**Check Bipartite Graph**

#include<bits/stdc++.h>

bool isGraphBirpatite(vector<vector<int>> &edges) {

// Write your code here.

vector<vector<int>> adj(edges.size());

for(int i=0;i<edges.size();i++){

for(int j=0;j<edges[0].size();j++){

if(edges[i][j]){

adj[i].push\_back(j);

adj[j].push\_back(i);

}

}

}

vector<int> color(edges.size(),0);

for(int i=0;i<edges.size();i++){

if(color[i]) continue;

queue<int> q;

q.push(i);

color[i]=1;

while(!q.empty()){

auto front =q.front();

q.pop();

for(auto child:adj[front]){

if(color[child]==0)

{

color[child]=-color[front];

q.push(child);

}

else{

if(color[child]==color[front]) return false;

}

}

}

}

return true;

}