# Model by VoidJackLee

## Header

#include<stdio.h>  
#include <stdlib.h>  
#include<iostream>  
#include <string.h>  
#include <math.h>  
#include <set>  
#include <map>  
#include <unordered\_map>  
#include <queue>  
#define rep(i,a,b) for (int i = a;i <= b;i ++)  
#define REP(i,a,b) for (int i = a;i < b;i ++)  
#define ll long long  
  
#define \_\_T int TEST;scanf("%d",&TEST);while(TEST --)  
  
using namespace std;  
  
#define sci(a) read(a)  
#define scii(a,b) sci(a);sci(b)  
  
const int MAXN = 1e6 + 10;  
unordered\_map<int,ll> mp;  
vector<int> vec[MAXN];  
vector<int> g[MAXN];  
ll cnt[MAXN];  
int vis[MAXN];  
int in[MAXN];  
int vv[MAXN];  
  
void read(int &a)  
{  
 char ch;  
 a = 0;  
 ch = getchar();  
 while (!isdigit(ch)) ch=getchar();  
 while (isdigit(ch)) a=a\*10+ch-'0',ch=getchar();  
}  
  
void bfs(int n) {  
 queue<int> q;  
 q.push(n);  
 int t;  
 while (!q.empty()) {  
 t = q.front();  
 q.pop();  
 for (auto i : g[t]) {  
 in[i] --;  
 cnt[i] += cnt[t];  
 if (!in[i]) q.push(i);  
 }  
 }  
}  
  
void dfs(int n)  
{  
 if (vv[n]) return;  
 vv[n] = 1;  
 for (auto i : g[n]) {  
 in[i] ++;  
 dfs(i);  
 }  
}  
  
int main()  
{  
 int n;  
 int t,k,tt;  
 int x,y;  
 ll v;  
 ll mx;  
 ll sum;  
 \_\_T {  
 sci(n);  
 rep(i,1,n) {  
 g[i].clear();  
 vv[i] = vis[i] = in[i] = cnt[i] = 0;  
 vec[i].clear();  
 }  
 mp.clear();  
 rep(i,1,n) {  
 sci(t);  
 if (t == 1) {  
 if (i == n) {  
 sci(k);  
 rep(ii,1,k) {  
 sci(tt);  
 mp[tt] ++;  
 }  
 vis[i] = 1;  
 } else {  
 sci(k);  
 rep(ii,1,k) {  
 sci(tt);  
 vec[i].push\_back(tt);  
 }  
 vis[i] = 1;  
 }  
 } else {  
 scii(x,y);  
 g[i].push\_back(x);  
 g[i].push\_back(y);  
 }  
 }  
  
 mx = 0;  
 sum = 0;  
 if (mp.size()) {  
 for (auto j : mp) {  
 mx = max(mx,j.second);  
 sum += j.second;  
 }  
 } else {  
 cnt[n] = 1;  
 dfs(n);  
 bfs(n);  
 rep(i,1,n-1) {  
 if (!cnt[i] || !vis[i]) continue;  
 // printf("..%d %d\n",i,cnt[i]);  
  
 for (auto j : vec[i]) {  
 v = mp[j] += cnt[i];  
 mx = max(mx,v);  
 sum += cnt[i];  
 }  
 }  
 // printf(">>%lld %lld\n",mx,sum);  
 }  
 printf("%lld\n", mx \* 2 <= sum ? sum : (sum - mx) \* 2);  
  
  
 // for (auto i : mp[n]) {  
 // printf("////>%d %lld\n",i.first,i.second);  
 // }  
 }  
 return 0;  
}  
/\*  
8  
1 1 1  
1 2 2 2  
2 1 2  
1 3 3 3 3  
2 3 4  
1 4 4 4 4 4  
2 3 6  
2 5 7  
\*/

### gcd

**ax + by = gcd(a,b)的通解为：**

* 可以求a的逆元:quickpow(a % mod,mod - 2)，注意a的范围，它也要%mod过才不会爆精度。

https://www.luogu.com.cn/problem/P1939