

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef pair<int, ll> P;
const ll inf = numeric_limits<ll>::max()/2;
const int MAX_V = 10010;

int v,e;
vector< vector<int> > G;
vector< vector<int> > revG;
vector<int> vs;
int visit[MAX_V];
int cmp[MAX_V];

void add_edge(int from,int to){
    G[from].push_back(to);
    revG[to].push_back(from);
}

void dfs(int n){
    visit[n] = 1;
    for(int i : G[n]){
        if(!visit[i]) dfs(i);
    }
    vs.push_back(n);
}

void rdfs(int n,int k){
    visit[n] = 1;
    cmp[n] = k;
    for(int i : revG[n]){
        if(!visit[i]) rdfs(i,k);
    }
}

//ret :: num of cmp;
int scc(){
    fill(visit,visit + MAX_V, 0);
    vs.clear();
    for(int i = 0;i < v;++i){
        if(!visit[i])dfs(i);
    }
    fill(visit,visit + MAX_V, 0);
    int k = 0;
    for(int i = (int)vs.size() - 1;i >= 0;--i){
        if(!visit[vs[i]])rdfs(vs[i],k);
        ++k;
    }
    return k;
}

int main(void){
    cin >> v >> e;
    G.clear();revG.clear();
    G.resize(v);revG.resize(v);
    for(int i = 0;i < e;++i){
        int s,t;
        cin >> s >> t;
        add_edge(s,t);
    }
    scc();
    int q;
    cin >> q;
    for(int i = 0;i < q;++i){
        int u,v;
        cin >> u >> v;
        if(cmp[u] == cmp[v])puts("1");
        else puts("0");
    }
    return 0;
}
```

```
}
```