PtzCamp / Open Cup
趣题选讲

徐哲安

清华大学

August, 2021

题目大意[1]

- Chihiro 独立且等概率生成了 n 个在 $[1, 10^{18}]$ 中的随机整数 $a_1, a_2, \cdots, a_n$

[1]Winter 2017, Tsinghua U Deep Dark Fantasy Contest A

题目大意[1]

- Chihiro 独立且等概率生成了 n 个在 $[1, 10^{18}]$ 中的随机整数 $a_1, a_2, \cdots, a_n$
- Chihiro 选择了一个整数 m，接着生成了一个在 $[0, m)$ 中的随机整数 k，最后令 $b_i = (a_i + k) \bmod m$，并随机打乱得到序列 $\{b_i\}$

[1] Winter 2017, Tsinghua U Deep Dark Fantasy Contest A

题目大意[1]

- Chihiro 独立且等概率生成了 n 个在 $[1, 10^{18}]$ 中的随机整数 $a_1, a_2, \cdots, a_n$
- Chihiro 选择了一个整数 m，接着生成了一个在 $[0, m)$ 中的随机整数 k，最后令 $b_i = (a_i + k) \bmod m$，并随机打乱得到序列 $\{b_i\}$
- 给定序列 $\{a_i\}, \{b_i\}$，请求出一组合法的 m, k

[1] Winter 2017, Tsinghua U Deep Dark Fantasy Contest A

题目大意[1]

- Chihiro 独立且等概率生成了 n 个在 $[1, 10^{18}]$ 中的随机整数 $a_1, a_2, \cdots, a_n$
- Chihiro 选择了一个整数 m，接着生成了一个在 $[0, m)$ 中的随机整数 k，最后令 $b_i = (a_i + k) \bmod m$，并随机打乱得到序列 $\{b_i\}$
- 给定序列 $\{a_i\}, \{b_i\}$，请求出一组合法的 m, k
- $10^5 \le n \le 2 \times 10^5$，保证存在解满足 $0 \le k < m \le n^2$

[1] Winter 2017, Tsinghua U Deep Dark Fantasy Contest A

解答

- 设 $M = \max\{b_i\}$，那么 $m - M$ 期望是 $\Theta(n)$ 的，如果我们能快速 check 每一个 m，问题就得到解决了

解答

- 设 $M = \max\{b_i\}$，那么 $m - M$ 期望是 $\Theta(n)$ 的，如果我们能快速 check 每一个 m，问题就得到解决了
- 注意到 $\{a_i\}$ 的值都是真实的，b_i 间作差就可将 k 的影响去除。不难得到

$$na_{p_1} \equiv \sum(a_i) + \sum(b_1 - b_i) \pmod m$$

解答

- 设 $M = \max\{b_i\}$，那么 $m - M$ 期望是 $\Theta(n)$ 的，如果我们能快速 check 每一个 m，问题就得到解决了
- 注意到 $\{a_i\}$ 的值都是真实的，b_i 间作差就可将 k 的影响去除。不难得到

$$na_{p_1} \equiv \sum(a_i) + \sum(b_1 - b_i) \pmod m$$

- 观察到 $(a_{p_1} \bmod m)$ 至多有 $\gcd(n, m)$ 个解

解答

- 设 $M = \max\{b_i\}$，那么 $m - M$ 期望是 $\Theta(n)$ 的，如果我们能快速 check 每一个 m，问题就得到解决了
- 注意到 $\{a_i\}$ 的值都是真实的，b_i 间作差就可将 k 的影响去除。不难得到

$$na_{p_1} \equiv \sum(a_i) + \sum(b_1 - b_i) \pmod m$$

- 观察到 $(a_{p_1} \bmod m)$ 至多有 $\gcd(n, m)$ 个解
- 枚举 $(a_{p_1} \bmod m)$ 对于每个 $(a_{p_1} \bmod m)$ 都对应唯一的 k，之后暴力 check 即可，期望复杂度 $O(1)$

