数学专题

whx

October 4, 2017

前言

因为 ysy 已经讲过了数论的专题里的经典定理和习题，我主要在他的基础上补充一点更深入的理论知识，增进大家的理解。
noip2016 的趋势是原本的省选知识随着时代的发展下放到 noip，因此广阔的知识面也很重要。
也是为了后面几天讲课所需要的基础知识做铺垫。

原根

- ysy 讲过了质数一定会有原根，但是如何证明呢?（其实证明的工具 ysy 已经给出了）

原根

- ysy 讲过了质数一定会有原根，但是如何证明呢？（其实证明的工具 ysy 已经给出了）
- 考虑阶最大的元素 x，假设他的阶为 r，取一个 $y \notin \{x\}$

原根

- ysy 讲过了质数一定会有原根，但是如何证明呢？（其实证明的工具 ysy 已经给出了）
- 考虑阶最大的元素 x，假设他的阶为 r，取一个 $y \notin \{x\}$
- 如果 y 的阶不是 r 的约数，那么我们可以使阶 r' 更大，产生矛盾。

原根

- ysy 讲过了质数一定会有原根，但是如何证明呢?（其实证明的工具 ysy 已经给出了）
- 考虑阶最大的元素 x，假设他的阶为 r，取一个 $y \notin \{x\}$
- 如果 y 的阶不是 r 的约数，那么我们可以使阶 r' 更大，产生矛盾。
- y 的存在又与拉格朗日定理矛盾。

原根

- ysy 讲过了质数一定会有原根，但是如何证明呢?（其实证明的工具 ysy 已经给出了）
- 考虑阶最大的元素 x，假设他的阶为 r，取一个 $y \notin \{x\}$
- 如果 y 的阶不是 r 的约数，那么我们可以使阶 r' 更大，产生矛盾。
- y 的存在又与拉格朗日定理矛盾。
- 所以素数一定有原根。

BSGS

- 不知道 ysy 有没有讲这个算法

BSGS

- 不知道 ysy 有没有讲这个算法
- 利用结合率的性质分块

BSGS

- 不知道 ysy 有没有讲这个算法
- 利用结合率的性质分块
- 拓展？求 k 阶线性递推数列前 m 项里有没有连续的 k 项 $x_1 \ldots x_k$。

BSGS

- 不知道 ysy 有没有讲这个算法
- 利用结合率的性质分块
- 拓展？求 k 阶线性递推数列前 m 项里有没有连续的 k 项 $x_1 \ldots x_k$。
- 往后推根号项放到 hash 表里，然后算 $f_{t*\sqrt{n}}$

noip2014 解方程

- 给出一个一元 n 次方程，给出第 i 次项前系数 a_i，问此方程在 $[1, m]$ 上有多少可行解。

noip2014 解方程

- 给出一个一元 n 次方程，给出第 i 次项前系数 a_i，问此方程在 $[1,m]$ 上有多少可行解。
- $n \leq 100, a_i \leq 10^{10000}, m \leq 10^6$

noip2014 解方程

- 考虑 $\bmod p$ 检验，错误概率为 $\frac{1}{p}$

noip2014 解方程

- 考虑 $\bmod p$ 检验，错误概率为 $\frac{1}{p}$
- 如果只取一个 $p = 10^9$，复杂度为 $O(nm)$

noip2014 解方程

- 考虑 $\bmod p$ 检验，错误概率为 $\frac{1}{p}$
- 如果只取一个 $p = 10^9$，复杂度为 $O(nm)$
- 如果取三个 $p = 10^3$，只要检验 $x \bmod p_i$ 是否都为 0 即可，复杂度为 $O(np)$

noip2014 解方程

- 考虑 $\bmod p$ 检验，错误概率为 $\frac{1}{p}$
- 如果只取一个 $p = 10^9$，复杂度为 $O(nm)$
- 如果取三个 $p = 10^3$，只要检验 $x \bmod p_i$ 是否都为 0 即可，复杂度为 $O(np)$
- 觉得不够稳？策爷的确定性做法，考虑 $\bmod p$ 下不为 0 的时候，只有 $O(n)$ 个解，于是分块解决 $O(n\sqrt{nm})$。

