

C++算法

快快编程1836

快快编程
kkcoding.net

```

38 void dfs(ll u, ll fa, bool hvy){
39     for(ll i=hd[u]; i; i=nxt[i]){
40         ll v=to[i];
41         if(son[u]==v || fa==v) continue;
42         [redacted]
43     }
44     if(son[u]) [redacted]
45     for(ll i=hd[u]; i; i=nxt[i]){
46         ll v=to[i];
47         if([redacted]) continue;
48         aTree(v, 1);
49     }
50     aNode(u, 1);
51     ans[u]=res;
52     if(hvy) return;
53     [redacted]
54     mx=res=0;
55 }

```

```
25 void upd(ll c){
26     if(cnt[c]>mx)mx=cnt[c],res=c;
27     else if(cnt[c]==mx)res+=c;
28 }
29 void aNode(ll u,bool tag){
30     ll&c=clr[u];
31     if(tag)++cnt[c],upd(c);
32     else 
33 }
34 void aTree(ll u,bool tag){
35     for(int i=tI[u];i<=tO[u];++i)
36     {
37 }
```

讨论

若单个输入文件里
包含多组测试数据
需要清空哪些变量

nE, hd[],
timer,
mx, res, nClr,
dfs() 对根节点hvy设置0

快快编程1837

快快编程
kkcoding.net

暴力如何写作

快快编程
kkcoding.net

```

38 void dfs(ll u, ll fa, bool hvy){
39     for(ll i=hd[u]; i; i=nxt[i]){
40         ll v=to[i];
41         if(son[u]==v || fa==v) continue;
42         dfs(v, u, 0);
43     }
44     if(son[u]) dfs(son[u], u, 1);
45     for(ll i=hd[u]; i; i=nxt[i]){
46         ll v=to[i];
47         if(son[u]==v || fa==v) continue;
48         aTree(v, 1);
49     }
50     aNode(u, 1);
51     ans[u]=res;
52     if(hvy) return;
53     aTree(u, 0);
54     res=0;
55 }

```



```
25 void upd(ll c){
26     if(cnt[c]==k)res++;
27     else 
28 }
29 void aNode(ll u,bool tag){
30     ll&c=clr[u];
31     if(tag)++cnt[c],upd(c);
32     else --cnt[c];
33 }
34 void aTree(ll u,bool tag){
35     aNode(id[i],tag);
36 }
37 }
```

快快编程1838

快快编程
kkcoding.net

如何暴力?

每个深度用一个set
维护该层姓名集合

```
set<int> S[N*2];
```

```

46 void input(){
47     map<string,int>names;
48     int nNames=0;
49     scanf("%d",&n);
50     for(int u=1;u<=n;++u){
51         string s;
52         cin>>s;
53         if(names.count(s))iName[u]=names[s];
54         else iName[u]=names[s]=++nNames;
55         scanf("%d",&p[u]);
56         if(p[u])addedge(p[u],u);
57     }
58     scanf("%d",&m);
59     for(int i=1;i<=m;++i){
60         int v,k;
61         scanf("%d %d",&v,&k);
62         q[v].push_back((query){i,k});
63     }
64 }

```

```
65 void solve(){
66     for(int u=1;u<=n;++u)if(!p[u])dfs_pre(u);
67     for(int u=1;u<=n;++u)if(!p[u])dfs(u,0);
68     for(int i=1;i<m;++i)printf("%d ",ans[i]);
69     printf("%d\n",ans[m]);
70 }
```

```
34 void dfs(int u, bool hvy){
35     for(int i=hd[u]; i; i=nxt[i])
36         if(son[u]!=to[i]) dfs(to[i], 0);
37     if(son[u]) dfs(son[u], 1);
38     for(int i=hd[u]; i; i=nxt[i])
39         if(son[u]!=to[i]) aTree(to[i], 1);
40     aNode(u, 1);
41     for(int i=0; i<q[u].size(); ++i)
42         ans[q[u][i].id]=S[ ].size();
43
44
45 }
```

```
26 void aNode(int u, bool tag){  
27     if(tag) S[d[u]].insert(iName[u]);  
28     else   
29 }  
30 void aTree(int u, bool tag){  
31     for(int i=tI[u]; i<=tO[u]; ++i)  
32         aNode(id[i], tag);  
33 }
```

快快编程1839

快快编程
kkcoding.net

如何求k级祖先

如何暴力?

