

```
#include<bits/stdc++.h>
using namespace std;
typedef pair<int, int> P;
const int inf = numeric_limits<int>::max()/2;

struct edge{int from; int to; int cost;};

int v,e;
vector< edge > edges;
vector<int> dist;

// return : has negative cycle
bool bellman_ford(int s){
    dist.clear();
    dist.resize(v);
    fill(dist.begin(),dist.end(),inf);
    dist[s] = 0;
    for(int i = 0;i < v;++i){
        bool update = false;
        for(auto ed : edges){
            if(dist[ed.from] != inf &&
               dist[ed.from] + ed.cost < dist[ed.to]){
                update = true;
                dist[ed.to] = dist[ed.from] + ed.cost;
            }
        }
        if(!update)return false;
    }
    return true;
}

int main(void){
    int r;
    cin >> v >> e >> r;
    for(int i = 0;i < e;++i){
        int s,t,d;
        cin >> s >> t >> d;
        edges.push_back((edge){s,t,d});
    }
    if(bellman_ford(r)){
        puts("NEGATIVE CYCLE");
    }else{
        for(int i = 0;i < v;++i){
            if(dist[i] == inf)cout << "INF" << endl;
            else cout << dist[i] << endl;
        }
    }
    return 0;
}
```