

华中农业大学出题组

华中农业大学第十四届程序设计竞赛

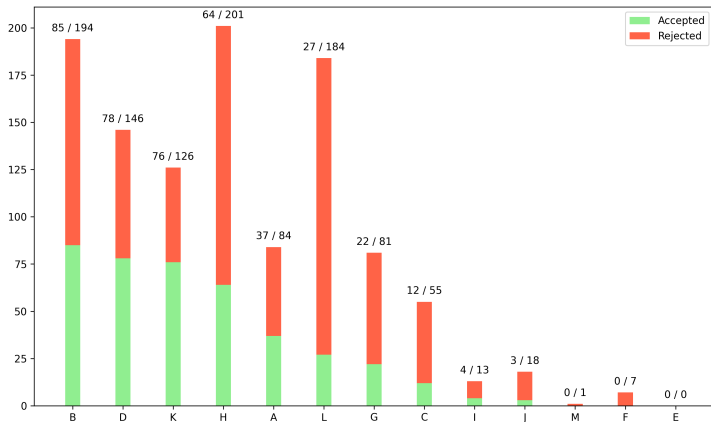
2025 年 4 月 13 日

概况

- Easy: B D
- Medium-Easy: H K
- Medium: A C G
- Medium-Hard: E I J
- Hard: F M
- Guess: L

概况

封榜前，各题提交情况如下：



B. 午餐！

题目大意
 给定 a_1, a_2, a_3 ，通过交换操作使其满足 $a_1 > a_2$ 且 $a_2 < a_3$ 。

- 实际上，充分的交换操作意味着我们可以对序列随意排列。由 $a_1 > a_2$ 且 $a_2 < a_3$ 可知 a_2 是 a_1, a_2, a_3 中最小的数。
- 我们可以对 $\{a_1, a_2, a_3\}$ 排序，设排序后的序列为 $\{b_1, b_2, b_3\}$ 。只需要判断是否满足 $b_1 < b_2$ 且 $b_1 < b_3$ 即可。

D. 怯战蜥蜴 V

题目大意

给定序列 $a_1, a_2, \dots, a_n$ 。求 $x (1 \leq x \leq n)$ ，使得不存在等于 x 的 a_i 。

- 只需要判断 $a_1, a_2, \dots, a_n$ 是否两两不等。如果 $a_1, a_2, \dots, a_n$ 两两不等，则无解。
- 否则使用记忆化数组记录没有出现在 $a_1, a_2, \dots, a_n$ 中的数并输出即可。
- 时间复杂度 $O(n)$ 。

H. 防线部署

题目大意

找出满足横坐标相等或纵坐标相等的二元组的个数。

- 对于一个给定的横坐标 x_t ，假设存在 m 个点满足横坐标等于 x_t ，那么这个横坐标对答案的贡献即为 $\frac{m(m-1)}{2}$ ；纵坐标同理。
- 实现上使用 `std::map` 或者直接使用数组统计即可。

K. ruskal 重构数

题目大意

给定一个数。最多执行一次交换相邻位的操作，最大化结果。

- 把数看成字符串 s ，找到最小的 i ，使得 $s_i < s_{i+1}$ ，交换 s_i 与 s_{i+1} 即可。
- 如果这样的 i 不存在，则不执行任何操作。

K. ruskal 重构数

证明

Case 3

如果 $x \geq 2$, 则存在 $i_1, i_2, \dots, i_x$ 使得 $s_{i_p} < s_{i_{p+1}} (1 \leq p \leq x)$ 。

我们不妨设 $\{i_p\}$ 是递增的。容易得到, 对于所有满足 $2 \leq p \leq x$ 的 p , 都有 $i_1 < i_p$ 。

如果我们交换 s_{i_1} 与 s_{i_1+1} , 那么第 i_1 位即为 s_{i_1+1} ; 如果我们交换 s_{i_p} 与 s_{i_p+1} , 那么第 i_1 位即为 s_{i_1} 。