解答

- 设 $M = \max\{b_i\}$，那么 $m - M$ 期望是 $\Theta(n)$ 的，如果我们能快速 check 每一个 m，问题就得到解决了
- 注意到 $\{a_i\}$ 的值都是真实的，b_i 间作差就可将 k 的影响去除。不难得到

$$na_{p_1} \equiv \sum(a_i) + \sum(b_1 - b_i) \pmod m$$

- 观察到 $(a_{p_1} \bmod m)$ 至多有 $\gcd(n, m)$ 个解
- 枚举 $(a_{p_1} \bmod m)$ 对于每个 $(a_{p_1} \bmod m)$ 都对应唯一的 k，之后暴力 check 即可，期望复杂度 $O(1)$
- 期望时间复杂度 $O(n \log n)$

题目大意[2]

- Chihiro 有一张 $n \times m$ 的网格纸，网格 (i, j) 的颜色为 $c_{i,j}$

[2] Summer 2017, UniBuc Contest G

题目大意[2]

- Chihiro 有一张 $n \times m$ 的网格纸，网格 (i, j) 的颜色为 $c_{i,j}$
- 每次，Chihiro 会选择一条横轴线或者竖轴线，沿着轴线将较小的一侧折到较大的一侧去，要求两边对应网格颜色相同

[2] Summer 2017, UniBuc Contest G

题目大意[2]

- Chihiro 有一张 $n \times m$ 的网格纸，网格 (i, j) 的颜色为 $c_{i,j}$
- 每次，Chihiro 会选择一条横轴线或者竖轴线，沿着轴线将较小的一侧折到较大的一侧去，要求两边对应网格颜色相同
- 问存在多少子矩形可以通过折叠得到

[2] Summer 2017, UniBuc Contest G

题目大意[2]

- Chihiro 有一张 $n \times m$ 的网格纸，网格 (i, j) 的颜色为 $c_{i,j}$
- 每次，Chihiro 会选择一条横轴线或者竖轴线，沿着轴线将较小的一侧折到较大的一侧去，要求两边对应网格颜色相同
- 问存在多少子矩形可以通过折叠得到
- 数据范围：$nm \le 10^6$

[2] Summer 2017, UniBuc Contest G

解答

- 显然两维是独立的，我们只需要考虑一维的问题

解答

- 显然两维是独立的，我们只需要考虑一维的问题
- 可以把对折的过程看作：若存在一个回文中心能扩展到最左侧（或最右侧），就能把较短的一侧删去

解答

- 显然两维是独立的，我们只需要考虑一维的问题
- 可以把对折的过程看作：若存在一个回文中心能扩展到最左侧（或最右侧），就能把较短的一侧删去
- 定义合法前缀：每次贪心删去能删的最短前缀后，累计被删去的前缀。显然所有能被删去的前缀都是合法前缀

解答

- 显然两维是独立的，我们只需要考虑一维的问题
- 可以把对折的过程看作：若存在一个回文中心能扩展到最左侧（或最右侧），就能把较短的一侧删去
- 定义合法前缀：每次贪心删去能删的最短前缀后，累计被删去的前缀。显然所有能被删去的前缀都是合法前缀
- 区间 $[l, r]$ 能被得到的必要条件：$[1, l-1]$ 是合法前缀同时 $[r+1, n]$ 是合法后缀

解答

- 显然两维是独立的，我们只需要考虑一维的问题
- 可以把对折的过程看作：若存在一个回文中心能扩展到最左侧（或最右侧），就能把较短的一侧删去
- 定义合法前缀：每次贪心删去能删的最短前缀后，累计被删去的前缀。显然所有能被删去的前缀都是合法前缀
- 区间 $[l, r]$ 能被得到的必要条件：$[1, l-1]$ 是合法前缀同时 $[r+1, n]$ 是合法后缀
- 可以证明这也是一个充分条件

题目大意[3]

