组合数

Clonoth

2025.7

组合数

- 组合数：$\binom{n}{m}$ 表示从 n 个不同的物品中选出 m 个的方案数。

组合数

- 组合数：$\binom{n}{m}$ 表示从 n 个不同的物品中选出 m 个的方案数。
- 取补集：$\binom{n}{m} = \binom{n}{n-m}$。

组合数

- 组合数：$\binom{n}{m}$ 表示从 n 个不同的物品中选出 m 个的方案数。
- 取补集：$\binom{n}{m} = \binom{n}{n-m}$。
- 通项公式：$\binom{n}{m} = \dfrac{n!}{m!(n-m)!}$。

组合数

- 组合数：$\binom{n}{m}$ 表示从 n 个不同的物品中选出 m 个的方案数。
- 取补集：$\binom{n}{m} = \binom{n}{n-m}$。
- 通项公式：$\binom{n}{m} = \dfrac{n!}{m!(n-m)!}$。
- 递推式：$\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。

组合数

- 组合数：$\binom{n}{m}$ 表示从 n 个不同的物品中选出 m 个的方案数。
- 取补集：$\binom{n}{m} = \binom{n}{n-m}$。
- 通项公式：$\binom{n}{m} = \dfrac{n!}{m!(n-m)!}$。
- 递推式：$\binom{n}{m} = \binom{n-1}{m} + \binom{n-1}{m-1}$。
- 也是递推式：$\binom{n}{k} = \dfrac{n}{k}\binom{n-1}{k-1} = \dfrac{n-k+1}{k}\binom{n}{k-1}$。

多重组合数

- 多重组合数：$\binom{n}{s_1,\ldots,s_k}$ 表示从 n 个不同的物品中依次选出不交的 $s_1,\ldots,s_k$ 个物品的方案数，其中 $\sum_{i=1}^{k} s_i = n$。

多重组合数

- 多重组合数：$\binom{n}{s_1,\dots,s_k}$ 表示从 n 个不同的物品中依次选出不交的 $s_1,\dots,s_k$ 个物品的方案数，其中 $\sum\limits_{i=1}^{k} s_i = n$。
- $\binom{n}{s_1,\dots,s_k} = \dfrac{n!}{\prod\limits_{i=1}^{k} s_i!}$。

多重组合数

- 多重组合数：$\binom{n}{s_1,\dots,s_k}$ 表示从 n 个不同的物品中依次选出不交的 $s_1,\dots,s_k$ 个物品的方案数，其中 $\sum\limits_{i=1}^{k} s_i = n$。
- $\binom{n}{s_1,\dots,s_k} = \dfrac{n!}{\prod\limits_{i=1}^{k} s_i!}$。
- 容易发现 $\binom{n}{m} = \binom{n}{m, n-m}$。

性质

- 上指标求和：$\sum_{i=m}^{n}\binom{i}{m}=\binom{n+1}{m+1}$。

性质

- 上指标求和：$\sum_{i=m}^{n} \binom{i}{m} = \binom{n+1}{m+1}$。
- 证明考虑将 $\binom{n+1}{m+1}$ 按照杨辉三角递推式展开。

性质

- 上指标求和：$\sum_{i=m}^{n} \binom{i}{m} = \binom{n+1}{m+1}$。
- 证明考虑将 $\binom{n+1}{m+1}$ 按照杨辉三角递推式展开。
- 或者也可以考虑 $S = 1, 2, \ldots, n+1$ 的 $m+1$ 元子集数量。

性质

- 上指标求和：$\sum_{i=m}^{n}\binom{i}{m}=\binom{n+1}{m+1}$。
- 证明考虑将 $\binom{n+1}{m+1}$ 按照杨辉三角递推式展开。
- 或者也可以考虑 $S=1,2,\dots,n+1$ 的 $m+1$ 元子集数量。
- $\binom{n}{m}\binom{m}{k}=\binom{n}{k}\binom{n-k}{m-k}$。

性质

- 上指标求和：$\sum_{i=m}^{n}\binom{i}{m}=\binom{n+1}{m+1}$。
- 证明考虑将 $\binom{n+1}{m+1}$ 按照杨辉三角递推式展开。
- 或者也可以考虑 $S=1,2,\dots,n+1$ 的 $m+1$ 元子集数量。
- $\binom{n}{m}\binom{m}{k}=\binom{n}{k}\binom{n-k}{m-k}$。
- 考虑组合意义即可。

