解题报告

北京市十一学校 时中

1 Inversions

1.1 试题来源

Atcoder Grand Contest 023E
`https://atcoder.jp/contests/agc023/tasks/agc023_e`

1.2 题目大意

给一个长为N的序列$\{A\}$，对所有满足$P_i \le A_i$的排列P，求其逆序数之和。

1.3 数据范围

$1 \le N \le 2 \times 10^5$，$1 \le A_i \le N$

1.4 解题过程

首先考虑如何求出满足条件的排列的个数。设$C_k = \sum_{i=1}^{N} [A_i \ge k] - N + k$,那么总个数$S = \prod_{i=1}^{N} C_i$。
对于计算答案，比较合理的方式应该是对每对$(i, j)(i < j)$计算在哪些排列里会产生贡献。对于$A_i \le A_j$的情况，只要将A_j改为A_i,计算出排列个数s。那么(i, j)在$\frac{s}{2}$个排列里是逆序对。

原因比较简单。因为$P_i > P_j$，所以当$A_i < P_j$时显然不会是逆序对。而对$A_i \ge P_j$的情况，$(\cdots, P_i, \cdots, P_j, \cdots)$与$(\cdots, P_j, \cdots, P_i, \cdots)$一一对应，所以有一半是逆序的。

同理，虽然$A_j > P_i$的情况下逆序时的贡献不好计算，只要计算顺序对的出现次数，从全部排列中减去就可以得到逆序的贡献。

以下只针对$A_i \le A_j$的情况讨论。
直接计算复杂度过大。考虑修改的变化不大:

$$s = \prod_{i=1}^{A_i} C_i \times \prod_{A_i+1}^{A_j} (C_i - 1) \times \prod_{A_j+1}^{N} C_i$$
$$T_k = \sum_{i=1}^{k} \frac{C_i - 1}{C_i}$$
$$s = \frac{T_{A_j}}{T_{A_i}} S$$

可以看出，一个数对的两部分分别产生贡献。只要对每个i统计满足$i < j, A_i \le A_j$的$\sum T_{A_j}$即可。这可以使用树状数组做二维偏序在$O(n \log n)$的时间内求得。通过简单的预处理，C_i, T_i均可$O(n)$求得。
上式可能会有除0的问题。对$C_i = 0$的情况，代表没有满足题目约束的排列，特判即可。将T_k表示为$T_k = b_k * 0_k^c$。那么s非零当且仅当$c_{A_i} = c_{A_j}$，显然c单调递增，这样的T是一段区间，可以直接查询。通过预处理可以$O(1)$定位这段区间。
通过上述方法，即可在$O(n \log n)$的时间复杂度内通过此题。

2 One Third

2.1 试题来源

Atcoder Grand Contest 032F
https://atcoder.jp/contests/agc032/tasks/agc032_f

2.2 题目大意

在周长为1圆上均匀随机地选N个点，对所有两点之间的圆弧x，找到使$|x - \frac{1}{3}|$最小的。求$|x - \frac{1}{3}|$的期望值。

2.3 数据范围

$2 \le N \le 10^6$

2.4 解题过程

$|x-\frac{1}{3}|$不好表示，不妨在加入一个点p时再加入另外两个辅助点p', p'',放在p经过$\frac{1}{3}$与$\frac{2}{3}$的位置。这样，两点之间的距离x的$|x-\frac{1}{3}|$就变成了一个点和另一个点辅助点之间的距离。

在这样的表示方法下，我们不妨称一个点和其辅助点分别由不同的颜色(不妨称p是红色，p'是蓝色，p''是白色)。由于0长线段一定不是最优的解，原问题即是找到端点颜色不同且包含红色的最短线段。通过松弛条件，我们就是要找到端点颜色不同的最短线段。因为对于一个不包含红色的异色线段，旋转120°,总能获得一个对应的含红线段。同样由于对称性，我们只需考察一个三分之一圆弧内的最小值。

以一个点的红点和蓝点截取这个三分之一圆弧。随机选点等价于在这段圆弧上随机选择一个点并随机选择一个颜色。此时我们只需考察相邻的两点，因为对(a, b, c),其中(a, c)异色，必有(a, b)异色或(b, c)异色。我们考虑计算出有k个异色相邻点的概率P_k，以及在此时最短边的期望长度L_k。第一部分只涉及相对位置关系，而第二部分只与切割位置有关，两者独立。