- n 个人计划争夺 n 个位置，位置 i 上有 a_i 枚金币

[3] Summer 2017, Warsaw U Contest E

题目大意[3]

- n 个人计划争夺 n 个位置，位置 i 上有 a_i 枚金币
- 每个人都会选择恰好一个目标，若某个目标被多人选择，那么会有恰好一个人（等概率随机）得到 a_i 枚金币

[3]Summer 2017, Warsaw U Contest E

题目大意[3]

- n 个人计划争夺 n 个位置，位置 i 上有 a_i 枚金币
- 每个人都会选择恰好一个目标，若某个目标被多人选择，那么会有恰好一个人（等概率随机）得到 a_i 枚金币
- 假设所有人都以相同的概率分布来决策，请问每个人的期望收益是多少？

[3] Summer 2017, Warsaw U Contest E

题目大意[3]

- n 个人计划争夺 n 个位置，位置 i 上有 a_i 枚金币
- 每个人都会选择恰好一个目标，若某个目标被多人选择，那么会有恰好一个人（等概率随机）得到 a_i 枚金币
- 假设所有人都以相同的概率分布来决策，请问每个人的期望收益是多少？
- 数据范围：$n \leq 5000$

[3] Summer 2017, Warsaw U Contest E

初步分析

- 给定 $a_1, a_2, \cdots, a_n$，求一组 $x_1, x_2, \cdots, x_n$，满足 $\sum x_i = n-1, x_i \in [0,1]$，最小化 $\sum_{i=1}^{n} a_i \cdot x_i^n$

初步分析

- 给定 $a_1, a_2, \cdots, a_n$，求一组 $x_1, x_2, \cdots, x_n$，满足 $\sum x_i = n - 1, x_i \in [0, 1]$，最小化 $\sum_{i=1}^{n} a_i \cdot x_i^n$
- 不妨令 $a_1 \geq a_2 \geq \cdots \geq a_n$

初步分析

- 给定 $a_1, a_2, \cdots, a_n$，求一组 $x_1, x_2, \cdots, x_n$，满足 $\sum x_i = n-1, x_i \in [0,1]$，最小化 $\sum_{i=1}^{n} a_i \cdot x_i^n$
- 不妨令 $a_1 \geq a_2 \geq \cdots \geq a_n$
- 注意到目标是关于 x_i 的齐次式，这启发我们可以两两合并得到答案：设 $f(k)$ 是前缀 $a_1, a_2, \cdots, a_k$ 对应的答案

初步分析

- 给定 $a_1, a_2, \cdots, a_n$，求一组 $x_1, x_2, \cdots, x_n$，满足 $\sum x_i = n-1, x_i \in [0,1]$，最小化 $\sum_{i=1}^{n} a_i \cdot x_i^n$
- 不妨令 $a_1 \geq a_2 \geq \cdots \geq a_n$
- 注意到目标是关于 x_i 的齐次式，这启发我们可以两两合并得到答案：设 $f(k)$ 是前缀 $a_1, a_2, \cdots, a_k$ 对应的答案
- 枚举 x_k 分配到的量，剩下的等比例缩放，有如下递推式：

$$f(k) = \min_{0 \leq x \leq 1} a_k \cdot x^n + f(k-1) \cdot \left(\frac{k-1-x}{k-2} \right)^n$$

初步分析

- 给定 $a_1, a_2, \cdots, a_n$，求一组 $x_1, x_2, \cdots, x_n$，满足 $\sum x_i = n-1, x_i \in [0,1]$，最小化 $\sum_{i=1}^{n} a_i \cdot x_i^n$
- 不妨令 $a_1 \geq a_2 \geq \cdots \geq a_n$
- 注意到目标是关于 x_i 的齐次式，这启发我们可以两两合并得到答案：设 $f(k)$ 是前缀 $a_1, a_2, \cdots, a_k$ 对应的答案
- 枚举 x_k 分配到的量，剩下的等比例缩放，有如下递推式：

$$f(k) = \min_{0 \leq x \leq 1} a_k \cdot x^n + f(k-1) \cdot \left(\frac{k-1-x}{k-2} \right)^n$$