而 $s_{i_1} < s_{i_1+1}$ 。所以最优的方案是交换 s_{i_1} 与 s_{i_1+1} 。

证毕。

A. 新踏上旅程

题目大意

选拔分为训练赛和过题情况两个方面的考核。需要分别计算每个选手在这两个方面的分数。

- 按照题意模拟即可。注意在模拟过程中的精度问题：计算单场训练赛积分时可以直接使用整型的运算而不是浮点型；计算最高的 k 场的单场训练赛积分的算术平均值时应先将这 k 场积分求和后再除以 k 。
- 时间复杂度 $O(n(m + k + n \log n))$ 。

q 次询问, 每次询问给定 l 与 r , 求 $\min_{k=l}^{r-1} \left\{ \frac{1}{\sum_{i=l}^k a_i} + \frac{1}{\sum_{i=k+1}^r a_i} \right\}$ 的值。

- 注意到: $\frac{1}{\sum_{i=l}^k a_i} + \frac{1}{\sum_{i=k+1}^r a_i} = \frac{\sum_{i=l}^r a_i}{\sum_{i=l}^k a_i \times \sum_{i=k+1}^r a_i}$
- 而对于给定的 l 与 r , $\sum_{i=l}^r a_i$ 是固定的, 所以我们只需要最大化 $\sum_{i=l}^k a_i \times \sum_{i=k+1}^r a_i$ 。

- 又因为 $\sum_{i=l}^k a_i + \sum_{i=k+1}^r a_i = \sum_{i=l}^r a_i$, 所以 $\left| \sum_{i=l}^k a_i - \sum_{i=k+1}^r a_i \right|$ 越小, $\sum_{i=l}^k a_i \times \sum_{i=k+1}^r a_i$ 越大。

C. 怯战蜥蜴 IV

- 又因为 $\sum_{i=l}^k a_i - \sum_{i=k+1}^r a_i$ 是随 k 单调的，我们只需要找到 k_1 与 k_2 ，使得：
 - ① $\sum_{i=l}^{k_1} a_i \leq \sum_{i=k_1+1}^r a_i$ 且 $\sum_{i=l}^{k_1} a_i - \sum_{i=k_1+1}^r a_i$ 尽可能接近 0
 - ② $\sum_{i=l}^{k_2} a_i \geq \sum_{i=k_2+1}^r a_i$ 且 $\sum_{i=l}^{k_2} a_i - \sum_{i=k_2+1}^r a_i$ 尽可能接近 0
 并比较选择 k_1 与 k_2 的结果即可。
- 使用前缀和处理 $\{a\}$ 的区间和，时间复杂度为 $O(n + q(\log n + \log V))$ 。

G. 喜欢数学的不是波奇酱

题目大意

给定区间 $[L, R]$ ，求出在这个区间内满足所有的相邻位数的差的绝对值为 1 的数的数量。

- 注意到合法的值域不超过 10^{18} ，即：合法的数的长度最多为 19 位。我们可以枚举出所有满足所有的相邻位数的差的绝对值为 1 的数，判断是否在给定区间内即可。
- 时间复杂度为 $O(9 \times 2^k)$ ，其中 k 为枚举值域的位数。

E. 奇思妙想波奇酱

题目大意

对于给定的 $b_1, b_2, \dots, b_{n+k}$ ，判断是否存在原序列 $a_1, a_2, \dots, a_n$ 。

- 设 dp_i 表示 $b_1, b_2, \dots, b_i$ 是否存在对应的原序列。考虑 dp_i 可以用于更新的状态。

- $$b_{i+1} = \sum_{t=i+2}^j (t-i-1)^2 \times b_t$$

或

$$b_j = \sum_{t=i+1}^{j-1} (t-i)^2 \times b_t$$

那么 $dp_j = 1$ 。最后只需考虑 dp_{n+k} 的值即可。

- 注意到 $\sum_{i=l}^r (i-l+1)^2 \times a_i \geq \sum_{i=l}^r (i-l+1)^2$ 。
- 设 $x = r - l + 1$ ，即划分子段的长度为 x 。
- 那么， $\sum_{i=l}^r (i-l+1)^2 = \frac{x(x+1)(2x+1)}{6} \leq 10^8$ 。得到 $x \leq 669$ 。
- 也就是说，对于给定的 i ，在枚举可能更新到的 dp_j 时，我们只需要考虑满足划分子段长度不超过 669 的部分。
- 在实现时，注意题目中给定的序列长度为 n 。
- 时间复杂度为 $O(669n)$ 。