二项式定理

- $(a+b)^n=\sum_{i=0}^{n}\binom{n}{i}a^{n-i}b^i$。

二项式定理

- $(a+b)^n = \sum_{i=0}^{n} \binom{n}{i} a^{n-i} b^i$。
- 证明可以考虑组合意义或数学归纳法。

二项式定理

- $(a+b)^n = \sum_{i=0}^{n} \binom{n}{i} a^{n-i} b^i$。
- 证明可以考虑组合意义或数学归纳法。
- 一些推论:

二项式定理

- $(a+b)^n=\sum_{i=0}^{n}\binom{n}{i}a^{n-i}b^i$。
- 证明可以考虑组合意义或数学归纳法。
- 一些推论:
  - $\sum_{i=0}^{n}\binom{n}{i}=(1+1)^n=2^n$。

二项式定理

- $(a+b)^n = \sum\limits_{i=0}^{n} \binom{n}{i} a^{n-i} b^i$。
- 证明可以考虑组合意义或数学归纳法。
- 一些推论：
  - $\sum\limits_{i=0}^{n} \binom{n}{i} = (1+1)^n = 2^n$。
  - $\sum\limits_{i=0}^{n} (-1)^i \binom{n}{i} = (1+(-1))^n = [n=0]$。

二项式定理

- $(a+b)^n=\sum_{i=0}^{n}\binom{n}{i}a^{n-i}b^i$。
- 证明可以考虑组合意义或数学归纳法。
- 一些推论：
  - $\sum_{i=0}^{n}\binom{n}{i}=(1+1)^n=2^n$。
  - $\sum_{i=0}^{n}(-1)^i\binom{n}{i}=(1+(-1))^n=[n=0]$。
  - 上式指出对于 $n\geq 1$，从 $S=\{1,\ldots,n\}$ 选出大小为奇数的子集的方案数与选出大小为偶数的子集的方案数相同。

AT_tenka1_2014_final_d 高橋君 题意

m 次询问，每次给定 n, k，求 $\sum_{i=0}^{k} \binom{n}{i}$。

$n, m, k \leq 10^5$。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。
- $(n,k)\to(n,k+1),(n,k-1)$ 是显然的，只需要考虑 $(n,k)\to(n+1,k),(n-1,k)$。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。
- $(n,k)\to(n,k+1),(n,k-1)$ 是显然的，只需要考虑 $(n,k)\to(n+1,k),(n-1,k)$。
- 考虑杨辉三角递推式，$\sum_{i=0}^{k}\binom{n+1}{i}=\sum_{i=0}^{k}\binom{n}{i}+\binom{n}{i-1}=2\sum_{i=0}^{k}\binom{n}{i}-\binom{n}{k}$。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。
- $(n,k)\to(n,k+1),(n,k-1)$ 是显然的，只需要考虑 $(n,k)\to(n+1,k),(n-1,k)$。
- 考虑杨辉三角递推式，$\sum\limits_{i=0}^{k}\binom{n+1}{i}=\sum\limits_{i=0}^{k}\binom{n}{i}+\binom{n}{i-1}=2\sum\limits_{i=0}^{k}\binom{n}{i}-\binom{n}{k}$。
- 代换并反解上述等式可以得到 $\sum\limits_{i=0}^{k}\binom{n-1}{i}=\dfrac{\sum\limits_{i=0}^{k}\binom{n}{i}+\binom{n}{k}}{2}$。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。
- $(n,k)\to(n,k+1),(n,k-1)$ 是显然的，只需要考虑 $(n,k)\to(n+1,k),(n-1,k)$。
- 考虑杨辉三角递推式，$\sum_{i=0}^{k}\binom{n+1}{i}=\sum_{i=0}^{k}\binom{n}{i}+\binom{n}{i-1}=2\sum_{i=0}^{k}\binom{n}{i}-\binom{n}{k}$。
- 代换并反解上述等式可以得到 $\sum_{i=0}^{k}\binom{n-1}{i}=\frac{\sum_{i=0}^{k}\binom{n}{i}+\binom{n}{k}}{2}$。
- 时间复杂度 $O(n\sqrt{m})$。