- 显然这个式子覆盖了所有合法的情况。反之，我们可以归纳证明这个递推式得到了正确的解

解答

- 设问题规模不超过 $k-1$ 时得到了正确的解

解答

- 设问题规模不超过 $k-1$ 时得到了正确的解
- 若不合法的情况，即对于 $i \in I$ 有 $x_i > 1$，我们可对分布 $\{x_i\}$ 略作调整。发现

$$\sum_{i=1}^{k-1} \max(x_i - 1, 0) \le \sum_{i=1}^{k-1} x_i \cdot \frac{1 - x_k}{k - 1 - x_k} = 1 - x_k$$

我们将 $\max(x_i - 1, 0)$ 多出来的部分减去，并补到 x_k 上。由于 $a_k = \min\{a_i\}$，稍加计算发现新的解合法并更优

解答

- 设问题规模不超过 $k-1$ 时得到了正确的解
- 若不合法的情况，即对于 $i \in I$ 有 $x_i > 1$，我们可对分布 $\{x_i\}$ 略作调整。发现

$$\sum_{i=1}^{k-1} \max(x_i - 1, 0) \le \sum_{i=1}^{k-1} x_i \cdot \frac{1 - x_k}{k - 1 - x_k} = 1 - x_k$$

  我们将 $\max(x_i - 1, 0)$ 多出来的部分减去，并补到 x_k 上。由于 $a_k = \min\{a_i\}$，稍加计算发现新的解合法并更优
- 剩下就是如何计算递推式的问题了，设后者关于 x 的函数为 $g(x)$，求导后是个单增函数，二分求出零点即可

解答

- 设问题规模不超过 $k-1$ 时得到了正确的解
- 若不合法的情况，即对于 $i \in I$ 有 $x_i > 1$，我们可对分布 $\{x_i\}$ 略作调整。发现

$$\sum_{i=1}^{k-1} \max(x_i - 1, 0) \le \sum_{i=1}^{k-1} x_i \cdot \frac{1 - x_k}{k - 1 - x_k} = 1 - x_k$$

  我们将 $\max(x_i - 1, 0)$ 多出来的部分减去，并补到 x_k 上。由于 $a_k = \min\{a_i\}$，稍加计算发现新的解合法并更优
- 剩下就是如何计算递推式的问题了，设后者关于 x 的函数为 $g(x)$，求导后是个单增函数，二分求出零点即可
- 时间复杂度为 $O(n \cdot \log \epsilon^{-1})$

题目大意[4]

- Chihiro 有一棵有根树 T，一开始只有根节点 1

[4]Summer 2018, Yuhao Du Contest 5 A

题目大意[4]

- Chihiro 有一棵有根树 T，一开始只有根节点 1
- 接下来会有 m 次事件，Chihiro 会增加一个叶子节点，或者询问子树 u 的自同构数量

[4] Summer 2018, Yuhao Du Contest 5 A

题目大意[4]

- Chihiro 有一棵有根树 T，一开始只有根节点 1
- 接下来会有 m 次事件，Chihiro 会增加一个叶子节点，或者询问子树 u 的自同构数量
- 数据范围：$n, m \leq 10^5$

[4] Summer 2018, Yuhao Du Contest 5 A

解答

- 定义 Hash 函数 $f(u)$ 为 y_d（若 u 是叶子）或 $\prod(f(v)+x_d)$（u 不是叶子，v 是 u 的子节点），其中 d 为节点 u 的深度

解答

- 定义 Hash 函数 $f(u)$ 为 y_d（若 u 是叶子）或 $\prod(f(v)+x_d)$（u 不是叶子，v 是 u 的子节点），其中 d 为节点 u 的深度
- 注意到加入一个叶子 u 后，只会对轻边的父亲产生影响，至多 $O(\log n)$ 个