I. 多远都要在一起

题目大意

给定一个 n 个点 n 条边的无向连通图。有多次询问，每次询问给定 k ，需要回答是否能够恰好走 k 步，从起点走到终点。

- 首先，一个 n 个点 n 条边的无向连通图是一棵基环树。

- 分析从起点到终点最短路径的长度。如果从起点到终点的最短路径的长度大于 k ，答案一定是否定的。
- 否则，分以下几种情况考虑。
 - ① 环的长度为偶数
 - ② 环的长度为奇数

- 此外，需要注意起点和终点在基环树上同一棵树的情况。这种情况需要使用 LCA 计算最短路径。

J. 技能树

题目大意

给定一棵以 1 号节点为根的技能树，每个节点有一个战力值。点亮一个节点需要满足其父节点已被点亮（根节点可直接点亮），且每点亮一个节点消耗 1 金币。点亮某个节点获得的战斗力是该节点到根路径上所有节点战力值排序后的中位数。求使用不超过 m 枚金币时能获得的最大战斗力。

注意

本题中位数定义与标准的中位数定义略有不同。

J. 技能树

- 首先，我们可以预处理每个节点可以贡献的战斗力。我们只需要使用 dfs 遍历一遍整棵树，维护一棵权值线段树。
- 在访问到一个节点时，将这个节点对应的战力值插入权值线段树，在回溯时删除即可。

- 接下来，我们定义 $dp_{u,k}$ 表示在以 u 为根的子树中选 k 个节点能获得的最大战斗力。
- 对于 u 的每个子节点 v ，我们有：

$$dp_{u,i+j} = \max_v \{dp_{u,i} + dp_{v,j}\}$$

- 其中，对于每个 u ，我们初始化 $dp_{u,1} = v_u$ ，即只选择 u 获得的战斗力。
- 在这个过程中，我们相当于对每个节点 u 保证选取自己，然后对其子节点进行 0-1 背包。最后答案即为 $\max_k \{dp_{u,k}\}$ 。
- 时间复杂度为 $O(n \log n + nm)$ 。

F. 不喜欢数学的波奇酱

题目大意

定义 $Bq(x, y, v) = d(\gcd(x, y))^{[d(\gcd(x, y)) \leq v]}$, 求 $\prod_{i=1}^n \prod_{j=1}^m Bq(i, j, v)$ 。

我们有：

$$\begin{aligned}
 \prod_{i=1}^n \prod_{j=1}^m d(\gcd(i, j)) &= \prod_{i=1}^n \prod_{j=1}^m d(x) [\gcd(i, j) = x] \\
 &= \prod_{x=1}^{\min(n, m)} d(x)^{\sum_{i=1}^n \sum_{j=1}^m [\gcd(i, j) = x]}.
 \end{aligned}$$

F. 不喜欢数学的波奇酱

考虑指数部分：

$$\begin{aligned}
 \sum_{i=1}^n \sum_{j=1}^m [\gcd(i, j) = x] &= \sum_{i=1}^{\lfloor \frac{n}{x} \rfloor} \sum_{j=1}^{\lfloor \frac{m}{x} \rfloor} [\gcd(i, j) = 1] \\
 &= \sum_{i=1}^{\lfloor \frac{n}{x} \rfloor} \sum_{j=1}^{\lfloor \frac{m}{x} \rfloor} \sum_{k=1}^{\min(\lfloor \frac{n}{x} \rfloor, \lfloor \frac{m}{x} \rfloor)} \mu(k) [k \mid i] [k \mid j] \\
 &= \sum_{k=1}^{\min(\lfloor \frac{n}{x} \rfloor, \lfloor \frac{m}{x} \rfloor)} \mu(k) \sum_{i=1}^{\lfloor \frac{n}{x} \rfloor} [k \mid i] \sum_{j=1}^{\lfloor \frac{m}{x} \rfloor} [k \mid j] \\
 &= \sum_{k=1}^{\min(\lfloor \frac{n}{x} \rfloor, \lfloor \frac{m}{x} \rfloor)} \mu(k) \left\lfloor \frac{n}{kx} \right\rfloor \left\lfloor \frac{m}{kx} \right\rfloor.
 \end{aligned}$$

