

2020/06/06——2020/06/12周报

团队训练

由于周末有数分小测和工图期末所以没有进行。

林星涵

专题

陶吟翔

专题

[树的直径和重心](#)

郭衍培

专题

本周推荐

林星涵：「2017 山东二轮集训 Day6」汇合

[传送门](#)

题目大意：求任意两点的最近公共祖先。

数据范围 n (点数) ≤ 900000 m (询问数) ≤ 100000 内存限制 4 MiB

解题思路：乍看之下 2s 时限，900000 的点数，可以用爬树法水过，但是由于内存限制感人，显然 $&nlogn&$ 的空间是会MLE的。因此，这题我们需要采用树上分块的思想，用块来统计信息爬树。只需处理好每一块深度最浅的点，每一块的上一块的编号，每一块在由块组成的这个新树中的深度等信息，别的操作基本上都与爬树法相同。

陶吟翔[FJOI2014]树的重心

[传送门](#)

题目大意：给出一棵有 n 个节点的树，问有多少个联通子树和这棵树的重心相同。

数据范围 $n \leq 200$ 多组数据。

解题思路：首先用 $O(n)$ 的时间求出重心，由于要计数所以我们考虑 DP 一棵树有可能有一个或两个重心。

如果有两个重心，说明无论以哪个为重心，子树的构成都是一样的，所以如果要选一个联通子树使重心与原来相同，必须在这两个重心的子树上选同样多的点。令 $f_{\{u,i\}}$ 表示在子树u里选*i*个点且包括u的方案数，那么有 $f_{\{u,i\}} = \sum_{v \in son_u} \sum_{j=1}^{\min(i-1, size_v)} f_{\{u,i-j\}} \times f_{\{v,j\}}$ 答案为 $\sum_{j=1}^{\min(size_x, size_y)} f_{\{g_x, i\}} \times f_{\{g_y, j\}}$ 如果只有一个重心，说明我们选的子树的最大值不能超过总数的二分之一，我们先算出 f 的值，然后定义 $F_{\{i,j\}}$ 为选了*i*个点，最大子树为*j*个点的方案数，初始有 $F_{\{i,i\}} = \sum_{v \in son_g} f_{\{v,i\}}$ 随后对于 $\forall k \leq i$ 有 $F_{\{i,\max(j,k)\}} = F_{\{i-j,k\}} \times f_{\{v,j\}}$ 答案为 $1 + \sum_{i=1}^n \sum_{j=1}^{n-2j} [F_{\{i,j\}}]$

郭衍培：

题目链接

题目大意：给定一个长度为n的序列A和一个长度为m的序列B保证B序列单增。将A分成m个子串，满足第i个子串最小值为 b_i 求满足要求的划分方案。答案模998244353

数据范围 $n,m \leq 2 \times 10^5$

解题思路：要利用好B单增的性质。由于 $b[j]$ 递增，所以A中最后一个等于 $b[j]$ 的元素一定在第j个子串中。维护一个后缀最小 $c[i] = \min_{k=i}^n a[k]$ 若 $c[1] < b[1]$ 结果显然为0。对于 $j > 1$ 只需看C中有多少个元素等于 $b[j]$ 每个等于 $b[j]$ 的元素之前都可以切开。下一次切开的地方，一定有 $c > b[j]$ 所以，最终答案就是所有等于 $b[j]$ 的元素个数的乘积。

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:hotpot:200606-200612&rev=1591961764>

Last update: 2020/06/12 19:36