解答

- 定义 Hash 函数 $f(u)$ 为 y_d（若 u 是叶子）或 $\prod(f(v)+x_d)$（u 不是叶子，v 是 u 的子节点），其中 d 为节点 u 的深度
- 注意到加入一个叶子 u 后，只会对轻边的父亲产生影响，至多 $O(\log n)$ 个
- 考虑离线启发式合并，以时间为下标维护子树的 Hash 值

解答

- 定义 Hash 函数 $f(u)$ 为 y_d（若 u 是叶子）或 $\prod(f(v)+x_d)$（u 不是叶子，v 是 u 的子节点），其中 d 为节点 u 的深度
- 注意到加入一个叶子 u 后，只会对轻边的父亲产生影响，至多 $O(\log n)$ 个
- 考虑离线启发式合并，以时间为下标维护子树的 Hash 值
- 我们只需要继承重儿子的所有信息，按时间暴力遍历轻儿子的所有信息，打上标记即可，答案可以一并维护

解答

- 定义 Hash 函数 $f(u)$ 为 y_d（若 u 是叶子）或 $\prod(f(v)+x_d)$（u 不是叶子，v 是 u 的子节点），其中 d 为节点 u 的深度
- 注意到加入一个叶子 u 后，只会对轻边的父亲产生影响，至多 $O(\log n)$ 个
- 考虑离线启发式合并，以时间为下标维护子树的 Hash 值
- 我们只需要继承重儿子的所有信息，按时间暴力遍历轻儿子的所有信息，打上标记即可，答案可以一并维护
- 时间复杂度 $O(n\log^2 n)$

题目大意[5]

- 给出 n 和 k，求

$$\sum_{x=1}^{n} \operatorname{lcm}(x, x+1, \ldots, x+k)$$

[5] Winter 2020, Yuhao Du Contest 7 L

题目大意[5]

- 给出 n 和 k，求

$$\sum_{x=1}^{n} \operatorname{lcm}(x, x+1, \ldots, x+k)$$

- $n \leq 10^{11}, k \leq 30$

[5]Winter 2020, Yuhao Du Contest 7 L

解答

- 令 $L = \mathrm{lcm}(1, 2, \ldots, k)$，注意到当 $x \equiv r \pmod{L}$ 时

$$\frac{x(x+1)\cdots(x+k)}{\mathrm{lcm}(x, x+1, \ldots, x+k)}$$

是一个固定的整数 c_r

解答

- 令 $L = \text{lcm}(1, 2, \ldots, k)$，注意到当 $x \equiv r \pmod{L}$ 时
$$\frac{x(x+1)\cdots(x+k)}{\text{lcm}(x, x+1, \ldots, x+k)}$$
是一个固定的整数 c_r
- 因此 $\text{lcm}(x, x+1, \ldots, x+k) = \frac{1}{c_r}\sum_{i=1}^{k+1} a_i \cdot (Lt+r)^i$

解答

- 令 $L = \mathrm{lcm}(1, 2, \ldots, k)$，注意到当 $x \equiv r \pmod{L}$ 时

$$\frac{x(x+1)\cdots(x+k)}{\mathrm{lcm}(x, x+1, \ldots, x+k)}$$

  是一个固定的整数 c_r

- 因此 $\mathrm{lcm}(x, x+1, \ldots, x+k) = \frac{1}{c_r}\sum_{i=1}^{k+1} a_i \cdot (Lt+r)^i$
- 我们可以 $O(Lk)$ 求出答案

优化

- 设 $L = L_1 \times L_2$ 且 L_1, L_2 互质

优化

- 设 $L = L_1 \times L_2$ 且 L_1, L_2 互质
- 注意到 c_r 可表示为 $c'_{r \bmod L1} c''_{r \bmod L_2}$，启发我们运用 Meet in the Middle 技巧

