Jump to Qs

Flag

Language: Help C++

```cpp
#include <bits/stdc++.h>
using namespace std;

#define ll long long#define int long long
#define pb push_back
#define all(x) x.begin(), x.end()
#define vi vector<int>
#define br cout << endl;


ll dfs(int i,int par, vi&timer, vi&timem,vi &value, vi g[], int t)
{
    // if(g[i].size() == 1 and i!= 1){

    //     int ans = 0;
    //     if(t == timer[i]){
    //         ans = value[i]/2;
    //     }else if(t < timer[i]){
    //         ans = value[i];
    //     }else if(t > timer[i]){
    //         ans = 0;
    //     }
    //     return ans;
    // }
    ll ans = 0;
    if(t == timer[i]){
        ans = value[i]/2;
    }else if(t < timer[i]){
        ans = value[i];
    }
    ll x = 0;
    for(auto child : g[i])
    {
        if(child != par)
        {
            x = max(x, dfs(child, i, timer, timem, value, g, t + 1));
        }
    }

    // cout << i << " " << ans+ x;br;
    return ans + x;
}

int  dfstime(int i,int par, vi &v, vi g[], int rpos)
{
    if(i == rpos)
    {
        v.pb(rpos);
        return 1;
    }

    v.pb(i);

    for(auto child : g[i])
    {
        if(child == par) continue;
        if(dfstime(child, i, v, g, rpos)){
            return 1;
        }
    }
    v.pop_back();
    return 0;
}
signed main() {

    int n, rpos; cin >> n >> rpos;
    vector<int> g[n+1];

    for(int i=0; i<n-1; i++){
        int u, v; cin >> u >> v;
        g[u].pb(v);
        g[v].pb(u);
    }

    vi value(n+1, 0);
    for(int i=1; i<=n; i++){
        cin>> value[i];
    }

    vector<int> timer(n+1, INT_MAX);

    vector<int> timem(n+1, -1); timem[1] = 0;

    int t = 0;

    vector<int> robberPath;
    dfstime(1, -1, robberPath, g, rpos);

    for(int i= robberPath.size() - 1; i>=0; i--){
        timer[robberPath[i]] = robberPath.size() - i-1;
    }


    // for(int i=0; i<timer.size(); i++){
    //     cout << timer[i] << " ";
    // }
    // br;
    // for(int i=0; i<robberPath.size(); i++){
    //     cout << robberPath[i] << " ";
    // }
    // br;

   cout << dfs(1,-1,  timer, timem, value, g, 0);br;

    return 0;
}
```

TEST CASE 1 + 13 Hidden (1 failed)

Test Case 1:

Result: Passed