QOJ6555

Statement

给定 n ($n \le 1000$) 个点无向图 $G=(V,E)$。求导出子图边数为偶数的点集数量。

Solution

考虑一个 n 元二次多项式 $F(x_1,\dots,x_n) := \sum_{i\neq j,\ (i,j)\in E} x_i x_j + \sum_{(i,i)\in E} x_i$。题目相当于求 n 元组 $(x_1,\dots,x_n)\in\{0,1\}^n$ 使得 $F(x_1,\dots,x_n) \bmod 2 = 0$。

直接计算 $=0$ 的方案有些困难，所以可以考虑计算 $=0$ 和等于 $=1$ 的方案数差值。

考虑拆开 F：$F(x_1,\dots,x_2) = x_1 L(x_2,\dots,x_n) + Q(x_2,\dots,x_n)$。

如果 $L(x_2,\dots,x_n)=1$，那么对于每一种 $x_2,\dots,x_n$ 方案，x_1 选与不选恰好对应两种最终 F 值不同的方案，所以对差值的贡献为 0。

那么只剩下 $L(x_2,\dots,x_n)=0$ 的情形。注意到 $L(x_2,\dots,x_n)$ 是一个一次的多项式，所以可以移项得到 x_2 关于 $x_3,\dots,x_n$ 的表达式。将这个表达式回带到 Q 得到 $Q'(x_3,\dots,x_n)$。那么我们就是要求 $Q'(x_3,\dots,x_n)=k$ ($k\in\{0,1\}$) 的方案数，这是一个形式相同子问题，递归计算即可。

时间复杂度 $O(n^3)$（吐槽数据范围）。

CF1919E

Statement

对于一个长为 n 的只包含 $1,-1$ 的序列 a，记其前缀和数组为 p。现在给出 n 和排序后的 p，计数有多少个序列 a 满足条件。$1\le n\le 5000$。

Solution

把前缀和转化为直线，那么题目相当于告诉我们每一个高度的顶点出现了多少次，记 i 出现次数为 c_i。

先枚举 s 表示最终 a_i 的和，然后令 $a \leftarrow \{\underbrace{1,1,\cdots,1}_{s\text{ 个}},\underbrace{-1,-1,\cdots,-1}_{p_n-s\text{ 个}}\}$。接着，我们每次向一个位置插入 $\{-1,1\}$，不断插入就可以构造出最终的序列。最终序列与构造方式是一一对应的。

如果我们在一个高度为 k 的位置后插入，那么高度为 $k,k-1$ 的点数量均增加 1。于是我们可以从高到低考虑。对于最高点，初始只有 1 个，那么就需要插入 $c_{p_n}-1$ 次，使用插板法计算方案数；对于高度为 p_n-1，我们知道初始时有几个点，也知道上一次的贡献，于是也能够计算；以此类推，就能够求出方案数了。

一次计算是 $O(n)$ 的，总复杂度 $O(n^2)$。

CF1867F

Statement

给定 n 个点的有根树 G，记 $P(G)=\{\text{subtree}(u)\mid 1\le u\le n\}$。构造另一个 n 个点的有根树 G'，使得 $|\{u\mid \exists T\in P(G):\text{subtree}(u)\sim T\}|$ 最小化，其中 $\sim$ 指有根树的同构。

Solution

我们称存在 $P(G)$ 中元素与其同构的子树为*同构存在*的。

注意到一个事实，如果 T 是同构存在的，那么其所有子树都是同构存在的。那么如果我们能够找到一个大小最小的树 T 使得其不是同构存在的，那么我们可以令 G' 为 T 上方接一条长度为 $n-|T|$ 的链，这样的 G' 一定满足题目最小化的条件。

找到最小的 T 可以枚举 $k=1,2,\dots,n$，暴力搜索所有大小为 k 的无标号有根树，并树哈希判断其是否是同构存在的。根据抽屉原理，最多搜索 $n+1$ 个树就能找到一个不是同构存在的树。精细实现搜索过程即可做到复杂度 $O(n)$。

CF1260F

Statement

给定一棵 n 个节点的树，每个节点有一个颜色 h，h_i 为 $[L_i,R_i]$ 内的一个整数。

现在，对于所有 $\prod(R_i-L_i+1)$ 种不同的染色方案，求出下列式子之和：

$$\sum_{h_i=h_j,1\le i<j\le n}\text{dis}(i,j)$$

$2\le n\le 10^5,1\le L_i\le R_i\le 10^5$，答案对 10^9+7 取模。

Solution

CF1799G

Statement

n 个人，每人可以给任意一个人投一票，但是不能投给同组的人（包括自己）。第 i 个人属于第 t_i 组，希望得到 c_i 票，求使得每个人都满足愿望的投票方案数，模 998244353。