Stain

- 坐标系上有 $k \leq 10^5$ 个污点，用一个 $n \times n$ 的正方形网格去覆盖这些污点。

Stain

- 坐标系上有 $k \leq 10^5$ 个污点，用一个 $n \times n$ 的正方形网格去覆盖这些污点。
- 可以放缩旋转这个网格，要求每个污点都在一个格点上。

Stain

- 坐标系上有 $k \leq 10^5$ 个污点，用一个 $n \times n$ 的正方形网格去覆盖这些污点。
- 可以放缩旋转这个网格，要求每个污点都在一个格点上。
- n 最小为多大?

高斯整数

- 唯一分解整环：存在唯一的分解可以使得两种之间只差可逆元

高斯整数

- 唯一分解整环：存在唯一的分解可以使得两种之间只差可逆元
- 高斯整数是唯一分解整环，也对应了一个正方形格

高斯整数

- 唯一分解整环：存在唯一的分解可以使得两种之间只差可逆元
- 高斯整数是唯一分解整环，也对应了一个正方形格
- 高斯整数可以做代余除法（模长的除法）

高斯整数

- 唯一分解整环：存在唯一的分解可以使得两种之间只差可逆元
- 高斯整数是唯一分解整环，也对应了一个正方形格
- 高斯整数可以做代余除法（模长的除法）
- 欧几里得整环一定是唯一分解环

高斯整数

- 唯一分解整环：存在唯一的分解可以使得两种之间只差可逆元
- 高斯整数是唯一分解整环，也对应了一个正方形格
- 高斯整数可以做代余除法（模长的除法）
- 欧几里得整环一定是唯一分解环
- 高斯整数 gcd

Solution

- 平移坐标系，把一个污点移动到原点上

Solution

- 平移坐标系，把一个污点移动到原点上
- 对所有坐标看做高斯整数求 gcd 即为答案

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$
- 有性质 $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$ 所以一定互质

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$
- 有性质 $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$ 所以一定互质
- 事实上连分数展开不仅可以保证 $\frac{1}{q_n q_{n+1}}$ 的精度，还可以保证是分母小于等于 q_n 的最佳近似

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$
- 有性质 $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$ 所以一定互质
- 事实上连分数展开不仅可以保证 $\frac{1}{q_n q_{n+1}}$ 的精度，还可以保证是分母小于等于于 q_n 的最佳近似
- Pell 方程 $x^2 - dy^2 = 1$ 的整数解

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$
- 有性质 $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$ 所以一定互质
- 事实上连分数展开不仅可以保证 $\frac{1}{q_n q_{n+1}}$ 的精度，还可以保证是分母小于等于 q_n 的最佳近似
- Pell 方程 $x^2 - dy^2 = 1$ 的整数解
- 他的解一定是对 $\sqrt{d}$ 连分数展开近似的某些项（和循环节有关）

Pell 方程和连分数展开

- 连分数展开，分子分母的递推式 $(p_n = a_n p_{n-1} + p_{n-1}, q_n = a_n q_{n-1} + q_{n-1})$
- 有性质 $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$ 所以一定互质
- 事实上连分数展开不仅可以保证 $\frac{1}{q_n q_{n+1}}$ 的精度，还可以保证是分母小于等于 q_n 的最佳近似
- Pell 方程 $x^2 - dy^2 = 1$ 的整数解
- 他的解一定是对 $\sqrt{d}$ 连分数展开近似的某些项（和循环节有关）
- 因为应用在大名鼎鼎的 shor 算法中，所以在这里提一下，感兴趣的同学可以自己学习。

期望可加性

- 把期望拆成若干个变量的和

期望可加性

- 把期望拆成若干个变量的和
- 不需要随机变量独立

期望可加性

- 把期望拆成若干个变量的和
- 不需要随机变量独立
- 几乎是最常用的技巧了

期望可加性

- 把期望拆成若干个变量的和
- 不需要随机变量独立
- 几乎是最常用的技巧了
- $E(X) = \sum_i P(X \geq i)$

随机变量独立

- 一个骰子，扔出 2 的倍数和 3 的倍数，二者概率独立吗?