令 $kx = T$, 则 $k = \frac{T}{x}$, 我们有:

$$\sum_{k=1}^{\min(\lfloor \frac{n}{x} \rfloor, \lfloor \frac{m}{x} \rfloor)} \mu(k) \left\lfloor \frac{n}{kx} \right\rfloor \left\lfloor \frac{m}{kx} \right\rfloor = \sum_{T \leq \min(n, m)} \mu\left(\frac{T}{x}\right) \left\lfloor \frac{n}{T} \right\rfloor \left\lfloor \frac{m}{T} \right\rfloor$$

F. 不喜欢数学的波奇酱

进而：

$$\begin{aligned}
 \prod_{x=1}^{\min(n,m)} d(x)^{\sum_{i=1}^n \sum_{j=1}^m [\gcd(i,j)=x]} &= \prod_{x=1}^{\min(n,m)} d(x)^{\sum_{T|x} T \leq \min(n,m)} \lfloor \frac{n}{T} \rfloor \lfloor \frac{m}{T} \rfloor \\
 &= \prod_{x=1}^{\min(n,m)} \prod_{\substack{T|x \\ T \leq \min(n,m)}} d(x)^{\mu(\frac{T}{x}) \lfloor \frac{n}{T} \rfloor \lfloor \frac{m}{T} \rfloor} \\
 &= \prod_{\substack{T|x \\ T \leq \min(n,m)}} \left(\prod_{x=1}^{\min(n,m)} d(x)^{\mu(\frac{T}{x})} \right)^{\lfloor \frac{n}{T} \rfloor \lfloor \frac{m}{T} \rfloor}.
 \end{aligned}$$

也就是说：

$$\prod_{i=1}^n \prod_{j=1}^m d(\gcd(i, j)) = \prod_{\substack{T|x \\ T \leq \min(n, m)}} \left(\prod_{x=1}^{\min(n, m)} d(x)^{\mu(\frac{T}{x})} \right)^{\lfloor \frac{n}{T} \rfloor \lfloor \frac{m}{T} \rfloor}$$

- 不妨令 $F(T) = \prod_{x=1}^{\min(n,m)} d(x)^{\mu(\frac{T}{x})} = \prod_{x|T} d(x)^{\mu(\frac{T}{x})}$ 。
- 对于任意 x ，只有当 $d(x) \leq v$ 时， $d(x)$ 才可能对 $F(T)$ 产生贡献。
- 考虑将 q 次询问离线并按 v 进行升序排序，
- 记 q 次询问为 $v_1, v_2, \dots, v_q$ 。考虑 v 的取值对 $F(T)$ 的影响，记这 q 次询问对应的 $F(T)$ 分别为 $F_1(T), F_2(T), \dots, F_q(T)$ 。

F. 不喜欢数学的波奇酱

- 在所有询问前, $F_i(T) = 1 (1 \leq i \leq q)$ 。
- 在第 i 次询问结束后, 对于所有正整数 x , 如果 $v_i < d(x) \leq v_{i+1}$, 那么对于所有的 $i < j \leq q$, 都有 $F_j(T) \leftarrow F_j(T) \times d(x)^{\mu(\frac{T}{x})}$, 可以使用树状数组维护。
- 维护完毕 $F(T)$ 后, 使用数论分块计算答案即可。
- 时间复杂度为 $O(q\sqrt{n}(\log(V) + \log(n)) + n \log^2 n)$, 其中 V 为模数的大小。