AT_tenka1_2014_final_d 高橋君 题解

- 双端点询问考虑莫队，问题转化为如何移动端点。
- $(n,k)\to(n,k+1),(n,k-1)$ 是显然的，只需要考虑 $(n,k)\to(n+1,k),(n-1,k)$。
- 考虑杨辉三角递推式，$\sum_{i=0}^{k}\binom{n+1}{i}=\sum_{i=0}^{k}\binom{n}{i}+\binom{n}{i-1}=2\sum_{i=0}^{k}\binom{n}{i}-\binom{n}{k}$。
- 代换并反解上述等式可以得到 $\sum_{i=0}^{k}\binom{n-1}{i}=\frac{\sum_{i=0}^{k}\binom{n}{i}+\binom{n}{k}}{2}$。
- 时间复杂度 $O(n\sqrt{m})$。
- 存在 $O(n\log^2 n)$ 做法，其中 n,m 同阶。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。
- 一些推论；

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。
- 一些推论：
  - $\sum_{i=-r}^{s}\binom{n}{r+i}\binom{m}{s-i}=\binom{n+m}{s+r}$。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。
- 一些推论：
  - $\sum_{i=-r}^{s}\binom{n}{r+i}\binom{m}{s-i}=\binom{n+m}{s+r}$。
  - $\sum_{i=1}^{n}\binom{n}{i}\binom{n}{i-1}=\sum_{i=1}^{n}\binom{n}{n-i}\binom{n}{i-1}=\binom{2n}{n-1}$。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。
- 一些推论：
  - $\sum_{i=-r}^{s}\binom{n}{r+i}\binom{m}{s-i}=\binom{n+m}{s+r}$。
  - $\sum_{i=1}^{n}\binom{n}{i}\binom{n}{i-1}=\sum_{i=1}^{n}\binom{n}{n-i}\binom{n}{i-1}=\binom{2n}{n-1}$。
  - $\sum_{i=0}^{n}\binom{n}{i}^2=\sum_{i=0}^{n}\binom{n}{i}\binom{n}{n-i}=\binom{2n}{n}$。

范德蒙德卷积

- $\sum_{i=0}^{k}\binom{n}{i}\binom{m}{k-i}=\binom{n+m}{k}$。
- 证明考虑组合意义。
- 一些推论：
  - $\sum_{i=-r}^{s}\binom{n}{r+i}\binom{m}{s-i}=\binom{n+m}{s+r}$。
  - $\sum_{i=1}^{n}\binom{n}{i}\binom{n}{i-1}=\sum_{i=1}^{n}\binom{n}{n-i}\binom{n}{i-1}=\binom{2n}{n-1}$。
  - $\sum_{i=0}^{n}\binom{n}{i}^2=\sum_{i=0}^{n}\binom{n}{i}\binom{n}{n-i}=\binom{2n}{n}$。
  - $\sum_{i=0}^{m}\binom{n}{i}\binom{m}{i}=\sum_{i=0}^{m}\binom{n}{i}\binom{m}{m-i}=\binom{n+m}{m}$。

CF785D Anton and School - 2 题意

给定一个长度为 n 的括号序列，你需要求出满足以下条件的子序列的数量。

- 该子序列长度为偶数。
- 设该子序列长度为 $2m$，则该子序列前 m 个字符均为 (，后 m 个字符均为)。

对 10^9+7 取模。

$n \le 2\times 10^5$。

CF785D Anton and School - 2 题解

- 考虑枚举子序列中最后一个 (的位置。

CF785D Anton and School - 2 题解

- 考虑枚举子序列中最后一个 (的位置。
- 对于原串中的每个 (，设其前面有 a 个 (，其后面有 b 个)，那么以其为最后一个 (的子序列个数为 $\sum_{i=0}^{a}\binom{a}{i}\binom{b}{i+1}=\sum_{i=0}^{a}\binom{a}{a-i}\binom{b}{i+1}=\binom{a+b}{a+1}$。

CF785D Anton and School - 2 题解

- 考虑枚举子序列中最后一个 (的位置。
- 对于原串中的每个 (，设其前面有 a 个 (，其后面有 b 个)，那么以其为最后一个 (的子序列个数为 $\sum_{i=0}^{a}\binom{a}{i}\binom{b}{i+1}=\sum_{i=0}^{a}\binom{a}{a-i}\binom{b}{i+1}=\binom{a+b}{a+1}$。
- 时间复杂度 $O(n)$。