优化

- 设 $L = L_1 \times L_2$ 且 L_1, L_2 互质
- 注意到 c_r 可表示为 $c'_{r \bmod L1} c''_{r \bmod L_2}$，启发我们运用 Meet in the Middle 技巧
- 考虑 CRT，设 $r \equiv r_1 M_1 + r_2 M_2 \pmod{L}$，根据 $r_1 M_1 + r_2 M_2$ 是否超过 L 以及 r 是否超过 $n \bmod L$，可以将 $r_1 M_1 + r_2 M_2$ 的不同贡献划分出四段区间

优化

- 设 $L = L_1 \times L_2$ 且 L_1, L_2 互质
- 注意到 c_r 可表示为 $c'_{r \bmod L1} c''_{r \bmod L_2}$，启发我们运用 Meet in the Middle 技巧
- 考虑 CRT，设 $r \equiv r_1 M_1 + r_2 M_2 \pmod{L}$，根据 $r_1 M_1 + r_2 M_2$ 是否超过 L 以及 r 是否超过 $n \bmod L$，可以将 $r_1 M_1 + r_2 M_2$ 的不同贡献划分出四段区间
- 可以枚举 $r_1 M_1$ 同时二分找到对应的 $r_2 M_2$ 区间，根据之前的公式计算答案即可

优化

- 设 $L = L_1 \times L_2$ 且 L_1, L_2 互质
- 注意到 c_r 可表示为 $c'_{r \bmod L1} c''_{r \bmod L_2}$，启发我们运用 Meet in the Middle 技巧
- 考虑 CRT，设 $r \equiv r_1 M_1 + r_2 M_2 \pmod{L}$，根据 $r_1 M_1 + r_2 M_2$ 是否超过 L 以及 r 是否超过 $n \bmod L$，可以将 $r_1 M_1 + r_2 M_2$ 的不同贡献划分出四段区间
- 可以枚举 $r_1 M_1$ 同时二分找到对应的 $r_2 M_2$ 区间，根据之前的公式计算答案即可
- 时间复杂度 $O(k^2 \sqrt{L})$

题目大意[6]

- Chihiro 有一张大小为 m 的 Hash 表

[6]Winter 2021, North American Contest 1 D (by jcvb)

题目大意[6]

- Chihiro 有一张大小为 m 的 Hash 表
- 接着 Chihiro 会依次插入 $x = 0, 1, \cdots, n-1$ 到位置 $x^2 \bmod m$，若当前位置已经有 T 个数，那么需要支付 T 的代价来插入 x

[6] Winter 2021, North American Contest 1 D (by jcvb)

题目大意[6]

- Chihiro 有一张大小为 m 的 Hash 表
- 接着 Chihiro 会依次插入 $x = 0, 1, \cdots, n-1$ 到位置 $x^2 \bmod m$，若当前位置已经有 T 个数，那么需要支付 T 的代价来插入 x
- 问总代价是多少

[6] Winter 2021, North American Contest 1 D (by jcvb)

题目大意[6]

- Chihiro 有一张大小为 m 的 Hash 表
- 接着 Chihiro 会依次插入 $x = 0, 1, \cdots, n-1$ 到位置 $x^2 \bmod m$，若当前位置已经有 T 个数，那么需要支付 T 的代价来插入 x
- 问总代价是多少
- $n, m \le 10^9$

[6]Winter 2021, North American Contest 1 D (by jcvb)

解答

- 注意到 $x^2 \equiv y^2 \pmod{m} \iff (x+y)(x-y) \equiv 0 \pmod{m}$，考虑枚举 $x+y, x-y$ 与 m 的公约数以求得答案

解答

- 注意到 $x^2 \equiv y^2 \pmod{m} \Longleftrightarrow (x+y)(x-y) \equiv 0 \pmod{m}$，考虑枚举 $x+y, x-y$ 与 m 的公约数以求得答案
- 记 $f(a,b)$ 为满足 $a|x+y, b|x-y, 0 \le x, y \le n-1$ 的二元组 (x,y) 的数量