M. 染上你的颜色

题目大意

给定一棵树，维护树上的染色与查询操作。

- 因为 $\frac{i \times j}{\text{lcm}(i, j)} = \text{gcd}(i, j)$ ，所以

$$\text{lcm}^2(i, j) - (i + j) \times \text{lcm}(i, j) = i \times j$$

等价于

$$\text{lcm}(i, j) = i + j + \text{gcd}(i, j)$$

M. 染上你的颜色

- 不妨设 $i \leq j$ 。当 $i = j$ 时，等式显然不成立。
- 当 $i < j$ 时，有 $1 \leq \gcd(i, j) \leq i < j$ ，所以：

$$j < i + j + \gcd(i, j) < 3 \times j$$

- 又因为 $i + j + \gcd(i, j) = \text{lcm}(i, j)$ ，所以一定有：

$$j \mid i + j + \gcd(i, j)$$

- 于是 $i + j + \gcd(i, j) = 2 \times j$ 。

M. 染上你的颜色

- 进而：

$$\frac{i \times j}{\gcd(i, j)} = \text{lcm}(i, j) = i + j + \gcd(i, j) = 2 \times j$$

于是：

$$i = 2 \times \gcd(i, j)$$

$$j = 3 \times \gcd(i, j)$$

所以 $\frac{i}{j} = \frac{2}{3}$ 。

M. 染上你的颜色

定义：最大连通集

定义满足以下条件的非空集合 S 为最大连通集：

- ① 对于 S 中的任意两个不相等的元素 $i, j (i < j)$ ，存在正整数 p ，使得 $i \times 3^p = j \times 2^p$ 。
 - ② 任取不大于 n 的正整数 x ，如果存在 S 中的不等于 x 的元素 y 和正整数 p ，使得 $x \times 3^p = y \times 2^p$ ，那么一定有 $x \in S$ 。
- 举例：对于 $n = 100$ ， $S = \{8, 12, 18, 27\}$ 是一个最大连通集。而 $\{8, 12, 18, 27, 30\}$ 与 $\{12, 18, 27\}$ 都不是。

M. 染上你的颜色

- 根据题意，在波奇酱召唤波奇精灵对编号为 i 的区域施加染色魔法时，编号与 i 在同一最大连通集内的区域同样会被施加染色魔法。
- 于是，我们可以预处理出所有的最大连通集。
- 显然单个最大连通集的大小一定是对数级别的。接下来，我们可以将操作 1 视为多次修改树上节点权值。
- 我们只需要考虑如何动态维护树上最大权独立集，使用 ddp 维护即可。

L. 贪心的世界

证明

设题述原序列为 A ，有 $A_1 = y, A_2 = A_3 = \dots = A_k = x$ ，记最后的答案为 s_0 。

现有一段长度为 k 的序列 a ，有 $a_1 = y - x, a_2 = a_3 = \dots = a_k = 0$ ，记对此序列按题意操作若干次后得到的答案为 s_1 。我们有结论：

$$s_0 = s_1 + x。$$

L. 贪心的世界

证明

引理

- ① 引理 1: 对于任意正整数 a, b, c , 有 $\lfloor \frac{(a+c)+(b+c)}{2} \rfloor = \lfloor \frac{a+b}{2} \rfloor + c$
- ② 引理 2: 对于任意正整数 t , 有 $\lfloor \frac{\lfloor \frac{t}{2} \rfloor}{2} \rfloor = \lfloor \frac{t}{4} \rfloor$

引理显然。

L. 贪心的世界

结论 $s_0 = s_1 + x$ 证明

先证明结论： $s_0 = s_1 + x$ 。

设 x, y 为任意整数。

对于长度为 2 的序列 $A \{y, x\}$ 与序列 $B \{y-x, 0\}$ ，对于序列 A 经过且仅经过一次操作后即可获得最终答案 $s_1 = \lfloor \frac{y+x}{2} \rfloor = \lfloor \frac{(y-x+x)+(0+x)}{2} \rfloor$ ，根据引理 1，可得： $s_1 = \lfloor \frac{y-x}{2} \rfloor + x$ 。而对于序列 B ，经过且经过一次操作后同样可以得到最终答案： $s_2 = \lfloor \frac{y-x}{2} \rfloor$ 。对比两式，我们不难发现