随机变量独立

- 一个骰子，扔出 2 的倍数和 3 的倍数，二者概率独立吗?
- 准确的定义独立，一个随机变量的值不会影响到另外一个随机变量的概率分布。

随机变量独立

- 一个骰子，扔出 2 的倍数和 3 的倍数，二者概率独立吗?
- 准确的定义独立，一个随机变量的值不会影响到另外一个随机变量的概率分布。
- 相当于信息熵的贡献为 0。

关于期望的几个不等式

- $P(X \geq a \times E(X)) \leq \frac{1}{a}$

关于期望的几个不等式

- $P(X \geq a \times E(X)) \leq \frac{1}{a}$
- $P(X \geq E(X) + c \times \sigma) \leq \frac{1}{c^2}$

关于期望的几个不等式

- $P(X \geq a \times E(X)) \leq \frac{1}{a}$
- $P(X \geq E(X) + c \times \sigma) \leq \frac{1}{c^2}$
- 有什么实际意义呢？意味着期望复杂度内能出解的概率是个大于 0 的常数。

3-SAT 近似解

- 给出一个 3CNF-SAT，求一组近似解，使得至少 $\frac{7}{8}$ 的从句成立。

3-SAT 近似解

- 给出一个 3CNF-SAT，求一组近似解，使得至少 $\frac{7}{8}$ 的从句成立。
- Randomized algorithm

3-SAT 近似解

- 给出一个 3CNF-SAT，求一组近似解，使得至少 $\frac{7}{8}$ 的从句成立。
- Randomized algorithm
- Derandomization

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。
- 考虑如果我们现在有 t 个变量的取值是错的，我们随便取一个错误的分句翻转其中一个变量的取值。

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。
- 考虑如果我们现在有 t 个变量的取值是错的，我们随便取一个错误的分句翻转其中一个变量的取值。
- 有 $\frac{1}{3}$ 的概率我们离正确解更近一步了！我们每次随机一个初始解，进行 n 步这样的调整。

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。
- 考虑如果我们现在有 t 个变量的取值是错的，我们随便取一个错误的分句翻转其中一个变量的取值。
- 有 $\frac{1}{3}$ 的概率我们离正确解更近一步了！我们每次随机一个初始解，进行 n 步这样的调整。
- 那么正确概率就是 $\frac{1}{3^{n-t}}$

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。
- 考虑如果我们现在有 t 个变量的取值是错的，我们随便取一个错误的分句翻转其中一个变量的取值。
- 有 $\frac{1}{3}$ 的概率我们离正确解更近一步了！我们每次随机一个初始解，进行 n 步这样的调整。
- 那么正确概率就是 $\frac{1}{3^{n-t}}$
- 对所有二项分布的 t 的正确概率就是 $\frac{1}{2^n}\sum_t \frac{C(n,t)}{3^{n-t}} = (\frac{1}{2}+\frac{1}{6})^n = \left(\frac{2}{3}\right)^n$

3-SAT 随机算法

- 给出一个 n 个变量的 3CNF-SAT，求一组解。要求指数算法的底数尽量小。
- 随机调整！不妨假设只有一组解。
- 考虑如果我们现在有 t 个变量的取值是错的，我们随便取一个错误的分句翻转其中一个变量的取值。
- 有 $\frac{1}{3}$ 的概率我们离正确解更近一步了！我们每次随机一个初始解，进行 n 步这样的调整。
- 那么正确概率就是 $\frac{1}{3^{n-t}}$
- 对所有二项分布的 t 的正确概率就是 $\frac{1}{2^n}\sum_t \frac{C(n,t)}{3^{n-t}} = (\frac{1}{2}+\frac{1}{6})^n = \left(\frac{2}{3}\right)^n$
- 从而期望需要次数 1.5^n