解答

- 注意到 $x^2 \equiv y^2 \pmod{m} \iff (x+y)(x-y) \equiv 0 \pmod{m}$，考虑枚举 $x+y, x-y$ 与 m 的公约数以求得答案
- 记 $f(a,b)$ 为满足 $a|x+y, b|x-y, 0 \le x, y \le n-1$ 的二元组 (x,y) 的数量
- 若 $m = p^k$，此时枚举 $\gcd(m, x+y)$ 是 p 的几次幂，容易容斥得到答案为 $\sum_{0\le i\le k} f(p^i, p^{k-i}) - \sum_{1\le i\le k} f(p^i, p^{k+1-i})$

解答

- 注意到 $x^2 \equiv y^2 \pmod{m} \iff (x+y)(x-y) \equiv 0 \pmod{m}$，考虑枚举 $x+y, x-y$ 与 m 的公约数以求得答案
- 记 $f(a,b)$ 为满足 $a|x+y, b|x-y, 0 \le x, y \le n-1$ 的二元组 (x,y) 的数量
- 若 $m = p^k$，此时枚举 $\gcd(m, x+y)$ 是 p 的几次幂，容易容斥得到答案为 $\sum_{0 \le i \le k} f(p^i, p^{k-i}) - \sum_{1 \le i \le k} f(p^i, p^{k+1-i})$
- 对于一般情况 $m = \prod p_i^{k_i}$，枚举总量为 $\prod(2k_i+1) = d(m^2)$ 可以接受

计算 $f(a, b)$

- 令 $x+y=pa, x-y=qb$，那么

$$x=\frac{pa+qb}{2}, y=\frac{pa-qb}{2}$$

计算 $f(a,b)$

- 令 $x+y=pa, x-y=qb$，那么

$$x=\frac{pa+qb}{2}, y=\frac{pa-qb}{2}$$

- 结合约束 $0\leq x,y\leq n-1$，我们可以得到关于 p,q 的两个线性不等式，即求平面区域的整点数量，可以通过类欧算法解决

计算 $f(a, b)$

- 令 $x + y = pa, x - y = qb$，那么

$$x = \frac{pa + qb}{2}, y = \frac{pa - qb}{2}$$

- 结合约束 $0 \le x, y \le n - 1$，我们可以得到关于 p, q 的两个线性不等式，即求平面区域的整点数量，可以通过类欧算法解决
- 由于存在除二，我们还需要额外枚举 p, q 的奇偶性

计算 $f(a, b)$

- 令 $x + y = pa, x - y = qb$，那么

$$x = \frac{pa + qb}{2}, y = \frac{pa - qb}{2}$$

- 结合约束 $0 \le x, y \le n - 1$，我们可以得到关于 p, q 的两个线性不等式，即求平面区域的整点数量，可以通过类欧算法解决
- 由于存在除二，我们还需要额外枚举 p, q 的奇偶性
- 时间复杂度为 $O(d(m^2)\log m)$

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏

[7] XIX Open Cup, GP of ByteDance G

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏
- 有两个长度分别为 n, m 的序列 a, b，分别有一个指针 c, d，一开始均为 1

[7]XIX Open Cup, GP of ByteDance G

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏
- 有两个长度分别为 n, m 的序列 a, b，分别有一个指针 c, d，一开始均为 1
- 两人轮流游戏，Chihiro 先手，可行进行两种操作

[7] XIX Open Cup, GP of ByteDance G

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏
- 有两个长度分别为 n, m 的序列 a, b，分别有一个指针 c, d，一开始均为 1
- 两人轮流游戏，Chihiro 先手，可行进行两种操作
    1. 移动一个序列的指针，不能移动到原来的位置，且移动后的状态 (c, d) 不能在此前出现过。我们认为 $(1, 1)$ 在一开始已经出现过

