Research **queue**which is implemented in C library at <http://www.cplusplus.com/reference/queue/queue/>. You can use library **queue**in c++ for this question.

Using **queue**, complete function **bool isBipartite(vector<vector<int>> graph)** to determine if a graph is bipartite or not (the graph can be disconnected). In caat <https://en.wikipedia.org/wiki/Bipartite_graph>.

You can use below liberaries in this question.

#include <iostream>

#include <vector>

#include <queue>

**For example:**

| **Test** | **Result** |
| --- | --- |
| int G[6][6] = { {0, 1, 0, 0, 0, 1},  {1, 0, 1, 0, 0, 0},  {0, 1, 0, 1, 0, 0},  {0, 0, 1, 0, 1, 0},  {0, 0, 0, 1, 0, 1},  {1, 0, 0, 0, 1, 0} };  int n = 6;  vector<vector<int>> graph(n, vector<int>());  for (int i = 0; i < n; ++i) {  for (int j = 0; j < n; ++j) {  if (G[i][j]) graph[i].push\_back(j);  }  }  isBipartite(graph) ? cout << "Yes" : cout << "No"; | Yes |