- 有 k 个非负整数变量 $x_1, \ldots, x_k$ 要求 $x_1 + \cdots + x_k = n$。

- 有 k 个非负整数变量 $x_1, \ldots, x_k$ 要求 $x_1 + \cdots + x_k = n$。
- 解的数量 $\binom{n+k-1}{n}$。

- 有 k 个非负整数变量 $x_1, \ldots, x_k$ 要求 $x_1 + \cdots + x_k = n$。
- 解的数量 $\binom{n+k-1}{n}$。
- 若每个整数变量有下界 $x_i \ge l_i$，可令 $n' = n - l_1 - \cdots - l_k$，解的数量为 $\binom{n'+k-1}{n'}$。

- 有 k 个非负整数变量 $x_1, \ldots, x_k$ 要求 $x_1 + \cdots + x_k = n$。
- 解的数量 $\binom{n+k-1}{n}$。
- 若每个整数变量有下界 $x_i \geq l_i$，可令 $n' = n - l_1 - \cdots - l_k$，解的数量为 $\binom{n'+k-1}{n'}$。
- 所以不失一般性，假设所有整数解问题中变量的下界都是 0。

整数解问题

- 若每个变量有上界 $x_i \le r_i$，则可以通过容斥计算。

整数解问题

- 若每个变量有上界 $x_i \le r_i$，则可以通过容斥计算。
- 对于集合 $S \subseteq \{1, 2, \ldots, k\}$，钦定 $\forall i \in S, x_i > r_i$，此时令 $n' = n - \sum\limits_{i \in S} (r_i + 1)$，计算 $x_1 + \cdots + x_k = n'$ 的解的数量即可。注意容斥系数。

AT_jsc2019_qual_f Candy Retribution 题意

给定正整数 n, m, l, r 求满足以下条件的非负整数序列 $x_1, \dots, x_n$ 的数量。

- $l \le \sum_{i=1}^{n} x_i \le r$。
- 在将序列 $x_1, \dots, x_n$ 降序排列后得到 $b_1, \dots, b_n$，要求 $b_m = b_{m+1}$。

$1 \le m < n \le 3 \times 10^5, 1 \le l \le r \le 3 \times 10^5$

AT_jsc2019_qual_f Candy Retribution 题解

- 首先答案为 $\sum_{i=1}^{n} x_i \le r$ 的方案数减去 $\sum_{i=1}^{n} x_i \le l-1$ 的方案数。所以条件一转化为 $\sum_{i=1}^{n} x_i \le r$

AT_jsc2019_qual_f Candy Retribution 题解

- 首先答案为 $\sum_{i=1}^{n} x_i \le r$ 的方案数减去 $\sum_{i=1}^{n} x_i \le l-1$ 的方案数。所以条件一转化为 $\sum_{i=1}^{n} x_i \le r$
- 考虑计算 $b_m \neq b_{m+1}$ 的方案数。

AT_jsc2019_qual_f Candy Retribution 题解

- 首先答案为 $\sum\limits_{i=1}^{n} x_i \le r$ 的方案数减去 $\sum\limits_{i=1}^{n} x_i \le l-1$ 的方案数。所以条件一转化为 $\sum\limits_{i=1}^{n} x_i \le r$
- 考虑计算 $b_m \ne b_{m+1}$ 的方案数。
- 枚举 $b_m = k$，此时的限制为有 m 个变量要求 $x_i \ge k$ 其余 $n-m$ 个变量要求 $x_i < k$ 且存在 $x_i = k$，同时 $\sum\limits_{i=1}^{n} x_i \le r$。

AT_jsc2019_qual_f Candy Retribution 题解

- 首先答案为 $\sum_{i=1}^{n} x_i \le r$ 的方案数减去 $\sum_{i=1}^{n} x_i \le l-1$ 的方案数。所以条件一转化为 $\sum_{i=1}^{n} x_i \le r$
- 考虑计算 $b_m \ne b_{m+1}$ 的方案数。
- 枚举 $b_m = k$，此时的限制为有 m 个变量要求 $x_i \ge k$ 其余 $n-m$ 个变量要求 $x_i < k$ 且存在 $x_i = k$，同时 $\sum_{i=1}^{n} x_i \le r$。
- 对于存在 $x_i = k$ 的限制只需要用 m 个变量要求 $x_i \ge k$ 减去 m 个变量要求 $x_i \ge k+1$ 的方案数即可。