对P_k，若有k个相邻异色段，则必有$k+1$个连续相同颜色段。问题转化为求起始为红色，结束为蓝色的颜色序列个数；与求将n个物品分成k个连续非空段的方案数，都可以$O(n)$求得。

对L_k,问题也是两部分:求k段的期望长度和s_k，以及将单位线段分成k段的最短段的期望长度m_k。对s_k，由于每一段的期望长度均为$\frac{1}{3n}$,$s_k=\frac{k}{3n}$ 对m_k,设$Q(x)$表示最短段的长度大于x的概率。则:

$$
\begin{aligned}
m_k &= \int_0^{\frac{1}{k}} Q(x)dx \\
&= \int_0^{\frac{1}{k}} (1-kx)^{k-1}dx \\
\int (1-kx)^{k-1}dx &= -\frac{1}{k^2}(1-kx)^k + c \\
m_k &= \frac{1}{k^2}
\end{aligned}
$$

$Q(x)=(1-kx)^{k-1}$是考虑先对每一段预留x,然后随机分配。这样就得到$L_k = s_k * m_k = \frac{1}{3nk}$

最终答案即为$\sum P_k L_k$,复杂度$O(n)$

3 Drazil and Morning Exercise

3.1 试题来源

Codeforces Round 292 (Div. 1)
`https://codeforces.com/contest/516/problem/D`

3.2 题目大意

给一颗n个点的有正边权v_i树，每个点的权值为离它最远的叶子和这个点的距离。有q次询问，请找到一个包含点最多的联通块，使它包含的所有点的权值的极差$\leq l$。

3.3 数据范围

$2 \leq N \leq 10^5$，$1 \leq q \leq 50$

3.4 解题过程

一个直接的想法是考虑计算每个叶子的权值（可以使用换根DP方法$O(n)$求得），然后使用双指针维护权值极差$\leq l$的点集。问题就变为树上动态加/删点/边，求最大联通块大小，可以直接使用动态树维护。复杂度$O(qn \log n)$

上述方法并没有挖掘题目性质。首先考虑权值的计算。我们有:

引理3.1. *在正边权树上，对一条直径的端点X, Y，有：*

$$\forall i, j, dis(i, j) \leq \max(dis(i, X), dis(i, Y))$$

Proof. 设p_i为直径XY上距离i最近的点，显然这个点唯一。
1.(i,j)过p_i，那么不妨设(p_i,j)与(p_i,X)不重合，则有：

$$\begin{aligned}dis(X,j) &= dis(X,p_i)+dis(p_i,j)\\ dis(X,Y) &= dis(X,p_i)+dis(p_i,Y)\\ dis(X,Y) &\ge dis(X,j)\\ \Longrightarrow dis(p_i,Y) &\ge dis(p_i,j)\\ dis(i,p_i)+dis(p_i,Y) &\ge dis(i,p_i)+dis(p_i,j)\\ dis(i,Y) &\ge dis(i,j)\end{aligned}$$

2.(i,j)不过p_i,则有：

$$\begin{aligned}dis(j,X) &= dis(i,j)+dis(i,p_i)+dis(p_i,X)\\ dis(X,Y) &= dis(X,p_i)+dis(p_i,Y)\\ dis(X,Y) &\ge dis(X,j)\\ \Longrightarrow dis(p_i,Y) &\ge dis(p_i,i)+dis(i,j)\\ dis(i,p_i)+dis(p_i,Y) &\ge dis(i,j)\\ dis(i,Y) &\ge dis(i,j)\end{aligned}$$

□

所以权值$v_i = \max(dis(i,X), dis(i,Y))$,其中$X,Y$是一条直径的端点。根据这一性质，我们可以得到：

定理3.1. 以v_x最小的x为根，则所有点的父亲权值都不大于它自身的权值，即$v_i \ge v_{fa_i}$。

Proof. 取XY中点c,有$v_i = dis(i,c)+\frac{dis(X,Y)}{2}$。以$c$为根，有$dis(i,c) \ge dis(fa_i,c)$。
当c是一点时，$c = x$。
当c是在一条边(a,b)上时，x是这条边的一个顶点。c为根的父子关系与x为根的关系只有(a,b)不同，而由于x是最小的，显然有$v_x \le v_{son_x}$。 □

因此，如果我们从大往小做双指针，实际上是没有分裂的，只会删去叶子结点，可以直接使用并查集维护权值。复杂度$O(n\log n+qn\alpha(n))$。