

# Summer Training Week 4

## 比赛简记

2020.08.01 2020牛客暑期多校训练营（第七场） pro 5/6/10 rank 27/???

2020.08.03 2020牛客暑期多校训练营（第八场） pro 4/5/11 rank 42/???

2020.08.06 2020-2021 BUAA ICPC Team Supplementary Training 02 pro 6/6/10 rank 4/???

## Max.D.

### 专题

复习了一下LCT

### 比赛

本周暂无

### 题目

[牛客第八场 A. All-Star Game](#)

[去年牛客第八场 E. Explorer](#)

两道都是一种类型的题目，无向图维护连通性，模板题还可以参考LOJ 121□去年不补题，欠下的一年之后还qwq□□

离线之后，可持久并查集（线段树分治□xsy下面有提到）和LCT都可以做，后者效率稍慢。

## Hardict

### 专题

### 比赛

### 题目

## MountVoom

### 专题

无

### 比赛

求求来点cf div1

### 题目

无

## 个人总结

### 陈铭煊 Max.D.

多练习，跟上队友思路。

### 龙鹏宇 Hardict

### 肖思炀 MountVoom

感觉有些浮躁，希望能静下心来，做题的时候更加集中一些。

## 本周推荐

### 陈铭煊 Max.D.

来源：

[牛客2020多校第七场 C. A National Pandemic](#)

一道动态点分治题，后来又学到了一种树链剖分的做法，两个 $\log$ 居然跑得比前者还要快。

标签：

动态点分治 树链剖分 树上距离的化简

题意：

$T$ 组数据，每组给出  $n$  个点组成的树，每个点有初始权值  $F(x)=0$  以及  $m$  个操作，操作分三种：

1. 给出点  $x$  和  $w$  为所有点增加权值  $w - \text{dist}(x, i)$   $\text{dist}(x, i)$  表示  $x$  和  $i$  的树上距离；
2. 更新点  $x$  处的权值为  $\min(F(x), 0)$ ；
3. 查询某个  $F(x)$

$1 \leq T \leq 5, 1 \leq n, m \leq 5 \times 10^4, 0 \leq w \leq 10000$  需要对操作 3 进行回答。

题解：

看到这种树上距离有关的更新，很容易想到了动态点分治，可惜没学到家，比赛时还仔细回忆了半天细节，最后还没写出来。

基本思路是，构造点分树，点分树上每个点实际上代表了一个树上连通区域，记录三个值  $S, dp, ddp$   $S[i]$  代表点分树上节点  $i$  的子树中，操作 1 被执行的总次数  $dp[i]$  表示点分树上子树  $i$  中每个点到  $i$  父亲  $pa[i]$  的距离之和（如果  $i$  为根，取  $pa[i]=i$ ） $ddp[i]$  表示点分树上子树  $i$  中每个点到  $i$  的距离之和。这三个值都可以在点分树上不断走根的过程中求出。

这个时候我们如果想知道操作 1 为  $F(x)$  带来的总的  $\text{dist}$  之和，可以从  $x$  开始，向上这样累加：

```
int x=u;
while (pa[u]) {
    ans += ddp[pa[u]] - dp[u] + Dis(pa[u], x) * (S[pa[u]] - S[u]);
    u = pa[u];
}
```

每次累加的值实际上是  $pa[u]$  的非  $u$  点分子树到  $x$  的总贡献，这个值由两部分组成，一部分是  $pa[u]$  在  $u$  这棵子树外的所有点，到达  $pa[u]$  产生的贡献和，也就是  $ddp[pa[u]] - dp[u]$  另一部分就是  $x$  到  $pa[u]$  这段路径长度产生的贡献——所有这些点都要走过这一段路。

注意一下  $ddp$  的值不一定等于儿子  $dp$  值的和哦。另外  $Dis$  函数可以用  $\text{RMQ}$  的方法求，这样就是真正  $O(m \log n)$  效率了。

现在既然距离的贡献都解决了，其他就很容易了，操作 2 额外记忆一个  $\delta$  即可。

其实要是明白透彻了，实现起来也不难。

第二种方法就显得很巧妙了：

对于操作 1，我们考虑一次修改对  $y$  来说会增加  $w - \text{dis}(x, y)$   $w - \text{dis}(x, y) = w - (\text{dep}(x) + \text{dep}(y) - 2 * \text{dep}(\text{lca})) = w - \text{dep}(x) - \text{dep}(y) + 2 * \text{dep}(\text{lca})$  所以，对于每次 1 操作，我们将其到根上所有点的  $\text{cnt} += 2$  询问的时候那部分就是求它到根的权值和。所以，树上路径加，路径查询，用树链剖分，再用数据结构来区间加，区间求和即可。

不得不说，这个式子很是传神。

评论：

好题

## 龙鹏宇 Hardict

来源：

标签：

题意：

题解：

评论：

## 肖思炀 MountVoom

来源：

[牛客第八场 A. All-Star Game](#)

标签：

线段树分治，并查集

题意：

球迷 $j$ 愿意观看球员 $i$ 的比赛需要满足 $\square j$ 是 $i$ 的粉丝或者 $j$ 和 $j'$ 都是 $i$ 的粉丝且 $j'$ 是 $i$ 的粉丝。

选择最少的球员进全明星赛，使得所有球迷都愿意观看。且有 $q$ 次粉丝关系的修改 $\square x$ 是 $y$ 的粉丝变为不是 $\square x$ 不是 $y$ 的粉丝变为是，每次修改都会询问。

题解：

把球员和粉丝相连，答案即为 $(n$ 个球员的连通分量个数) $-$  (孤立球员个数)，如果有球迷的度为 $0$ ，则答案为 $-1$ 。

把每个关系存在的时间划分成线段树上的 $\log$ 个区间，线段树每个节点维护了在这段时间存活的所有边，最后遍历一次线段树即可，因为回溯时需要删边，所以需要用到启发式合并的并查集。

最终复杂度为 $O(n \log^2 n)$

评论：

比赛做得太慢了以至于没有时间写这道题...

From:

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

Permanent link:

[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:weekly\\_digest\\_9&rev=1596790376](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:weekly_digest_9&rev=1596790376) 

Last update: **2020/08/07 16:52**