LA(u, k)

暴力

对于随机数据
单次查询平均复杂度 $O(\log n)$

简化
链状

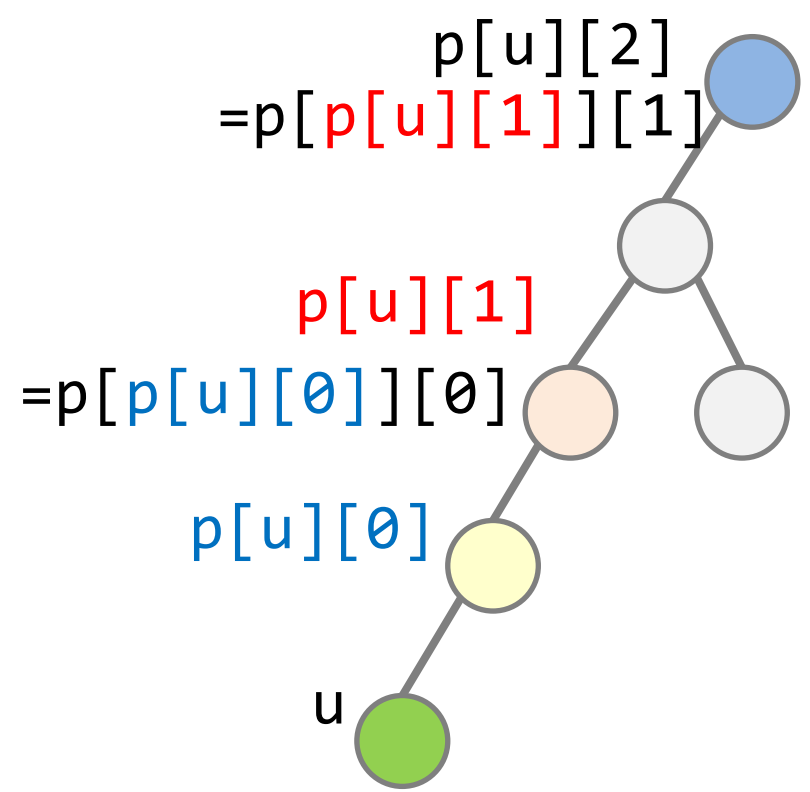
链状用一维数组储存节点序列
数组能快速索引

树链剖分

倍增

binary
lifting

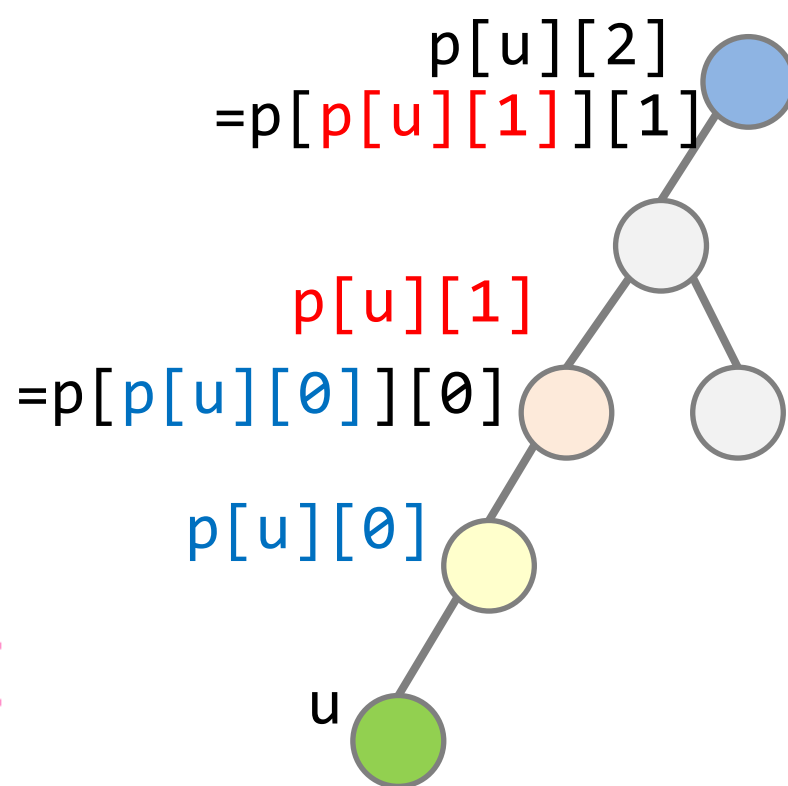
预处理ST表(稀疏表)
 $p[u][i]$ 代表 u 的第 2^i 个祖先



请写出 $p[u][i]$ 的递推式

```
L=log(n)/log(2)+1;
```

```
3  const int N=200009;  
4  vector<int> to[N];  
5  int L,n,q,p[N][20];  
6  void dfs(int u,int fa){  
7      p[u][0]=fa;  
8      for(int i=1;i<=L;++i)  
9          p[u][i]=p[p[u][i-1]][i-1];  
10     for(int i=0;i<to[u].size();++i)  
11         if(to[u][i]!=fa) dfs(to[u][i],u);  
12 }
```



```

13 int LA(int u, int k){
14     for(int i=0; i<=L; ++i)
15         if((1<<i)&k)
16             u=p[u][i];
17     return u;
18 }

```

$$k = (7)_{10} = (111)_2$$

$$(1 \ll 0) \& k \neq 0$$

$$(1 \ll 1) \& k \neq 0$$

$$(1 \ll 2) \& k \neq 0$$

```
60 scanf("%d",&m);
61 for(int i=1;i<=m;++i){
62     int v,k;
63     scanf("%d %d",&v,&k);
64     k=min(d[v]-1,k);
65     int u=LA(v,k);
66     q[u].push_back((query){i,k});
67 }
```

```
26 void aNode(int u, bool tag){
27     if(tag)++cnt[d[u]];
28     else 
29 }
30 void aTree(int u, bool tag){
31     for(int i=tI[u];i<=tO[u];++i)
32         aNode(id[i],tag);
33 }
```

