Almost Convex(QOJ7906)

给定平面坐标系中 n 个点构成的点集 S，保证三点不共线。称一个简单多边形是好的，当且仅当其顶点均为 S 中元素，且 S 中所有点均在多边形的边上或内部。

设 f 为好多边形中的顶点数最小值，求有多少个好多边形的顶点数 $\le f+1$。

$3 \le n \le 2000$。

sol

显然 f 为凸包大小，其余好多边形为凸包删一条边 (x,y)，再选一个点 z，加入 (x,z) 和 (y,z)，枚举边，将剩下的点分别按 x 和 y 极角排序，一个点可以作为 z 当且仅当没有被二维偏序，所以做完了。复杂度 $O(n^2 \log n)$。

Information Spread(QOJ7885)

给定 n 个点，m 条边的有向图，每条边有权值 $w(0 \le w \le 1)$。

每个点有两个 01 权值 a_i 和 v_i，初始时 $a_i = [i=1], v_i = 0$。

定义如下伪代码：

```
void dfs(int x)
{
    if (v[x])
        return;
    v[x] = 1;
    for (auto [y, w] : E[x])
    {
        if(a[x] == 1 && a[y] == 0)
        {
            if(random(w)) // random(w) return true with probability w.
                a[y] = 1;
        }
        dfs(y);
    }
    return;
}
```

求出在调用一次 `dfs(1)` 之后，每个 a_i 等于 1 的概率，对 998244353 取模。

$3 \le n \le 10^5, n-1 \le m \le 3 \cdot 10^5$，保证点 1 能到达所有点。

sol

先拉出 dfs 树，注意到非树边只会把单个点的 a 改成 1，不会造成后续影响，所以对每个点只用考虑直接指向它的非树边，建虚树跑 dp 就行，复杂度 $O(m \log m)$。

Energy Distribution(QOJ7753)

给定一个 n 个点的完全图，点 i 与 j 之间连边有权值 $w_{i,j}(0 \le w_{i,j} \le 1000)$。

需要给点分配非负实数点权 e_i，满足 $\sum e_i = 1$，最大化 $\sum\limits_{1 \le i < j \le n} e_i e_j w_{i,j}$。只需求出这个最大值。

绝对或相对误差 $\le 10^{-6}$。

$1 \le n \le 10$。

sol

考虑拉乘，手算一下得到线性方程组。由于有非负的限制，所以要枚举取 0 的 e_i 集合。如果解出非唯一解或负数解则直接舍去。可以舍去非唯一解的原因是如果这个解包含最优解，则可以通过微调让一个本不为 0 的 e_i 取 0，故最优解会在 0 更多的情况中被考虑。复杂度 $O(2^n n^3)$。

Bus Lines(QOJ7610)

一颗 n 个点的树，有 m 条公交线路，通过第 i 条线路可以在 x_i 到 y_i 的简单路径上任意两点间移动，移动一次的代价为 1，保证任何两点联通。定义 $f(i,j)$ 表示点 i,j 最短路，对每个 i 求 $\sum\limits_{j \ne i} f(i,j)$。

$1 \le n, m \le 2 \cdot 10^5$。

sol

把问题改成，设点 i 花 j 步能到的点集为 $S_{i,j}$，求 $\sum\limits_j (n - |S_{i,j}|)$，若以 i 为根，则 $n - |S_{i,1}|$ 即为 $\sum siz_x$，其中 $x \notin S_{i,1}$ 且 $fa_x \in S_{i,1}$，由于图连通，这些 x 在第二步就会被访问，变成子问题，即 $f_i = \sum siz_x + f_x$。剩下的优化就是数据结构各显神通了。

Balanced Array(QOJ7877)

定义长为 l 的数组 a 是好的，当且仅当 $\exists 1 \le k \le \frac{l-1}{2}$，使得 $\forall 1 \le i \le l-2k, a_i + a_{i+2k} = 2a_{i+k}$。

给定长为 n 的数组 A，求其的每个前缀是否是好的。

$1 \le n \le 2 \cdot 10^6$。

sol

考虑哈希+双指针。复杂度 $O(n)$。

Palindrome Path(QOJ7751)

一个 $n \times m$ 的平面迷宫，每一格要么是空地要么是障碍，给定起点终点，需要通过上下左右移动从起点移动到终点，且必须经过所有空地。

允许往障碍上走，但此时只会原地不动。

需判断无解，有解输出一个仅包含 `LRUD` 的字符串表示构造。

要求构造的字符串长度 $\le 10^6$ 且为回文串。

$1 \le n, m \le 30$。

sol

可以构造一个串让当前的人向任意方向走一步，且如果人在终点，走这个串倒过来则还会留在终点。具体思路是先往一个方向走到撞墙再掉头。

Say Hello to the Future(QOJ7899)

