# Xushu

2286

#include<cstdio>

#include<cstring>

#include<algorithm>

#include<iostream>

#include<vector>

using namespace std;

const int M = 5e+5+123;

long long N = 12345678910111213;

struct Edge{

int to,val,next;

Edge(int to=0,int val=0,int next=0):to(to),val(val),next(next){}

}G[M];

int head[M];

int fa[M][20];

long long dist[M];

int dfn[M];

int deep[M];

int times;

int n,m;

int dfs(int num){

for(int i=head[num];i!=-1;i=G[i].next){

int to=G[i].to;

if(to==fa[num][0]) continue;

fa[to][0]=num;

deep[to]=deep[num]+1;

dist[to]=min(dist[num],(long long)G[i].val);

dfn[to]=++times;

dfs(to);

}

return 0;

}

int bin[22];

int lca(int x,int y){

if(deep[x]<deep[y])swap(x,y);

int tmp=deep[x]-deep[y];

for(int i=0;bin[i]<=tmp;i++)

if(tmp&bin[i])x=fa[x][i];

if(x==y)return x;

for(int i=18;i>=0;i--)

if(fa[x][i]!=fa[y][i])

x=fa[x][i],y=fa[y][i];

if(x==y)return x;

return fa[x][0];

}

int init(){

bin[0]=1;

for(int j=1;j<=19;j++){

bin[j]=bin[j-1]\*2;

for(int i=1;i<=n;i++){

fa[i][j]=fa[fa[i][j-1]][j-1];

}

}

return 0;

}

int cmp(int A,int B){

return dfn[A]<dfn[B];

}

int stk[M];

int c[M],t[M];

struct pos{

int to,next;

pos(int to=0,int next=0):to(to),next(next){}

}H[M];

int h[M],ht;

//重建新树

int work(){

stk[1]=1;

stk[2]=c[1];

int sz=2;ht=0;t[0]=0;

for(int i=2;i<=m;i++){

int father=lca(c[i],stk[sz]);

if(deep[stk[sz]]<=deep[father]){

if(stk[sz]!=c[i]) stk[++sz]=c[i];

continue;

}

while (1){

if (deep[stk[sz-1]]<=deep[father]){

ht++,H[ht].to=stk[sz],H[ht].next=h[father],h[father]=ht;

t[++t[0]]=father;

sz--;

if (stk[sz]!=father) stk[++sz]=father; break;

}

ht++,H[ht].to=stk[sz],H[ht].next=h[stk[sz-1]],h[stk[sz-1]]=ht;

t[++t[0]]=stk[sz-1];

sz--;

}

if(stk[sz]!=c[i]) stk[++sz]=c[i];

}

return 0;

}

long long Dp[M];

int vis[M];

int dp(int num){

if(vis[num]) {

Dp[num]=dist[num];

return 0;

}

long long sum=0;

for(int i=h[num];i!=-1;i=H[i].next){

dp(H[i].to);

sum+=Dp[H[i].to];

}

Dp[num]=min(sum,dist[num]);

return 0;

}

int read(){

int x=0; char ch=getchar();

while (ch<'0' || ch>'9') ch=getchar();

while (ch>='0' && ch<='9'){ x=x\*10+ch-'0'; ch=getchar(); }

return x;

}

int cnt;

int main(){

dist[1]=N;

deep[1]=1;

times=1;

n=read();

for(int i=1;i<=n;i++) head[i]=-1,h[i]=-1;

for(int i=1;i<n;i++){

int x,y,z;

x=read(),y=read(),z=read();

++cnt;

G[cnt].next=head[x];

G[cnt].to=y;

G[cnt].val=z;

head[x]=cnt;

++cnt;

G[cnt].next=head[y];

G[cnt].to=x;

G[cnt].val=z;

head[y]=cnt;

}

dfs(1),init();dfn[n+1]=n+1;fa[n+1][0]=1;deep[n+1]=2;

int T;

T=read();

while(T--){

for(int j=1;j<=t[0];j++)h[t[j]]=-1;

m=read();

for(int i=1;i<=m;i++){

c[i]=read();

vis[c[i]]=1;

}

sort(1+c,1+c+m,cmp);

c[++m]=n+1;

work();

dp(1);

printf("%lld\n",Dp[1]);

for(int i=1;i<=m;i++) vis[c[i]]=0;

}

return 0;

}