AT_jsc2019_qual_f Candy Retribution 题解

- 首先答案为 $\sum\limits_{i=1}^{n} x_i \le r$ 的方案数减去 $\sum\limits_{i=1}^{n} x_i \le l-1$ 的方案数。所以条件一转化为 $\sum\limits_{i=1}^{n} x_i \le r$
- 考虑计算 $b_m \neq b_{m+1}$ 的方案数。
- 枚举 $b_m = k$，此时的限制为有 m 个变量要求 $x_i \ge k$ 其余 $n-m$ 个变量要求 $x_i < k$ 且存在 $x_i = k$，同时 $\sum\limits_{i=1}^{n} x_i \le r$。
- 对于存在 $x_i = k$ 的限制只需要用 m 个变量要求 $x_i \ge k$ 减去 m 个变量要求 $x_i \ge k+1$ 的方案数即可。
- 注意到此时对变量和的限制为 $\le r$，此时只需要新建一个没有限制的非负整数变量 x_{n+1} 即可。

AT_jsc2019_qual_f Candy Retribution 题解

- 到此为止问题转化为了上文中的变量和问题，只需要枚举 j 并钦定 $n-m$ 个变量有 j 个变量 $>m$ 即可。

AT_jsc2019_qual_f Candy Retribution 题解

- 到此为止问题转化为了上文中的变量和问题，只需要枚举 j 并钦定 $n-m$ 个变量有 j 个变量 $> m$ 即可。
- 注意到枚举的 j 只有 $O(\frac{r}{i})$ 个，同时 $O(\sum_{i=1}^{r}\frac{r}{i}) = O(r\log r)$，所以总复杂度为 $O(n+r\log r)$。

反射容斥

- 反射容斥用于处理一类不碰到直线 $y = x + b$ 的格路计数问题。

反射容斥

- 反射容斥用于处理一类不碰到直线 $y = x + b$ 的格路计数问题。
- 卡特兰数：从 $(0,0) \to (n,n)$ 不碰到 $y = x + 1$ 的方案数。

反射容斥

- 反射容斥用于处理一类不碰到直线 $y=x+b$ 的格路计数问题。
- 卡特兰数：从 $(0,0)\to(n,n)$ 不碰到 $y=x+1$ 的方案数。
- 从 $(0,0)$ 到 (n,n) 没有限制的方案数为 $\binom{2n}{n}$。

反射容斥

- 反射容斥用于处理一类不碰到直线 $y = x + b$ 的格路计数问题。
- 卡特兰数：从 $(0,0) \to (n,n)$ 不碰到 $y = x + 1$ 的方案数。
- 从 $(0,0)$ 到 (n,n) 没有限制的方案数为 $\binom{2n}{n}$。
- 对于每一条碰到了 $y = x + 1$ 的路径，考虑将其从 $(0,0)$ 至第一次碰到的位置作关于 $y = x + 1$ 的对称，变为从 $(-1,1) \to (n,n)$ 的路径。

反射容斥

- 反射容斥用于处理一类不碰到直线 $y = x + b$ 的格路计数问题。
- 卡特兰数：从 $(0,0) \to (n,n)$ 不碰到 $y = x + 1$ 的方案数。
- 从 $(0,0)$ 到 (n,n) 没有限制的方案数为 $\binom{2n}{n}$。
- 对于每一条碰到了 $y = x + 1$ 的路径，考虑将其从 $(0,0)$ 至第一次碰到的位置作关于 $y = x + 1$ 的对称，变为从 $(-1,1) \to (n,n)$ 的路径。
- 容易发现 $(-1,1) \to (n,n)$ 的路径与 $(0,0) \to (n,n)$ 碰到 $y = x + 1$ 的路径构成双射，所以原问题为 $\binom{2n}{n} - \binom{2n}{n-1} = \binom{2n}{n} - \frac{n}{n+1}\binom{2n}{n} = \frac{\binom{2n}{n}}{n+1}$。

反射容斥

- 推广：计算 $(0,0) \to (n,m)$ 不碰到 $L: y = x + l$ 和 $R: y = x + r$ 的方案数，其中 $l < 0 < r$。

反射容斥

- 推广：计算 $(0,0) \to (n,m)$ 不碰到 $L: y = x + l$ 和 $R: y = x + r$ 的方案数，其中 $l < 0 < r$。
- 对于路径定义由 L, R 构成的字符串 $s_1, \ldots, s_k$ 表示依次碰到 $s_1, \ldots, s_k$，且 $s_i \neq s_{i=1}$，即连续两次碰到同一条线只算一次。