$\sum_{i=1}^n c_i=n$，$1\le t_i\le n$，$0\le c_i\le n$，$1\le n\le 200$。

Solution

AGC005F Many Easy Problems

Statement

从前，有这样一个问题：

> *给定一棵 n 个点的数和一个整数 K，对于一个节点集合 $S \subseteq V$，定义 $f(S)$ 为其在树上的斯坦纳树的大小（即满足 $S \subseteq T \subseteq V$ 且 T 连通时 $|T|$ 的最小值）。求 $\sum\limits_{S\subseteq V,|S|=k} f(S)$，模 $924844033 = 2^{21} \times 441 + 1$。*

现在有人觉得这题太简单，想对 $K = 1, 2, \ldots, n$ 求出上面这个问题的答案。帮帮他。$n \le 2 \times 10^5$。

Solution

问题问的是“和的和”，考虑拆贡献。考虑每个点在哪些情况下会产生贡献。

看起来“被选到”挺麻烦的，考虑每个点在哪些情况下不会产生贡献。发现点 u 不会产生贡献当且仅当所选 K 个点都在它的同一个子树内。

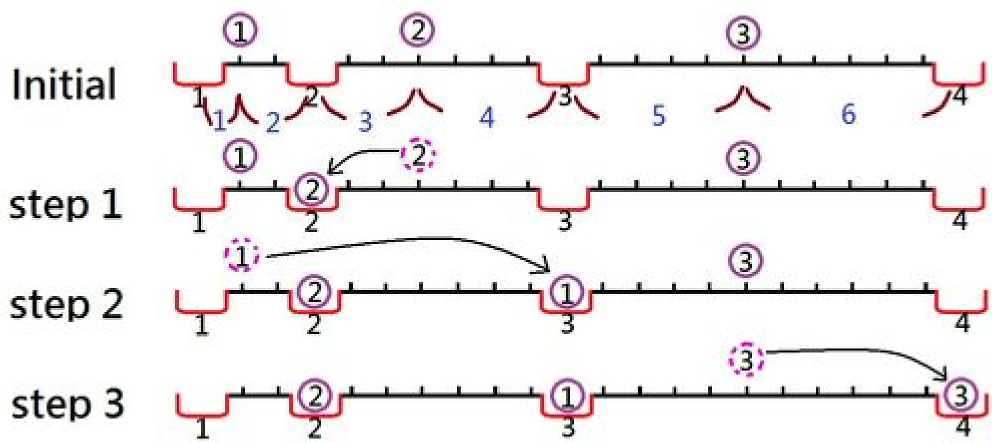
于是 u 的贡献就是 $\sum\limits_{(v,u)\in E} \binom{siz_v}{K}$，其中 siz_v 是删去 u 之后 v 所在子树的大小。

所以总贡献会形如 $\sum\limits_{i=1}^{n-1} c_i \binom{i}{K}$，其中 $\sum c_i = 2n - 2$，我们要对每个 K 求这个式子。

组合数拆掉，变成 $\dfrac{1}{K!} \sum\limits_i c_i i! \dfrac{1}{(i-K)!}$，这就是差卷积的形式，NTT 即可。复杂度 $O(n \log n)$。

AGC007C Pushing Balls

Statement



一条直线上有 n 个球和 $n+1$ 个洞，排列形如 $hole_0, ball_1, hole_1, ball_1, \ldots, ball_n, hole_n$。每两个物体（洞或球）之间的**距离**从左到右形成等差数列（注意不是位置）。

接下来 n 秒，每秒均匀随机一个球会均匀随机向左滚或向右滚，滚到洞里后这个球和这个洞就消失了，花费是球到洞的距离。n 秒后所有球都会消失。

求总花费的期望。显然每个球都会进一个洞。$n \le 2 \times 10^5$，等差数列的首项和公差 ≤ 100。

Solution

问题问的是“和的和”，考虑拆贡献。把贡献拆到每一段上面。

答案形如每段的长度乘上它产生贡献的期望次数。然后发现由于推球的动作非常对称，所以左数第一段产生贡献的期望次数等于右数第一段产生贡献的期望次数，左数第二段产生贡献的期望次数等于右数第二段产生贡献的期望次数，以此类推。于是我们可以把左数第 k 段的长度和右数第 k 段的长度取平均而答案不变！