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。
- 三染色视角下三角形三点一定颜色不同。

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。
- 三染色视角下三角形三点一定颜色不同。
- 只用 AB 染色，令 t 表示这种二染色正确的个数。(本来是 C 的一定会错误)

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。
- 三染色视角下三角形三点一定颜色不同。
- 只用 AB 染色，令 t 表示这种二染色正确的个数。(本来是 C 的一定会错误)
- 每次随机一个同色三角形的一个顶点翻转，分别有 $\frac{1}{3}$ 的概率 $t++$,t 不变,$t--$

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。
- 三染色视角下三角形三点一定颜色不同。
- 只用 AB 染色，令 t 表示这种二染色正确的个数。(本来是 C 的一定会错误)
- 每次随机一个同色三角形的一个顶点翻转，分别有 $\frac{1}{3}$ 的概率 $t++$,t 不变,$t--$
- $t=0$ 或者 $t=n$ 就得到了正确的染色

三分图二染色

- 可三染色图用两种颜色染色，使得没有同色三角形。
- 固定一组使用 ABC 三染色的方案，然后使用 AB 二染色。
- 三染色视角下三角形三点一定颜色不同。
- 只用 AB 染色，令 t 表示这种二染色正确的个数。(本来是 C 的一定会错误)
- 每次随机一个同色三角形的一个顶点翻转，分别有 $\frac{1}{3}$ 的概率 $t++$,t 不变,$t--$
- $t=0$ 或者 $t=n$ 就得到了正确的染色
- 随机游走期望步数为 n^2 步。(忽略 $t=0$ 也能得到正确染色的话）已经得到了一个多项式算法!

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率
- 每次单点染黑，随机一个任意颜色的点染，求期望多少次染为全黑?

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率
- 每次单点染黑，随机一个任意颜色的点染，求期望多少次染为全黑?
- 期望可加性，考虑第 i-1 个被染黑的点染黑到第 i 个被染黑的点染黑的期望，期望可加性

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率
- 每次单点染黑，随机一个任意颜色的点染，求期望多少次染为全黑?
- 期望可加性，考虑第 i-1 个被染黑的点染黑到第 i 个被染黑的点染黑的期望，期望可加性
- 每次给一个区间反色，求 m 次之后期望多少次全黑?

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率
- 每次单点染黑，随机一个任意颜色的点染，求期望多少次染为全黑?
- 期望可加性，考虑第 i-1 个被染黑的点染黑到第 i 个被染黑的点染黑的期望，期望可加性
- 每次给一个区间反色，求 m 次之后期望多少次全黑?
- 先差分一下，变成单点修改，每次找两个点出来反色

染色题合集

- 每次单点染黑，随机一个任意颜色的点染，求 m 次后的期望黑点个数?
- 期望可加性，考虑每个点为黑色的概率
- 每次单点染黑，随机一个任意颜色的点染，求期望多少次染为全黑?
- 期望可加性，考虑第 i-1 个被染黑的点染黑到第 i 个被染黑的点染黑的期望，期望可加性
- 每次给一个区间反色，求 m 次之后期望多少次全黑?
- 先差分一下，变成单点修改，每次找两个点出来反色
- 然后 f_i 表示有 i 个点为黑色的概率，$O(n)$ 解这个特殊的方程即可

染色题进阶

- min-max 容斥

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T \subseteq S} (-1)^{|T|-1} min\{T\}$

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T\subseteq S}(-1)^{|T|-1} min\{T\}$
- 每次单点染黑，随机一个任意颜色的点染，概率不均匀，求期望多少次有 n - 1 个黑色?

