

京东 E 卡

如果每种物品都有无数种的话，那么直接用插板法可得方案数为 $\binom{s+n-1}{n-1}$ 。

但是问题在于每种元素个数都有一个上界，这样会得到一些不合法的方案。考虑将其减去。

令 $f(S)$ 表示集合 S 中的物品集合都超出上界的方案。那么有

$$f(S) = \binom{s+n-1-\sum_{i \in S}(d_i+1)}{n-1}$$

令 g_S 表示恰好是集合 S 里的元素超上界的方案，那么由定义得

$$f(S) = \sum_{T \subseteq S} g(T)$$

由容斥原理得答案

$$g(\emptyset) = \sum_T (-1)^{|T|} f(T)$$

删删乐

解决这道题需要三个重要的观察：

1. $\sum a_i = n - 1$ ，即 a_i 的和为边数，所以 $\gcd(a_1, a_2, \dots, a_n)$ 是 $n - 1$ 的约数。
2. 对于所有叶子节点，它们只和父亲相连，那么 $a \in \{0, 1\}$ 。
3. 每条边要么被计入儿子的 a 要么被计入父亲的 a ，不同的序列 a 共有 2^{n-1}

于是所有叶子节点都不能比父亲先删，否则 $\gcd = 1$ 。

现在考虑某个叶子节点 v 的父亲 u ， u 只剩一条指向 $f a_u$ 的边可以影响 a_u 。

令 s_u 表示 u 所连叶子节点个数，那么 $a_u \in \{s_u, s_u + 1\}$ ，于是 \gcd 只会是 s_u 或者 $s_u + 1$ 的约数。

那么我们得到了一个检测 \gcd 是否可能是 $k(k \geq 2)$ 的算法。从叶子节点开始不断给边定归属，从底向上执行这样的操作。由于每个点的 a 都只可能有两种取值，而 $k \geq 2$ ，于是每条边的归属是唯一的。

对每个 $k \geq 2$ 且 $k | n - 1$ ， $O(n)$ 检测是否能使 $\gcd = k$ ， $\gcd = 1$ 的答案就是 2^{n-1} 减去其余答案。

数学专题签到处

令

$$f_x = \sum_{i=1}^N \binom{3i}{x}$$

加法公式展开三次，得到

$$\begin{aligned} f_x &= \sum_{i=1}^N \binom{3i-3}{x-3} + 3 \binom{3i-3}{x-2} + 3 \binom{3i-3}{x-1} + \binom{3i-3}{x} \\ &= f_{x-3} - \binom{3N}{x-3} + 3 \left(f_{x-2} - \binom{3N}{x-2} \right) + 3 \left(f_{x-1} - \binom{3N}{x-1} \right) + f_x - \binom{3N}{x} \end{aligned}$$

f_x 被消掉了，得到了一个二阶线性递推， $O(N)$ 预处理， $O(1)$ 查询。

树上序列

不妨先考虑 dp，令 $f_{u,j}$ 表示只考虑 u 的子树，并且 $a_u = j$ 时的方案数，那么

$$f_{u,j} = \prod_{v \in \text{son}(u)} \sum_{k=1}^j f_{v,k}$$

令 $g_{u,j}$ 为 $f_{u,j}$ 关于 j 的前缀和，即

$$g_{u,j} = \sum_{k=1}^j f_{u,k}$$

对于叶子节点来说，有 $g_{u,j} = 1$ 。

对于非叶子节点，有

$$\Delta g_{u,j} = g_{u,j} - g_{u,j-1} = f_{u,j} = \prod_{v \in \text{Son}(u)} g_{v,j}$$

换一种记号，即

$$\Delta g_u(x) = g_u(x) - g_u(x-1) = f_u(x) = \prod_{v \in \text{Son}(u)} g_v(x)$$

对于叶子节点， $g_u(x) = 1$ 为一个零次多项式。

对于非叶子节点，假设 $g_v(x)$ 为关于 x 的 k_v 次多项式，那么 $\Delta g_u(x)$ 为 $\sum_{v \in \text{Son}(u)} k_v$ 次多项式，所以 $g_u(x)$ 至多是一个 $k_u = 1 + \sum_{v \in \text{Son}(u)} k_v$ 次多项式。根据这个式子，不断展开 k_v ，可以说 $k_u \leq sz_u$ ，其中 sz_u 为 u 的子树大小。