反射容斥

- 推广：计算 $(0,0) \to (n,m)$ 不碰到 $L: y = x + l$ 和 $R: y = x + r$ 的方案数，其中 $l < 0 < r$。
- 对于路径定义由 L, R 构成的字符串 $s_1, \dots, s_k$ 表示依次碰到 $s_1, \dots, s_k$，且 $s_i \neq s_{i=1}$，即连续两次碰到同一条线只算一次。
- 对于由 L, R 构成的字符串 $S = s_1, \dots, s_k$，定义 $f(S)$ 表示由路径定义的字符串以 S 作为子串的路径数量。

反射容斥

- 推广：计算 $(0,0) \to (n,m)$ 不碰到 $L: y = x + l$ 和 $R: y = x + r$ 的方案数，其中 $l < 0 < r$。
- 对于路径定义由 L, R 构成的字符串 $s_1, \dots, s_k$ 表示依次碰到 $s_1, \dots, s_k$，且 $s_i \neq s_{i=1}$，即连续两次碰到同一条线只算一次。
- 对于由 L, R 构成的字符串 $S = s_1, \dots, s_k$，定义 $f(S)$ 表示由路径定义的字符串以 S 作为子串的路径数量。
- 那么原问题的答案为 $f(\varnothing) - f(L) - f(R) + f(LR) + f(RL) - f(LRL) - f(RLR) + f(LRLR) + f(RLRL) - \cdots$。

反射容斥

- 在计算 $f(S)$ 时只需要将 $(0,0)$ 依次按照 $s_1,\dots,s_k$ 对称即可，设最后得到的点为 (x,y)，计算 $(x,y)\to(n,m)$ 的方案数即可。

反射容斥

- 在计算 $f(S)$ 时只需要将 $(0,0)$ 依次按照 $s_1,\dots,s_k$ 对称即可，设最后得到的点为 (x,y)，计算 $(x,y)\to(n,m)$ 的方案数即可。
- 注意到不为 0 的 $f(S)$ 数量为 $O(\frac{n+m}{r-l})$。

UOJ424. 【集训队作业2018】count 题意

如果一个序列满足序列长度为 n，序列中的每个数都是 1 到 m 内的整数，且所有 1 到 m 内的整数都在序列中出现过，则称这是一个挺好序列。

对于一个序列 A，记 $f_A(l,r)$ 为 A 的第 l 个到第 r 个数中最大值的下标（如果有多个最大值，取下标最小的）。

两个序列 A 和 B 同构，当且仅当 A 和 B 长度相等，且对于任意 $i \le j$，均有 $f_A(i,j) = f_B(i,j)$。

给出 n,m，求有多少种不同构的挺好序列。答案对 998244353 取模。

$n,m \le 10^5$。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。
- 注意到两个序列同构等价于它们的大根笛卡尔树相同（笛卡尔树中左儿子严格小于当前结点，右儿子小于等于当前结点）。所以问题转化为有多少二叉树可以作为某个挺好序列的大根笛卡尔树。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。
- 注意到两个序列同构等价于它们的大根笛卡尔树相同（笛卡尔树中左儿子严格小于当前结点，右儿子小于等于当前结点）。所以问题转化为有多少二叉树可以作为某个挺好序列的大根笛卡尔树。
- 考虑二叉树与合法括号序列的双射：(左子树)右子树。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。
- 注意到两个序列同构等价于它们的大根笛卡尔树相同（笛卡尔树中左儿子严格小于当前结点，右儿子小于等于当前结点）。所以问题转化为有多少二叉树可以作为某个挺好序列的大根笛卡尔树。
- 考虑二叉树与合法括号序列的双射：(左子树)右子树。
- 注意到一个二叉树可以作为某个挺好序列的大根笛卡尔树当且仅达它对应合法括号序列的嵌套深度不超过 m。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。
- 注意到两个序列同构等价于它们的大根笛卡尔树相同（笛卡尔树中左儿子严格小于当前结点，右儿子小于等于当前结点）。所以问题转化为有多少二叉树可以作为某个挺好序列的大根笛卡尔树。
- 考虑二叉树与合法括号序列的双射：(左子树)右子树。
- 注意到一个二叉树可以作为某个挺好序列的大根笛卡尔树当且仅达它对应合法括号序列的嵌套深度不超过 m。
- 而这对应一条 $(0,0) \to (n,n)$ 的不碰到 $y = x + 1$ 和 $y = x - m - 1$ 的路径。