[7] XIX Open Cup, GP of ByteDance G

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏
- 有两个长度分别为 n, m 的序列 a, b，分别有一个指针 c, d，一开始均为 1
- 两人轮流游戏，Chihiro 先手，可行进行两种操作
  1. 移动一个序列的指针，不能移动到原来的位置，且移动后的状态 (c, d) 不能在此前出现过。我们认为 $(1, 1)$ 在一开始已经出现过
  2. 结束游戏，分数为 $a_c + b_d$

[7] XIX Open Cup, GP of ByteDance G

题目大意[7]

- Chihiro 和 Kohakunushi 在玩一个游戏
- 有两个长度分别为 n, m 的序列 a, b，分别有一个指针 c, d，一开始均为 1
- 两人轮流游戏，Chihiro 先手，可行进行两种操作
  1. 移动一个序列的指针，不能移动到原来的位置，且移动后的状态 (c, d) 不能在此前出现过。我们认为 $(1, 1)$ 在一开始已经出现过
  2. 结束游戏，分数为 $a_c + b_d$
- Chihiro 会最小化游戏分数，而 Kohakunushi 会最大化它，求最终分数

[7] XIX Open Cup, GP of ByteDance G

问题转化

- 将两个数组排好序，此时初始位置不再是 $(1, 1)$

问题转化

- 将两个数组排好序，此时初始位置不再是 $(1, 1)$
- 二分答案，变成判定性问题，状态可以看做一个 $n \times m$ 的网格

问题转化

- 将两个数组排好序，此时初始位置不再是 $(1,1)$
- 二分答案，变成判定性问题，状态可以看做一个 $n \times m$ 的网格
- 游戏可以这样描述，每个格子有一个颜色，白色表示 Chihiro 赢，黑色表示 Kohakunushi 赢。这样问题就转化为经典的二分图博弈问题

问题转化

- 将两个数组排好序，此时初始位置不再是 $(1,1)$
- 二分答案，变成判定性问题，状态可以看做一个 $n \times m$ 的网格
- 游戏可以这样描述，每个格子有一个颜色，白色表示 Chihiro 赢，黑色表示 Kohakunushi 赢。这样问题就转化为经典的二分图博弈问题
- Chihiro 赢当且仅当其实起始位置在最大匹配上

问题转化

- 将两个数组排好序，此时初始位置不再是 $(1,1)$
- 二分答案，变成判定性问题，状态可以看做一个 $n \times m$ 的网格
- 游戏可以这样描述，每个格子有一个颜色，白色表示 Chihiro 赢，黑色表示 Kohakunushi 赢。这样问题就转化为经典的二分图博弈问题
- Chihiro 赢当且仅当其实起始位置在最大匹配上
- 即需要快速求出原图和删掉起始位置的最大匹配

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集
- 注意到白色和黑色一定会分别占据不相交的行和列

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集
- 注意到白色和黑色一定会分别占据不相交的行和列
- 同样，不难反证得到白色一定会占据列的前缀和行的前缀

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集
- 注意到白色和黑色一定会分别占据不相交的行和列
- 同样，不难反证得到白色一定会占据列的前缀和行的前缀
- 固定了行的前缀 r 后，可以确定最优的列的前缀 c

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集
- 注意到白色和黑色一定会分别占据不相交的行和列
- 同样，不难反证得到白色一定会占据列的前缀和行的前缀
- 固定了行的前缀 r 后，可以确定最优的列的前缀 c
- 再注意到 c 关于 r 单调，双指针即可

问题解决

- 最大匹配不太好求，注意到最大独立集等于点数减最大匹配，原问题等价于求最大独立集
- 注意到白色和黑色一定会分别占据不相交的行和列
- 同样，不难反证得到白色一定会占据列的前缀和行的前缀
- 固定了行的前缀 r 后，可以确定最优的列的前缀 c
- 再注意到 c 关于 r 单调，双指针即可
- 时间复杂度 $O(n\log A)$

谢谢大家!