那么我们要的答案 $g_1(V)$ ，是一个至多 n 次的多项式 $g_1(x)$ 在 $x = V$ 处的点值。

我们可以 dp 求出 $g_1(x)$ 在 $x = 1, 2, \dots, n+1$ 处的点值，然后用拉格朗日插值求出多项式 $g_1(x)$ 的系数，最后代入 $x = V$ 即可。复杂度 $O(n^2)$ 。

矩阵翻转

考虑对所求列出式子，将每一列状压成一个数 a_i ，枚举每行翻不翻转，可以得到答案

$$\min_{x \in [0, 2^n)} \sum_{i=1}^m f(a_i \oplus x)$$

其中 $f(x) = \min(\text{popcnt}(x), n - \text{popcnt}(x))$ 。

令 $\text{cnt}_x = \sum_{i=1}^m [a_i = x]$ ，考虑枚举 $y = a_i \oplus x$ ，得

$$\begin{aligned} & \min_{x \in [0, 2^n)} \sum_{y \in [0, 2^n)} f(y) \sum_{i=1}^m [a_i \oplus x = y] \\ &= \min_{x \in [0, 2^n)} \sum_{y \in [0, 2^n)} f(y) \text{cnt}_{x \oplus y} \\ &= \min_{x \in [0, 2^n)} \sum_{j \oplus k = x} f(j) \text{cnt}_k \end{aligned}$$

FWT 即可，复杂度 $O(n2^n)$ 。

数树

我们知道由矩阵树定理求得的矩阵的行列式的值是该图所有生成树中边权乘积的和，即

$$\sum_T \prod_{e \in T} \text{val}_e$$

如果把给定树的树边标成 x ，非树边标成 1，那么一个生成树的所有边的权值积就是一个单项式 $1^p \cdot x^{n-1-p}$ ，其中 x 的指数就蕴含了有多少条边是给定树的树边。通过这个方法，对所有这样的单项式求和，就得到一个多项式，其中 x^k 的系数就表示恰好有 k 条边相同的生成树个数。我们又知道一个 $n-1$ 次多项式的系数只需要 n 个点值就能被确定，所以分别令 $x = 1 \dots n$ 跑矩阵树定理，得到一个线性方程组。最后高斯消元，就能得到多项式系数。

繁星点点

令 $G(x, y)$ 表示恰好有 x 行 y 列的颜色相等的方案数，那么答案就可以表示为 $3^{n^2} - G(0, 0)$

令 $F(x, y)$ 表示选 x 行 y 列为同种颜色，剩下的可以随便选的可重方案数。

尝试用 G 表示 F ，容易知道在 $F(x, y)$ 中， $G(i, j)$ 的每个方案，都被重复计数了 $\binom{i}{x} \binom{j}{y}$ 次。

$$F(x, y) = \sum_{i=x}^n \sum_{j=y}^n \binom{i}{x} \binom{j}{y} G(i, j)$$

那么根据二项式反演，有

$$G(x, y) = \sum_{i=x}^n \sum_{j=y}^n (-1)^{i-x+j-y} \binom{i}{x} \binom{j}{y} F(i, j)$$

代入 $x = 0, y = 0$ ，答案就是

$$3^{n^2} - \sum_{i=0}^n \sum_{j=0}^n (-1)^{i+j} F(i, j)$$

那么只需要将 F 求出即可，根据 F 的定义容易知道

$$F(i, j) = \begin{cases} \binom{n}{i} \binom{n}{j} \times 3 \times 3^{(n-i)(n-j)}, & ij \neq 0 \\ \binom{n}{i} \times 3^i \times 3^{n(n-i)}, & ij = 0 \text{ and } i \neq 0 \\ \binom{n}{j} \times 3^j \times 3^{n(n-j)}, & ij = 0 \text{ and } j \neq 0 \\ 3^{n^2}, & i + j = 0 \end{cases}$$