UOJ424. 【集训队作业2018】count 题解

- 首先若 $m > n$，答案显然为 0。
- 注意到两个序列同构等价于它们的大根笛卡尔树相同（笛卡尔树中左儿子严格小于当前结点，右儿子小于等于当前结点）。所以问题转化为有多少二叉树可以作为某个挺好序列的大根笛卡尔树。
- 考虑二叉树与合法括号序列的双射：(左子树)右子树。
- 注意到一个二叉树可以作为某个挺好序列的大根笛卡尔树当且仅达它对应合法括号序列的嵌套深度不超过 m。
- 而这对应一条 $(0,0) \to (n,n)$ 的不碰到 $y = x + 1$ 和 $y = x - m - 1$ 的路径。
- 时间复杂度 $O(n)$。

Lucas 定理

- Lucas 定理：对于素数 p，有 $\binom{n}{m} \equiv \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor}\binom{n \bmod p}{m \bmod p} \pmod{p}$。

Lucas 定理

- Lucas 定理：对于素数 p，有 $\binom{n}{m} \equiv \binom{\lfloor n/p \rfloor}{\lfloor m/p \rfloor}\binom{n \bmod p}{m \bmod p} \pmod p$。
- Lucas 定理指出，可以 $O(p)$ 预处理，$O(\log_p n)$ 求解 $\binom{n}{m} \bmod p$。

exLucas 算法

- exLucas 可以求解一般的 $\binom{n}{m} \bmod p$ 的值，而不要求 p 是素数。

exLucas 算法

- exLucas 可以求解一般的 $\binom{n}{m} \bmod p$ 的值，而不要求 p 是素数。
- 设 p 的质因子分解为 $p = p_1^{\alpha_1} \dots p_r^{\alpha_r}$，如果我们求出了 $\binom{n}{m} \bmod p_i^{\alpha_i}$ 的值，则可以通过 CRT 进行合并。

exLucas 算法

- exLucas 可以求解一般的 $\binom{n}{m} \bmod p$ 的值，而不要求 p 是素数。
- 设 p 的质因子分解为 $p = p_1^{\alpha_1} \dots p_r^{\alpha_r}$，如果我们求出了 $\binom{n}{m} \bmod p_i^{\alpha_i}$ 的值，则可以通过 CRT 进行合并。
- 问题转化为模数为 p^k 的情形，其中 p 为素数。

exLucas 算法

- 对于素数 p 和正整数 n，将 $n!$ 中所有的因子 p 都提取出来，可以得到分解 $n! = p^{\nu_p(n!)}(n!)_p$，其中 $p \nmid (n!)_p$。

exLucas 算法

- 对于素数 p 和正整数 n，将 $n!$ 中所有的因子 p 都提取出来，可以得到分解 $n! = p^{\nu_p(n!)}(n!)_p$，其中 $p \nmid (n!)_p$。
- 由于 $\binom{n}{m} = \dfrac{n!}{m!(n-m)!}$，考虑分别求出 $\nu_p(n!), \nu_p(m!), \nu_p((n-m)!), (n!)_p, (m!)_p, ((n-m))!_p$，由于 $(n!)_p, (m!)_p, ((n-m)!)_p$ 均与 p^k 互质，所有可以作除法。

exLucas 算法

- 对于素数 p 和正整数 n，将 $n!$ 中所有的因子 p 都提取出来，可以得到分解 $n! = p^{\nu_p(n!)}(n!)_p$，其中 $p \nmid (n!)_p$。
- 由于 $\binom{n}{m} = \frac{n!}{m!(n-m)!}$，考虑分别求出 $\nu_p(n!), \nu_p(m!), \nu_p((n-m)!), (n!)_p, (m!)_p, ((n-m))!_p$，由于 $(n!)_p, (m!)_p, ((n-m)!)_p$ 均与 p^k 互质，所有可以作除法。
- 问题转化为求解 $\nu_p(n!), (n!)_p \bmod p^k$。

