无环子图。）非常的感兴趣，他想知道对于特定的简单平面无向图是不是存在求生成树个数的简单方法。

OneDay 定义了这样的图：一个以 $\{0, 1, 2, \dots, n\}$ 为顶点的图，顶点 0 与其它 n 个顶点直接相连，对于顶点 i ($1 \leq i < n$)，顶点 i 与顶点 $i + 1$ 连有一条边。下面是 $n = 4$ 的图：



请问对于任意 n ，这样的图的生成树个数。

Input

有多组测试数据（数据组数 ≤ 50 ）。对于每组数据，有一个整数 n 。

Output

对于每组测试数据，输出一个整数表示所求得的答案（对 $10^9 + 7$ 取模）。

Limits

- $1 \leq n \leq 10^9$

Sample input and output

Sample Input	Sample Output
1	1
2	3

Problem F. 喵神的股票

Input file: Standard Input
Output file: Standard Output
Time limit: 2 seconds
Memory limit: 128 megabytes

喵神最近在关注股票，为了计算股票可能的盈利，他获取了一只股票最近 n 天的价格 a_1, a_2, \dots, a_n 。喵神每天可以在下列三种操作中选取一种：

1. 什么也不做；
2. 按照当天的价格买进一个单位的股票；
3. 按照当天的价格卖出部分或所有的股票。

现在喵神想知道，如果在 n 天前手中持有的股票数为0，并且假设有无限的钱，在这 n 天结束能够获得的最大利润是多少？

Input

有多组测试数据（数据组数 ≤ 10 ）。对于每组数据，第一行有一个整数 n ，第二行有 n 个整数 a_1, a_2, \dots, a_n 。

Output

对于每组测试数据，输出一个整数表示这 n 天结束能够获得的最大利润。

Limits

- $1 \leq n \leq 10^6$
- $1 \leq a_i \leq 100$

Sample input and output

Sample Input	Sample Output
5 1 2 3 4 5	10

Note

本题的输入数据量较大，建议使用高效的读写方式：C++使用 `scanf/printf` 代替 `cin/cout`，Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

Problem G. CSL 的字符诱惑

Input file: Standard Input
Output file: Standard Output
Time limit: 10 seconds
Memory limit: 256 megabytes

CSL 今天拿到了一个由小写字母构成的字符串，但是他觉得这个字符串不好，他希望通过重新排列这个字符串中的字符把它变成好的。他认为一个好的字符串 a 满足：字符串长度 m 为偶数，且 $\forall i(1 \leq i \leq m) a_i \neq a_{m-i+1}$

定义一个用户体验度 E ：字符串中的每个字符有它的价值 v_i ，如果最后构成的字符串中某个位置的字符 t_i 与原来的字符 s_i 相同，用户体验度就增加 v_i 。即用户体验度

$$E = \sum_{s_i=t_i} v_i$$

请帮 CSL 算一下，他最多可以获得多少体验度。

Input

有多组测试数据（数据组数 ≤ 120 ）。对于每组数据，第一行有一个正整数 n ，表示字符串的长度；第二行是一个由小写字母构成的字符串，保证可以变成好的字符串；第三行有 n 个数， v_1, v_2, \dots, v_n ，表示第 i 个字符的价值。

Output

对于每组测试数据，输出一个整数表示 CSL 可以获得的最大体验度。

Limits

- $2 \leq n \leq 100$
- $1 \leq v_i \leq 100$
- 保证 n 为偶数

Sample input and output

Sample Input	Sample Output
8 abacabac 1 1 1 1 1 1 1 1 8 abaccaba 1 2 3 4 5 6 7 8 8 abacabca 1 2 3 4 4 3 2 1	8 26 17

Note

本题的输入数据量较大，建议使用高效的读写方式：C++使用 `scanf/printf` 代替 `cin/cout`，Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

第一个样例：已经是好字符串，体验度为 $1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 = 8$ ；

第二个样例：变成 `bacacaba`，体验度为 $5 + 6 + 7 + 8 = 26$ ；

第三个样例：变成 `baacabca`，体验度为 $3 + 4 + 4 + 3 + 2 + 1 = 17$ 。

Problem H. TSL 序列

Input file: Standard Input
Output file: Standard Output
Time limit: 5 seconds
Memory limit: 256 megabytes

TSL 序列的定义是满足 $|a_i - a_{i-1}| \leq d$, ($0 < i \leq |a|$) 的序列。现在给你一个有 n 个元素的序列, 求出这个序列的最长 TSL 子序列的长度。

Input

有多组测试数据, (数据组数 ≤ 20)。对于每组数据, 第一行有两个整数 n, d 如题目所述。第二行有 n 个数 a_1, a_2, \dots, a_n , 中间以空格分割。

Output

对于每组测试数据, 输出一个整数表示该序列的最长 TSL 子序列的长度。

Limits

- $1 \leq n \leq 10^5$
- $0 \leq d \leq 10^8$
- $0 \leq a_i \leq 10^8$

Sample input and output

Sample Input	Sample Output
5 2 1 4 3 6 5 5 0 1 2 3 4 5	3 1

Note

本题的输入数据量较大, 建议使用高效的读写方式: C++ 使用 `scanf/printf` 代替 `cin/cout`, Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。
第一个样例: 最长 TSL 子序列为 1,3,5, 长度为 3