```
34 void dfs(int u, bool hvy){
35     for(int i=hd[u]; i; i=nxt[i])
36         if(son[u]!=to[i]) dfs(to[i], 0);
37     if(son[u]) dfs(son[u], 1);
38     for(int i=hd[u]; i; i=nxt[i])
39         if(son[u]!=to[i]) aTree(to[i], 1);
40     aNode(u, 1);
41     for(int i=0; i<q[u].size(); ++i)
42         ans[q[u][i].id]=
43     if(hvy) return;
44     aTree(u, 0);
45 }
```


快快编程1840

快快编程
kkcoding.net

给定字母集合
能重组回文串的条件是什么？

出现奇数次的字母最多**1**个时
字母集合可以重组回文串

```
38 void dfs(int u, bool hvy){
39     for(int i=hd[u]; i; i=nxt[i])
40         if(son[u]!=to[i]) dfs(to[i], 0);
41     if(son[u]) dfs(son[u], 1);
42     for(int i=hd[u]; i; i=nxt[i])
43         if(son[u]!=to[i]) aTree(to[i]);
44     aNode(u);
45     for(int i=0; i<q[u].size(); ++i){
46         int iQ=q[u][i].id, d=q[u][i].d;
47         ans[iQ]=cOdd[d]<=1;
48     }
49     if(hvy) return;
50     aTree(u);
51 }
```

```
27 void aNode(int u){
28     int&d::d[u];
29     int&ltr::ltr[u];
30     cOdd[d]-=cnt[d][ltr];
31     cnt[d][ltr]^=1;
32     cOdd[d]+=cnt[d][ltr];
33 }
34 void aTree(int u){
35     for(int i=tI[u];i<=tO[u];++i)
36         aNode(id[i]);
37 }
```

小并大/启发式合并

子树信息

离线问询

快快编程作业

1837,1838,1839

拓展题

1840,1841,1842