exLucas 算法

- 注意到 $\nu_p(n!) = \sum_{i=1}^{n} \nu_p(i) = \sum_{j \geq 1} \sum_{i=1}^{n} [\nu_p(i) \geq j] = \sum_{j \geq 1} \lfloor n/p^j \rfloor$，可以 $O(\log_p n)$ 计算。

exLucas 算法

- 注意到 $\nu_p(n!) = \sum_{i=1}^{n} \nu_p(i) = \sum_{j\geq 1}\sum_{i=1}^{n}[\nu_p(i) \geq j] = \sum_{j\geq 1}\left\lfloor n/p^j \right\rfloor$，可以 $O(\log_p n)$ 计算。
- 对于 $(n!)_p \bmod p^k$，考虑提取所有 p 的倍数并合并 $\bmod p^k$ 的循环节 $(n!)_p \equiv \prod_{1\leq i\leq n} i_p \equiv \prod_{1\leq i\leq n, p\mid i} i_p \prod_{1\leq i\leq n, p\nmid i} i_p \equiv$ $(\lfloor n/p \rfloor !)_p \prod_{1\leq i\leq n, p\nmid i} i \equiv$ $(\lfloor n/p \rfloor !)_p \left(\prod_{1\leq i< p^k, p\nmid i} i\right)^{\left\lfloor n/p^k \right\rfloor} \prod_{1\leq i\leq n \bmod p^k, p\nmid i} i \pmod{p^k}$。

exLucas 算法

- 注意到 $\nu_p(n!) = \sum_{i=1}^{n} \nu_p(i) = \sum_{j\geq 1}\sum_{i=1}^{n}[\nu_p(i) \geq j] = \sum_{j\geq 1} \lfloor n/p^j \rfloor$，可以 $O(\log_p n)$ 计算。
- 对于 $(n!)_p \bmod p^k$，考虑提取所有 p 的倍数并合并 $\bmod p^k$ 的循环节 $(n!)_p \equiv \prod_{1\leq i\leq n} i_p \equiv \prod_{1\leq i\leq n, p\mid i} i_p \prod_{1\leq i\leq n, p\nmid i} i_p \equiv (\lfloor n/p \rfloor !)_p \prod_{1\leq i\leq n, p\nmid i} i \equiv$ $(\lfloor n/p \rfloor !)_p \left(\prod_{1\leq i<p^k, p\nmid i} i\right)^{\lfloor n/p^k \rfloor} \prod_{1\leq i\leq n \bmod p^k, p\nmid i} i \pmod{p^k}$。
- 在 $O(p^k)$ 预处理 $\forall 1 \leq s < p^k, \prod_{1\leq i\leq s \bmod p^k, p\nmid i} i$ 后，可以 $O(\log_p n)$ 递归计算。

exLucas 算法

- 设 p 的质因子分解为 $p = p_1^{\alpha_1} \dots p_r^{\alpha_r}$，总时间复杂度为 $O(\sum\limits_{i=1}^{r} p_i^{\alpha_i})$ 预处理，$O(\sum\limits_{i=1}^{r} \log_{p_i} n)$ 处理单次询问。

exLucas 算法

- 设 p 的质因子分解为 $p = p_1^{\alpha_1} \dots p_r^{\alpha_r}$，总时间复杂度为 $O(\sum_{i=1}^{r} p_i^{\alpha_i})$ 预处理，$O(\sum_{i=1}^{r} \log_{p_i} n)$ 处理单次询问。
- 小优化：$\prod_{1 \le i < p^k, p \nmid i} i \equiv \pm 1 \pmod{p^k}$。

exLucas 算法

- 设 p 的质因子分解为 $p = p_1^{\alpha_1} \dots p_r^{\alpha_r}$，总时间复杂度为 $O(\sum\limits_{i=1}^{r} p_i^{\alpha_i})$ 预处理，$O(\sum\limits_{i=1}^{r} \log_{p_i} n)$ 处理单次询问。
- 小优化：$\prod\limits_{1 \le i < p^k, p \nmid i} i \equiv \pm 1 \pmod{p^k}$。
- Bonus：存在预处理时间复杂度为 $O(\sum\limits_{i=1}^{r} \operatorname{poly}(p_i, \alpha_i))$ 的算法。

习题

P7386 「EZEC-6」0-1 Trie
QOJ 4903. 细菌
[ARC068F] Solitaire
[AGC019F] Yes or No
AT_keyence2019_f Paper Cutting
CF1835E Old Mobile