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T \subseteq S} (-1)^{|T|-1} min\{T\}$
- 每次单点染黑，随机一个任意颜色的点染，概率不均匀，求期望多少次有 n - 1 个黑色?
- $p_i \leq 1000$

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T \subseteq S} (-1)^{|T|-1} min\{T\}$
- 每次单点染黑，随机一个任意颜色的点染，概率不均匀，求期望多少次有 n - 1 个黑色?
- $p_i \leq 1000$
- 求次大值

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T \subseteq S} (-1)^{|T|-1} min\{T\}$
- 每次单点染黑，随机一个任意颜色的点染，概率不均匀，求期望多少次有 n - 1 个黑色?
- $p_i \leq 1000$
- 求次大值
- 期望可加性，拆成第 i 个不出现的情况。

染色题进阶

- min-max 容斥
- $max\{S\} = \sum_{T\subseteq S}(-1)^{|T|-1}min\{T\}$
- 每次单点染黑，随机一个任意颜色的点染，概率不均匀，求期望多少次有 n - 1 个黑色?
- $p_i \leq 1000$
- 求次大值
- 期望可加性，拆成第 i 个不出现的情况。
- 然后 min-max 容斥，背包一下即可。

染色题进阶

- 每次给一个区间染色，求全染黑的期望次数?

染色题进阶

- 每次给一个区间染色，求全染黑的期望次数?
- 还是应用 min-max 容斥

染色题进阶

- 每次给一个区间染色，求全染黑的期望次数?
- 还是应用 min-max 容斥
- 考虑问题变成每个子集第一次被染黑的次数之后，贡献只和跨越这个子集的区间数有关

染色题进阶

- 每次给一个区间染色，求全染黑的期望次数?
- 还是应用 min-max 容斥
- 考虑问题变成每个子集第一次被染黑的次数之后，贡献只和跨越这个子集的区间数有关
- dp 一下即可

HAOI2015 T2

- 一个有 n 个元素的集合，你有一个空集

HAOI2015 T2

- 一个有 n 个元素的集合，你有一个空集
- 每次你会选择集合的一个子集，与当前集合进行取并集操作

HAOI2015 T2

- 一个有 n 个元素的集合，你有一个空集
- 每次你会选择集合的一个子集，与当前集合进行取并集操作
- 问期望多少次后第一次变成全集，选中每个子集的概率是输入给你的

HAOI2015 T2

- 考虑把概率表示成集合幂级数 f，期望等于不是全集的概率之和。

HAOI2015 T2

- 考虑把概率表示成集合幂级数 f，期望等于不是全集的概率之和。
- 也就是 $\sum_i f^i = \frac{1}{1-f}$

HAOI2015 T2

- 考虑把概率表示成集合幂级数 f，期望等于不是全集的概率之和。
- 也就是 $\sum_i f^i = \frac{1}{1-f}$
- 子集和变换之后每个点值的概率取 $\frac{1}{1-p}$ 再变换回来即可

HAOI2015 T2

- 如果不借助 or 卷积呢?

HAOI2015 T2

- 如果不借助 or 卷积呢?
- min-max 容斥

HAOI2015 T2

- 如果不借助 or 卷积呢?
- min-max 容斥
- 变成了每个子集中第一次有元素被染黑的时间

HAOI2015 T2

- 如果不借助 or 卷积呢?
- min-max 容斥
- 变成了每个子集中第一次有元素被染黑的时间
- 等于统计有交的集合个数，也就是统计被它的补包含的集合个数

HAOI2015 T2

- 如果不借助 or 卷积呢?
- min-max 容斥
- 变成了每个子集中第一次有元素被染黑的时间
- 等于统计有交的集合个数，也就是统计被它的补包含的集合个数
- 子集和变换之后直接算答案即可

