

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

int v,e;
vector< vector<P> >graph;
vector< vector<ll> > dist;

void warshall_floyd(){
    dist.clear();
    dist.resize(v);
    for(int i = 0;i < v;++i){
        dist[i].resize(v);
        fill(dist[i].begin(),dist[i].end(),inf);
    }
    for(int i = 0;i < v;++i){
        dist[i][i] = 0;
    }
    for(int i = 0;i < v;++i){
        for(int j = 0;j < (int)graph[i].size();++j){
            int to = graph[i][j].second;
            ll cost = graph[i][j].first;
            dist[i][to] = cost;
        }
    }
    for(int k = 0;k < v;++k){
        for(int i = 0;i < v;++i){
            for(int j = 0;j < v;++j){
                if(dist[i][k] != inf && dist[k][j] != inf)
                    dist[i][j] = min(dist[i][j],
                                      dist[i][k] + dist[k][j]);
            }
        }
    }
}

int main(void){
    cin >> v >> e;
    graph = vector< vector<P> >(v);
    for(int i = 0;i < e;++i){
        int s,t;
        ll d;
        cin >> s >> t >> d;
        graph[s].push_back(P(d,t));
    }
    warshall_floyd();
    for(int i = 0;i < v;++i){
        if(dist[i][i] < 0){
            puts("NEGATIVE CYCLE");
            return 0;
        }
    }
    for(int i = 0;i < v;++i){
        for(int j = 0;j < v;++j){
            if(j != 0)cout << " ";
            if(dist[i][j] == inf)cout << "INF";
            else cout << dist[i][j];
        }
        cout << endl;
    }
    return 0;
}
```