Problem I. OneDay 的小球

Input file: Standard Input
Output file: Standard Output
Time limit: 2 seconds
Memory limit: 128 megabytes

OneDay 有 n 个不同的盒子，起初在第一个盒子中有 n 种不同颜色的小球。

OneDay 有些无聊，想搞点幺蛾子。他想要把这些小球分别放到对应的盒子中，即第 i 个盒子中 ($1 \leq i \leq n$) 存放所有颜色为 i 的小球。为了做到这一点，OneDay 可以进行以下操作：

1. 选择任意一个非空的盒子，拿出盒子里所有的小球；
2. 选择 k 个空盒子（第 1 步中选择的盒子变成了空盒子，此处也可选）将上一步取出的球分成 k 个非空集合，分别放入这 k 个盒子中。他只能选择 $k = 2$ 或 $k = 3$ 。

OneDay 每一次操作的代价为他每次在第 1 步中拿出的小球的个数。那么整个过程他所需花费的最小代价是多少？

Input

有多组测试数据，（数据组数 ≤ 20 ）。对于每组数据，第一行有两个整数 n 表示盒子和颜色的数量。第二行有 n 个数 a_1, a_2, \dots, a_n ，表示第 i 种颜色小球的个数，中间以空格分割。

Output

对于每组测试数据，输出一个整数表示 OneDay 的最小代价。

Limits

- $1 \leq n \leq 2 \cdot 10^5$
- $1 \leq a_i \leq 10^9$

Sample input and output

Sample Input	Sample Output
3 1 2 3 4 2 3 4 5	6 19

Note

本题的输入数据量较大，建议使用高效的读写方式：C++使用 `scanf/printf` 代替 `cin/cout`，Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

第一个样例：从第一个盒子中拿出所有的球，选择 $k = 3$ ，将三种颜色的球分别放入对应的盒子中。总代价为6。

第二个样例需要两步：

1. 从第一个盒子中拿出所有的球，选择 $k = 3$ ，把所有颜色3的球放入第三个盒子，颜色4的球放入第四个盒子，其余的球放回第一个盒子。代价为14；
2. 从第一个盒子中拿出所有的球，选择 $k = 2$ ，把所有颜色1的球放入第一个盒子，颜色2的球放入第二个盒子。代价为5。

总代价为19。

Problem J. 利普希茨序列

Input file: Standard Input
Output file: Standard Output
Time limit: 1 second
Memory limit: 128 megabytes

对于函数 $f: \mathbb{R} \rightarrow \mathbb{R}$, 若 $\exists K \in \mathbb{R}$ (K 为常数), 不等式 $|f(x) - f(y)| \leq K|x - y|$ 对 $\forall x, y \in \mathbb{R}$ 恒成立, 称此函数利普希茨连续。今天我们研究这货的离散版本……

对于一个序列 $h[1..n]$, 我们定义利普希茨常数如下:

- 如果 $n < 2$, $L(h) = 0$
- 如果 $n \geq 2$, $L(h) = \max_{1 \leq i < j \leq n} \left\lceil \frac{|h[j] - h[i]|}{|j - i|} \right\rceil$

换句话说, $L = L(h)$ 是满足对 $\forall 1 \leq i, j \leq n$, $|h[i] - h[j]| \leq L|i - j|$ 恒成立的最小非负整数。

给定你一个长度为 n 的序列 $a[1..n]$ 以及 q 个查询。对于每个查询 $[l, r]$, 考虑子序列 $s = a[l..r]$, 试确定 s 的所有连续子序列的利普希茨常数之和

Input

有多组测试数据, (数据组数 ≤ 20)。对于每组数据, 第一行有两个整数 n 和 q , 表示序列 a 的长度以及查询的个数。第二行有 n 个数 a_1, a_2, \dots, a_n , 中间以空格分割。接下来 q 行每行两个整数 l, r 。

Output

对于每组测试数据, 每个查询输出一个整数表示序列 $a[l..r]$ 所有子序列的利普希茨常数之和。每组测试数据间空一行。

Limits

- $2 \leq n \leq 10^5$
- $1 \leq q \leq 100$
- $0 \leq a_i \leq 10^8$
- $1 \leq l < r \leq n$

Sample input and output

Sample Input	Sample Output
10 4	17
1 5 2 9 1 3 4 2 1 7	82
2 4	23
3 8	210
7 10	
1 9	2
7 6	0
5 7 7 4 6 6 2	22
1 2	59
2 3	16
2 6	8
1 7	
4 7	
3 5	

Note

本题的输入数据量较大，建议使用高效的读写方式：C++使用 `scanf/printf` 代替 `cin/cout`，Java 使用 `BufferedReader/PrintWriter` 代替 `Scanner/System.out`。

第一个样例的第一个查询：

$$L([5,2]) = 3$$

$$L([2,9]) = 7$$

$$L([5,2,9]) = 7$$

请注意输出格式，没有 Presentation Error。