#include<bits/stdc++.h>

using namespace std;

vector<vector<int> > graph;

bool cycle\_dfs(int src, vector<bool> &visited, vector<bool> &dfs\_v){

// if(visited[src] && dfs\_v[src]){

// return true;

// }

visited[src] = true;

dfs\_v[src] = true;

for(auto x: graph[src]){

if(visited[x]==false){

bool res = cycle\_dfs(x,visited,dfs\_v);

if(res==true){

return true;

}

}

else if(dfs\_v[x]){

return true;

}

}

dfs\_v[src] = false;

return false;

}

int main(){

int n;

int e;

cin>>n>>e;

graph.resize(n);

for(int i=0;i<e;i++){

int x,y;

cin>>x>>y;

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

}

vector<bool> visited(n,false);

vector<bool> dfs\_v(n,false);

for(int i=0;i<n;i++){

if(visited[i]==false){

bool ans = cycle\_dfs(i,visited,dfs\_v);

if(ans){

cout<<"true";

return 0;

}

}

}

cout<<"false";

}