

## 原创 HDU 3974(Assign the task)线段树维护dfs序

2019-09-03 22:09:32    laaahu\_    阅读数 47    更多

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明，附上原文出处链接和本声明。

本文链接：<https://blog.csdn.net/laaahu/article/details/100528637>

题目：

[在这里啊](#)

题意：

一个公司有一个老板及许多员工，除老板外，每个员工都有自己的领导，然而每次分配工作之后，如果自己的领导得到了工作则自己也会去干相同的工作。每次询问要问一个一人在干什么工作。

思路：

最简单可以想到的就是单点维护单点查询，这个我帮你们试过了，妥妥超时，所以我们要优化一下，要区间更新，单点查询，因为每个人都有自己的领导，所以从公司的老板开始做深搜，这样子之后统计一下每个点的开始进入搜从这个点出去的点编号，就可以得到这点如果被更新时，相应的应该更新的所有点的编号都在一个区间内，只需要更新这个区间即可。

查询就是简单的单点查询，细节看代码注释吧。

```
1  /*用线段树维护关系树的dfs序，区间更新单点查询*/
2  #include <iostream>
3  #include <cstdio>
4  #include <cstring>
5  #include <algorithm>
6  #include <cmath>
7  using namespace std;
8  const int maxn=5e4+10;
9  struct node{
10     int b,e,v;
11 }t[maxn*4];
12 struct edge{
13     int u,v,next;
14 }e[maxn];
15 int head[maxn];    //广搜要建图，使用链式前向星
16 int per[maxn];     //记录每个点领导，主要是为了找出这个公司老板 开始广搜
17 int num[maxn];     //记录dfs顺序
18 int cont,esum;
19 int sta[maxn],en[maxn]; //记录一个点开始的 搜索点 以及最后一个搜索点
20 int vis[maxn];      //广搜标记数组，看是否被搜过。
21 void init(){        //初始化
22     for(int i=0;i<maxn;i++){
23         per[i]=i;
24         vis[i]=0;
25         head[i]=-1;
26     }
27     cont=esum=0;
28 }
29 void dfs(int x){     //dfs找到正确的dfs序
30     num[x]=++cont;
31     vis[x]=1;
32     sta[x]=cont;
33     for(int i=head[x];i!=-1;i=e[i].next){
34         if(!vis[e[i].v]){
35             dfs(e[i].v);
36         }
37     }
38     en[x]=cont;
39     return ;
40 }
41 void add_edge(int uu, int vv){ //链式前向星加边
42     e[++esum].u=uu;e[esum].v=vv;
43     e[esum].next=head[uu];
44     head[uu]=esum;
45 }
46 void push_down(int node){     //向下维护线段树
```

猿衣酷

专属于程序员的卫衣

关闭

```

47     if(t[node].v!=-1){
48         t[node<<1].v=t[node<<1|1].v=t[node].v;
49         t[node].v=-1;
50     }
51 }
52 void create(int bb,int ee,int node){    //建树
53     t[node].b=bb;
54     t[node].e=ee;
55     t[node].v=-1;
56     if(bb==ee)return ;
57     int mid=(bb+ee)>>1;
58     create(bb,mid,node<<1);
59     create(mid+1,ee,node<<1|1);
60 }
61 void update(int bb,int ee,int node,int val){    //更新
62     if(bb<=t[node].b&&ee>=t[node].e){
63         t[node].v=val;
64         return ;
65     }
66     int mid=(t[node].b+t[node].e)>>1;
67     push_down(node);
68     if(bb>mid)update(bb,ee,node<<1|1,val);
69     else if(ee<=mid)update(bb,ee,node<<1,val);
70     else{
71         update(bb,mid,node<<1,val);
72         update(mid+1,ee,node<<1|1,val);
73     }
74 }
75 int query(int pos,int node){    //查找
76     if(t[node].b==t[node].e){
77         return t[node].v;
78     }
79     int mid=(t[node].b+t[node].e)>>1;
80     push_down(node);
81     if(pos<=mid)return query(pos,node<<1);
82     else return query(pos,node<<1|1);
83 }
84 int main()
85 {
86     int t,n,m,cnt=0;
87     scanf("%d",&t);
88     while(t--){
89         scanf("%d",&n);
90         create(1,n,1);
91         init();
92         for(int i=0;i<n-1;i++){
93             {
94                 int u,v;
95                 scanf("%d%d",&u,&v);
96                 per[u]=v;
97                 add_edge(v,u);
98             }
99             for(int i=1;i<=n;i++){
100                 if(per[i]==i){
101                     dfs(i);
102                     break;
103                 }
104             }
105             scanf("%d",&m);
106             printf("Case %d:\n",++cnt);
107             for(int i=0;i<m;i++){
108                 char ch[2];
109                 scanf("%s",ch);
110                 if(ch[0]=='C'){
111                     int x;
112                     scanf("%d",&x);
113                     printf("%d\n",query(num[x],1));
114                 }
115                 if(ch[0]=='T'){
116                     int x,y;
117                     scanf("%d%d",&x,&y);
118                     update(sta[x],en[x],1,y);

```

```
118 |           }  
119 |       }  
120 |   }  
121 |}  
122 |
```

有 0 个人打赏

文章最后发布于: 201

猿衣酷

专属于程序员的卫衣

关闭