分情况求出 $G(0, 0)$ ，即按 F 的取值分类，易得

$$G(0, 0) = \sum_{i=1}^n \sum_{j=1}^n (-1)^{i+j} F(i, j) + 2 \sum_{i=1}^n (-1)^i F(i, 0) + F(0, 0)$$

第一项

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=1}^n (-1)^{i+j} \binom{n}{i} \binom{n}{j} \times 3 \times 3^{(n-i)(n-j)} \\ &= 3 \sum_{i=1}^n (-1)^i \binom{n}{i} \sum_{j=1}^n \binom{n}{j} (-1)^j \times (3^{n-i})^{n-j} \\ &= (*) \end{aligned}$$

对后面的式子用二项式定理，注意下指标从 1 开始，故还要减去第 0 项的值 $(3^{n-i})^n$ ，得

$$(*) = 3 \sum_{i=1}^n (-1)^i \binom{n}{i} [(3^{n-i} - 1)^n - 3^{n(n-i)}]$$

可以 $O(n \log n)$ 直接求了。对于第二项只有单重求和，不用化简， $O(n \log n)$ 直接求。

嘴大疑惑盒

区间线性基，线段树维护，一次合并 $\log |V|$ ，复杂度 $O(n \log n \log |V|)$ 。

或者前缀线性基，复杂度 $O(n \log |V|)$ 。

派对

答案是

$$\sum_{i=1}^n \binom{n}{i} i^k = \sum_{j=1}^k \left\{ \begin{matrix} k \\ j \end{matrix} \right\} j! \binom{n}{j} 2^{n-j}$$

复杂度 $O(k^2)$ 。

有序对

首先，有

$$a = b \times \lfloor \frac{a}{b} \rfloor + a \bmod b$$

答案是

$$\begin{aligned} & \sum_{a=1}^x \sum_{b=1}^y \left[b \cdot \lfloor \frac{a}{b} \rfloor = a \bmod b \right] \\ &= \sum_{a=1}^x \sum_{b=1}^y \left[\frac{a}{b+1} = \lfloor \frac{a}{b} \rfloor \right] \left[b+1 \mid a \right] \end{aligned}$$

令 $\frac{a}{b+1} = k$ ，那么

$$\begin{aligned} \lfloor \frac{a}{b} \rfloor &= \lfloor \frac{kb+k}{b} \rfloor = k + \lfloor \frac{k}{b} \rfloor \\ & \sum_{a=1}^x \sum_{b=1}^y \left[\frac{a}{b+1} = \lfloor \frac{a}{b} \rfloor \right] \left[b+1 \mid a \right] \\ &= \sum_{a=1}^x \sum_{b=1}^y \sum_k \left[a = k(b+1) \right] \left[1 \leq k < b \right] \\ &= \sum_{b=1}^y \min(b-1, \lfloor \frac{x}{b+1} \rfloor) \end{aligned}$$

左边单增，右边单减，找到交点。左边等差求和，右边整数分块，复杂度 $O(\sqrt{x})$ 。

基础 LCM 练习题

一个显然的想法是枚举 gcd，然后求去掉 gcd 后互质的两个数的乘积最大值。

一个显然的性质是如果 x 和 y 互质，那么对任意 $y' \leq y$ 都不会和 $x' \leq x$ 来更新答案。维护一个栈，从大到小枚举 x ，找到最大的与 x 互质的 y ，然后把栈里面所有 $y' \leq y$ 都弹掉。

如何找到最大的与 x 互质的数？实际上我们只需要知道栈里面存不存在与 x 互质的数。如果存在，那么就一直弹栈，因为迟早都要删掉。直到没有互质的了，最后一个被弹掉的就是要找的数。与 x 互质的数的个数为（其中 c_i 表示 i 的个数）

$$\sum c_i [\gcd(i, x) = 1] = \sum c_i \sum_{d|i, d|x} \mu(d) = \sum_{d|x} \mu(d) \sum_{d|i} c_i$$

所以只需要时刻维护 w_d 表示 d 的倍数的个数。

具体而言，预处理出每个数的约数，和 μ ，弹栈和入栈的时候更新就可以了。

复杂度 $O(\sum_{d=1}^{|V|} \frac{|V|}{d} |V|^{\frac{1}{3}}) = O(|V|^{\frac{4}{3}} \log |V|)$