#include<bits/stdc++.h>

using namespace std;

vector<vector<int> > graph;

void toposort(int node, vector<int> &visited, stack<int> &st){

visited[node] = 1;

for(auto x: graph[node]){

if(!visited[x]){

toposort(x,visited,st);

}

}

st.push(node);

}

void dfs(int node, vector<int> &visited, vector<vector<int>> &transpose){

cout<<node<<" ";

visited[node] = 1;

for(auto x: transpose[node]){

if(!visited[x]){

dfs(x,visited, transpose);

}

}

}

int main(){

int n,m;

cin>>n>>m;

graph.resize(n+1);

while(m--){

int x,y;

cin>>x>>y;

graph[x].push\_back(y);

}

stack<int> st;

vector<int> visited(n+1,0);

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

if(!visited[i]){

toposort(i,visited,st);

}

}

vector<vector<int> > transpose(n+1);

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

visited[i] = 0;

for(auto x: graph[i]){

transpose[x].push\_back(i);

}

}

while(st.size()!=0){

int node = st.top();

st.pop();

if(!visited[node]){

dfs(node,visited,transpose);

cout<<endl;

}

}

}