一个等差数列取完平均后每一段长度都相等了，赢！

不妨设现在每一段长度都为 1，考虑推完球之后第 i 段的期望长度是多少。

模拟一下，假设现在是 `111111`，推完球之后的可能为：`1111`，`3111`，`1311`，`1131`，`1113`，`1111` 均匀随机。

于是新的每一段的期望长度也都相同。依次做即可。复杂度 $O(n)$。

CTSC2018 青蕈领主

Statement

给定长度为 n 的序列 a，求有多少个排列 p 满足：a_i 为最大的整数 $k(k \le n)$ 满足 $[i,k]$ 是一个连续段。对 998244353 取模。

定义一个区间 $[l,r]$ 是连续段当且仅当 $\max(l,r)-\min(l,r)=r-l$，其中 $\max(l,r),\min(l,r)$ 分别是 $[l,r]$ 中 p 的最大值和最小值。

T 组数据。

$n \le 5\times 10^4, T \le 100$。

Solution

利用一点连续段的结论，首先区间是不交或者包含的。

然后设 f_i 表示长度为 $i+1$ 的排列满足所有长度大于 1 的连续段右端点都在 $i+1$ 的排列的数量，那么答案就是 $\prod f_{|son_i|}$，其中 son_i 表示第 i 个点的儿子集合。

于是问题变成了如何求 f。

首先直接用析合树来处理复杂度是 $O(n^3)$ 的，无法接受。

接下来为了方便，我们先对排列转置，现在变成了数所有长度 >1 的连续段的最大值都为 $n+1$ 的方案数。

考虑插入法，考虑长度为 $n+1$ 的合法排列如何生成。

考虑删除 1，然后把剩下的数减去 1，如果删除之后剩下的排列是合法的，那么转移参数是好算的，注意 1 不能在 2 旁边。

如果不合法，考虑不合法的情形。

例如 `3 1 4`，删除 1 后就变成了 `3 4`，就不合法了，考虑如何处理这一点贡献。

首先不难发现出现不合法的情况之后不包含 $n+1$ 的极长连续段是唯一的，所以我们可以枚举这个连续段，因为是一个连续段所以我们可以把这个连续段看成一个点。

问题变成了如何处理连续段内部的贡献以及 1 的位置，观察一下，如果我们把 1 改成这个连续段的最大值 $+1$，那么删除 1 后出问题当且仅当原排列合法，所以又变成了一个子问题。

于是直接计算即可，写出方程有 $f_n=(n-1)f_{n-1}+\sum_{i=2}^{n-2} f_i f_{n-i}(n-i-1)$，可以用分治 NTT 做到 $O(n\log^2 n+Tn)$。

CF1844H

给定长度为 n 的排列 p，初始所有位置都不确定，然后接下来 n 次每次确定排列中的一个位置的值，每次操作之后求有多少确定排列剩下位置的数的方案使得排列是好的。

定义一个排列是好的当且仅当这个排列的每个环都是 3 的倍数。

$3 \le n \le 3\times 10^5, 3 \mid n$。

Solution

首先我们只关注长度 $\bmod 3 = 0, 1, 2$ 的链的数量。

然后发现长度 $\bmod 3 = 0$ 的链很好处理，最后乘点东西即可。

设长度 $\bmod 3 = 1, 2$ 的链数量分别为 a, b。

设 $g(a, b)$ 为对应的方案数。

考虑如何转移，一种想法是枚举一个长度为 1 的链的下一个是什么，然后把这东西拼起来。

如果下一个长度还是 1，那么贡献就是 $(a-1) \times g(a-2, b+1)$，如果长度为 2 贡献就是 $b \times g(a-1, b-1) \times (a+b-1)$。

于是有 $g_{a,b} = (a-1)g_{a-2,b+1} + b(a+b-1)g_{a-1,b-1}$，当然是对称的，所以还能写成 $g_{a,b} = (b-1)g_{a+1,b-2} + a(a+b-1)g_{a-1,b-1}$。

然后发现一个非常 6 的事情，两个式子总共只访问到了 $g_{a,b}, g_{a-1,b-1}, g_{a-2,b+1}, g_{a+1,b-2}$，也就是说我们可以知二推二。

于是维护一下 $(a, b), (a-1, b-1), (a-2, b+1), (a+1, b-2)$ 这四个位置的值，然后就可以递推了。