Candyland

Difficulty:

Hard

Prerequisites:

Divide and Conquer(Theory)

Problem in Brief:

Given a tree where each node has a value associated with it. For each node, find the value that appears maximum number of times in its subtree.

Editorial:

To solve this problem, we will make use of the divide and conquery strategy known as smaller to larger.

Consider the example problem:

Given a tree, every vertex has color. Query is how many vertices in subtree of vertex v are colored with color c?

First, we have to calculate the size of the subtree of every vertice. It can be done with simple dfs:

Candyland

```
int sz[maxn];
void getsz(int v, int p){
    sz[v] = 1;  // every vertex has itself in its
subtree
    for(auto u : g[v])
        if(u != p){
            getsz(u, v);
            sz[v] += sz[u]; // add size of child u to
its parent(v)
        }
}
```

Now we have the size of the subtree of vertex v in sz[v].

```
map<int, int> *cnt[maxn];
void dfs(int v, int p){
  int mx = -1, bigChild = -1;
  for(auto u : g[v])
    if(u != p){
      dfs(u, v);
      if(sz[u] > mx)
        mx = sz[u], bigChild = u;
    }
  if(bigChild != -1)
```

```
        cnt[v] = cnt[bigChild];
    else
        cnt[v] = new map<int, int> ();
    (*cnt[v])[ col[v] ] ++;
    for(auto u : g[v])
      if(u != p && u != bigChild){
          for(auto x : *cnt[u])
              (*cnt[v])[x.first] += x.second;
      }
    //now (*cnt[v])[c] is the number of vertices in subtree of vertex v
that has color c. You can answer the queries easily.


}
```

The above code is the use of smaller to larger technique. Each color in any node can travel upward upto log(N) times as when it moves to a larger map the size becomes twice at the very least.

We can easily modify this code to solve our problem.

Time Complexity:

We have seen that each element at any node may move up at most log(N) times and and it takes log(N) time for each insertion as we use a map. Hence the Time Complexity is

O(N * log^2(N))

Candyland

Similar Problems:

[First]
[Second]
[Third]