$s_1 = s_2 + x$ ，结论成立。

我们假设对于长度为 k 的序列 $A \{y, x, x, \dots, x\}$ 与序列 $B \{y-x, 0, 0, \dots, 0\}$ ，两者经过若干次操作后得到的最终答案为 s_1 与 s_2 ，满足 $s_1 = s_2 + x$ 。

L. 贪心的世界

结论 $s_0 = s_1 + x$ 证明

现考虑长度 $k+1$ 的序列 $A \{y, x, x, \dots, x\}$ 与序列 $B \{y-x, 0, 0, \dots, 0\}$ ，根据假设前提，对于序列 A 、 B 的前 k 项，我们可以通过若干次操作使得两个序列分别变为 $\{s_1, s_1, s_1, \dots, s_1, x\}$ 与 $\{s_2, s_2, s_2, \dots, s_2, 0\}$ ，其中 $s_1 = s_2 + x$ 。我们会发现对于序列 A 的任意一次操作（不妨假设当前操作的下标为 i, j ，即将 A_i 与 A_j 转化为 $\lfloor \frac{A_i + A_j}{2} \rfloor$ ），我们都可以在序列 B 中选择同样的两个下标 i, j 进行操作（即将 B_i 与 B_j 转化为 $\lfloor \frac{B_i + B_j}{2} \rfloor$ ）。可以发现，总存在 $\lfloor \frac{A_i + A_j}{2} \rfloor = \lfloor \frac{B_i + B_j}{2} \rfloor + x$ 。所以对于序列 A ， B 得到的最终答案 S_1, S_2 ，满足 $S_1 = S_2 + x$ 。

结论 $s_0 = s_1 + x$ 证毕。

L. 贪心的世界

证明

旁证：最优策略

在此之前，我们需要理解序列中 0 所起到的作用。

在任意操作过程中，设当前最小正整数为 x_1 ，如果我们选择任意两个正整数进行操作（不妨设此时对 a_i 与 a_j 进行操作，其中 $i < j$ ），得到：

$$a_i = a_j = \left\lfloor \frac{a_i + a_j}{2} \right\rfloor$$

会发现此时新的最小正整数 x_2 满足 $x_2 \geq x_1$ 。

L. 贪心的世界

证明

旁证：最优策略

原因是此时 $a_i, a_j \geq x_1$ ，所以：

$$\left\lfloor \frac{a_i + a_j}{2} \right\rfloor \geq \left\lfloor \frac{x_1 + x_1}{2} \right\rfloor = x_1$$

证明

构造与反证

设初始序列为 $a = \{t, 0, 0, \dots, 0\}$, 共 k 项。若我们按照上述策略连续执行 m 次操作 ($1 \leq m \leq k-1$), 则序列状态如下:

$$a_1 = \left\lfloor \frac{t}{2} \right\rfloor, \quad a_2 = \left\lfloor \frac{t}{2^2} \right\rfloor, \quad \dots, \quad a_m = a_{m+1} = \left\lfloor \frac{t}{2^m} \right\rfloor, \quad 0, 0, \dots$$

L. 贪心的世界

证明

构造与反证

我们使用反证法：

若第 $m+1$ 次操作不按策略（即不选 a_{m+1}, a_{m+2} ），而是选择 a_{m-1} 与 a_{m+2} ：

$$a_{m-1} = a_{m+2} = \left\lfloor \frac{\left\lfloor \frac{t}{2^{m-1}} \right\rfloor + 0}{2} \right\rfloor = \left\lfloor \frac{t}{2^m} \right\rfloor$$

L. 贪心的世界

证明

构造与反证

此时有：

$$a_{m-1} = a_m = a_{m+1} = a_{m+2}$$

即本轮操作没有减小最小正整数。

构造与反证

$$a_{m+1} = a_{m+2} = \left\lfloor \frac{\left\lfloor \frac{t}{2^m} \right\rfloor + 0}{2} \right\rfloor = \left\lfloor \frac{t}{2^{m+1}} \right\rfloor$$
$$2 \times \left\lfloor \frac{t}{2^{m+1}} \right\rfloor \leq \left\lfloor \frac{t}{2^m} \right\rfloor$$

