原题：[推导部分和](https://www.lanqiao.cn/problems/2094/learning/)

思路1：要把数列中每一个元素看作其到前一个点的距离

思路2：求两点之间的距离，看它们是否在一个并集上，若在一个并集上，就可以通过它们到根节点的距离求出它们之间的距离。

#include<bits/stdc++.h>

#define ll long long

using namespace std;

class disjoinset{

    public:

    vector<ll> dist;//到父节点的距离

    vector<int> parent;

    disjoinset(int n)

    {

        dist.resize(n,0);

        parent.resize(n);

        for(int i=0;i<n;i++)

        parent[i]=i;

    }

    int find(int i)

    {

        if(parent[i]==i)

        return i;

        int zu=find(parent[i]);

        dist[i]+=dist[parent[i]];

        parent[i]=zu;

        return zu;

    }

    void merge(int l,int r,ll s){

        int pal=find(l);

        int par=find(r);

        if(pal==par)

        return;

        dist[pal]=dist[r]-s-dist[l];

        parent[pal]=par;

    }

};

int main(){

    int n,m,q;

    cin>>n>>m>>q;

    disjoinset st(n+1);

    while(m--)

    {

        ll l,r,s;

        cin>>l>>r>>s;

        st.merge(l-1,r,s);

    }

    while(q--)

    {

        ll l,r;

        cin>>l>>r;

        if(st.find(l-1)!=st.find(r))

        {

            cout<<"UNKNOWN"<<endl;

        }

        else

        {

            cout<<st.dist[r]-st.dist[l-1]<<endl;

        }

    }

}