给定长为 n 的数组 a，定义一个划分方案为选定 $1 = r_0 < r_1 < r_2 < \cdots < r_k = n + 1$，即将数组劈成非空的 k 个子串。若 $\forall 1 \le i \le k, r_{i-1} \le j < r, a_j \le r_i - r_{i-1}$，则称划分方案是好的。

对于 $i \in [1, n]$，求出将 a_i 改为 1 后好的划分方案数，对 998244353 取模。

$1 \le n \le 2 \cdot 10^5$。

sol

考虑 cdq 分治。用二维偏序算左到右的贡献，复杂度 $O(n \log^2 n)$。

Bocchi the Rock(QOJ7559)

取圆上 n 个不同点，其将圆划分为 n 个弧。点可以染成红色或蓝色，弧可以染成黄色或粉色。

对于一个确定了染色方案的圆，定义连线方案为：选取若干对同色点，每一对之间连一条线段，要求线段不得相交（包括端点）。

连线方案会将圆划分成若干区域，称一种染色方案是好的，当且仅当存在一个连线方案，使得每个区域里的弧线全部同色。

圆上的部分点和部分弧已经染上色了，求有多少种将剩下的点和弧染色的方案是好的，对 998244353 取模。

$1 \le n \le 5 \cdot 10^4$。TL：10s。

sol

合法性判定为找到相邻两条弧颜色不同的点，判断它们能否两两同色匹配使线段不交。相当于一个首尾相接的 01 序列每次能删相邻的两个 0 或两个 1，删空即合法。不考虑首位相接时任何一个序列都能先删成 $010101\ldots$，只需要记录长度和第一位的值，合并两个信息就是加法或减法卷积，分治 ntt 即可。

Rolling For Days(QOJ7754)

卡池中有 n 张卡，类型为 $1 \sim m$，你需要 b_i 张类型为 i 的卡。

每次从卡池中等概率随机抽取一张卡，如果手上已经有 b_i 张与该卡同类型的卡，则将其放回卡池，否则收下。每个类型都有 b_i 张之后停止。求期望抽多少次。对 998244353 取模。

$1 \le n \le 1000, 1 \le m \le 12$，保证卡池中至少有 b_i 张类型为 i 的卡。

sol

先弄一个算概率的 dp 过程 $f_{S,x}$，S 表示还没顶到上界的类型集合，x 当前抽到的卡中对应类型在 S 中的卡片数量，枚举下一次抽卡有没有把某一类抽完，转移到 $f_{S,x+1}$ 或 $f_{S\setminus\{i\},x+1-b_i}$，初始值 $f_{\emptyset,0} = 1$，则有 $f_{\{1,2,\ldots,m\},0} = 1$，每一种方案对应的都是这两个状态之间的一条路径，且可以发现其期望抽数仅与这条路径经过了哪些点有关，因此把这个 dp 正着倒着各跑一遍算出经过每个点的概率，乘上每个点的期望抽数累加即可。

Colonization(QOJ7609)

在一颗树上初始有 k 个人从根处开始走，走过的点会被标记，如果任何时刻出现一个被标记的没有人的点与未被标记的点相连则失败。一颗无根树的权值为最小的 k 使得存在一个根和游走方案让所有点被标记。

给定 n，对每个 k 求出 n 个点权值为 k 的无标号无根树数量 $\bmod 10^9+7$。

$1 \le n \le 500$。

sol

对于给定树和根 a 用树形 dp 求 k_a 是容易的，且可以观察出 $k_a = O(\log n)$，对于同一颗树不同的根 a, b，$|k_a - k'_b| \le 1$。计算无标号无根树可以考虑找重心，对有根树计数同时要求儿子 $siz < \frac{n}{2}$，特判树有两个重心的情况。只考虑根的 k 时，可以设 $f_{i,j,1/2/3}$ 表示大小为 i 的有根树，$k_i = j$，且 i 的儿子中 k 为最大值的有 1 个、2 个还是更多个的方案数。再考虑换成其他地方 k 更小的情况，$g_{i,j,k}$ 表示大小为 i 的有根树，$k_i = j+1$，且如果再给根接一个 k 不超过 x 的儿子则能换根换到 k 取 j，否则取不到，可知 $g_{i,j,k}$ 要么来自于 $f_{i,j+1,2}$（此时 $k = j-1$），要么恰有一个 k 取最大值的儿子，枚举这个儿子是哪个 $g_{*,j,*}$ 转移。复杂度大概是 $O(n^2 \log^2 n)$。