即最小正整数被减小，原策略更优。若选更早位置如 a_k 与 a_{m+2} ，结论相同。

L. 贪心的世界

证明

待序列中不存在 0 时，该序列应形如：

$$a_1 = \lfloor \frac{t}{2^{k-1}} \rfloor, a_2 = \lfloor \frac{t}{2^{k-1}} \rfloor, a_3 = \lfloor \frac{t}{2^{k-2}} \rfloor, \dots, a_{k-1} = \lfloor \frac{t}{2^2} \rfloor, a_k = \lfloor \frac{t}{2} \rfloor$$

根据题意，有： $0 \leq t \leq 2^k - 1$ ， 我们进行如下分类讨论：

Case 1

由于 $t \leq 2^{k-1} - 1 < 2^{k-1}$, 则 $a_1 = a_2 = \lfloor \frac{t}{2^{k-1}} \rfloor = 0$ 恒成立。
且由于 $t < 2^{k-1}$, 不等式两边同除以 2^{k-2} , 得到

$$\lfloor \frac{t}{2^{k-2}} \rfloor < \lfloor \frac{2^{k-1}}{2^{k-2}} \rfloor = 2.$$

又因为 $\lfloor \frac{t}{2^{k-2}} \rfloor \in \mathbb{N}$, 所以有

$$\lfloor \frac{t}{2^{k-2}} \rfloor \leq 1,$$

L. 贪心的世界

证明

Case 1

我们再将 $a_1 = 0$ 与 a_3, a_4 分别进行操作，可以使得 $a_3 = a_4 = 0$ 。
 以此类推，我们最终可以使得整个序列 a 的所有元素均变为 0。
 由上述结论得，原序列 A 的最终答案为 x 。

L. 贪心的世界

证明

Case 2

当 t 满足 $2^{k-1} \leq t \leq 2^k - 1$ 时:

对于形如 $\lfloor \frac{t}{2^m} \rfloor$ 的表达式 (其中 $1 \leq m \leq k-1$), 均有:

$$2^{k-m-1} \leq \left\lfloor \frac{t}{2^m} \right\rfloor \leq \left\lfloor \frac{2^k - 1}{2^m} \right\rfloor < 2^{k-m},$$

即:

$$2^{k-m-1} \leq \left\lfloor \frac{t}{2^m} \right\rfloor \leq 2^{k-m} - 1.$$

证明

Case 2

我们考虑两种极端情况:

情况一：当所有 $\lfloor \frac{t}{2^m} \rfloor = 2^{k-m} - 1$ 时，设当前序列的最终答案为 s_1 ，并构造一个新序列 b ，使得 $b_i = a_i + 1 = 2^{k-m}$ 。记 b 的最终答案为 s_2 ，则有：

$$s_2 = s_1 + 1.$$

根据构造, 序列 b 形如:

$$b_1 = 2, \quad b_2 = 2, \quad b_3 = 4, \quad \dots, \quad b_k = 2^{k-1}.$$

证明

Case 2

情况二: 当所有 $\lfloor \frac{t}{2^m} \rfloor = 2^{k-m-1}$ 时, 序列 a 形如:

$$a_1 = 1, \quad a_2 = 1, \quad a_3 = 2, \quad \dots, \quad a_k = 2^{k-2}.$$

对 a_2, a_3 进行操作:

$$a_2 = a_3 = \left| \frac{1+2}{2} \right| = 1,$$

以此类推，最终所有元素都变为 1，即 $s_1 = 1$ ，从而：

$$s_0 = s_1 + x = x + 1.$$

L. 贪心的世界

证明

Case 2

一般情况：由于序列 a 的最终答案 s_1 应在两种极端之间，且这两种情况都得到 $s_1 = 1$ ，所以一般情况也有：

$$s_1 = 1, \quad s_0 = x + 1.$$

