#include <iostream>

#include <vector>

using namespace std;

const int logN =20;

const int maxn=5e5+5;

void DFS(int u)

{

for (int v:E[u])

if (v!=Par[u][0])

{

H[v]=H[u]+1;

Par[v][0]=u;

DFS(v);

}

}

void prepare()

{

H[0]=-1;

DFS(1);

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

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

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

}

int LCA(int u,int v)

{

if (H[u]<H[v])

swap(u,v);

for (int i=19; i>=0; --i)

if (H[Par[u][i]]>=H[v])

u=Par[u][i];

if (u==v)

return u;

for (int i=19; i>=0; --i)

if (Par[u][i]!=Par[v][i])

{

u=Par[u][i];

v=Par[v][i];

}

return Par[u][0];

}

int main()

{

freopen("vd.inp","r",stdin);

cin >> n;

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

{

int u,v,w;

cin >> u >> v >> w;

E[u].push\_back({w,v});

E[v].push\_back({w,u});

}

prepare();

int test;

cin >> test;

while (test>0)

{

test--;

int u,v;

cin >> u >> v;

cout << LCA(u,v) << endl;

}

return 0;

}