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧
- 向量

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧
- 向量
- 正交和线性相关

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧
- 向量
- 正交和线性相关
- 矩阵和矩阵的秩

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧
- 向量
- 正交和线性相关
- 矩阵和矩阵的秩
- 矩阵乘法和它的不同理解

向量、矩阵、线性相关

- 因为是 noip 课程，所以仔细的说下定义吧
- 向量
- 正交和线性相关
- 矩阵和矩阵的秩
- 矩阵乘法和它的不同理解
- 矩阵的行空间列空间和零空间

行列式

- 逆序对形式

行列式

- 逆序对形式
- 对初等变换的性质

高斯消元

- 线性方程组，写作 $Ax = b$

高斯消元

- 线性方程组，写作 $Ax = b$
- 高斯消元就是每次用 i 为主元消去别的方程里的 i

高斯消元

- 线性方程组，写作 $Ax = b$
- 高斯消元就是每次用 i 为主元消去别的方程里的 i
- 求线性方程组的解

高斯消元

- 线性方程组，写作 $Ax = b$
- 高斯消元就是每次用 i 为主元消去别的方程里的 i
- 求线性方程组的解
- 求行列式

高斯消元

- 线性方程组，写作 $Ax = b$
- 高斯消元就是每次用 i 为主元消去别的方程里的 i
- 求线性方程组的解
- 求行列式
- 矩阵的逆和伴随矩阵

matrix-tree

- 基尔霍夫矩阵：度数矩阵减去邻接矩阵

matrix-tree

- 基尔霍夫矩阵：度数矩阵减去邻接矩阵
- matrix-tree：删去某个点对应的一行一列求行列式

matrix-tree

- 基尔霍夫矩阵：度数矩阵减去邻接矩阵
- matrix-tree：删去某个点对应的一行一列求行列式
- 有向图的情况：入度矩阵减去邻接矩阵

matrix-tree

- 基尔霍夫矩阵：度数矩阵减去邻接矩阵
- matrix-tree：删去某个点对应的一行一列求行列式
- 有向图的情况：入度矩阵减去邻接矩阵
- 删去 root 的行和列求行列式

一些生成树计数题

- 求随机的生成树的期望权值

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数
- 生成函数，插值

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数
- 生成函数，插值
- 随机的生成树权值之和的 k 次方的期望（hihocoder28 D）

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数
- 生成函数，插值
- 随机的生成树权值之和的 k 次方的期望（hihocoder28 D）
- k 次方拆成从生成树上有序的选 i 条可以相同的边的权值之积

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数
- 生成函数，插值
- 随机的生成树权值之和的 k 次方的期望（hihocoder28 D）
- k 次方拆成从生成树上有序的选 i 条可以相同的边的权值之积
- 直接生成函数表示一个边选几次或者不选，插值即可

一些生成树计数题

- 求随机的生成树的期望权值
- 期望可加性
- 求恰好有 t 个黑边的生成树个数
- 生成函数，插值
- 随机的生成树权值之和的 k 次方的期望（hihocoder28 D）
- k 次方拆成从生成树上有序的选 i 条可以相同的边的权值之积
- 直接生成函数表示一个边选几次或者不选，插值即可
- 复杂度很高，但是是多项式的

一些生成树计数题

- 求 root 为 n 的生成树形图的所有方案的边权和之和

一些生成树计数题

- 求 root 为 n 的生成树形图的所有方案的边权和之和
- 有向图和无向图比有一个好：它影响到的两个位置（边和度数）在矩阵上在同一行

一些生成树计数题

- 求 root 为 n 的生成树形图的所有方案的边权和之和
- 有向图和无向图比有一个好：它影响到的两个位置（边和度数）在矩阵上在同一行
- 展开的余子式算这个边对方案数的贡献即可

一些生成树计数题

- 求 root 为 n 的生成树形图的所有方案的边权和之和
- 有向图和无向图比有一个好：它影响到的两个位置（边和度数）在矩阵上在同一行
- 展开的余子式算这个边对方案数的贡献即可
- 伴随矩阵