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 **void bfs(vector<vector<int>> graph, int start)** to traverse all the nodes of the graph from given start node using Breadth First Search algorithm and data structure **queue**, and print the order of visited nodes.

You can use below liberaries in this question.

#include <iostream>

#include <vector>

#include <queue>

**For example:**

| **Test** | **Result** |
| --- | --- |
| int init\_graph[10][10] = { {0, 1, 1, 0, 1, 0, 1, 0, 1, 0},  {0, 0, 1, 1, 0, 0, 0, 1, 0, 0},  {0, 1, 0, 0, 0, 1, 1, 0, 1, 1},  {1, 0, 0, 0, 0, 0, 0, 1, 0, 0},  {0, 1, 0, 0, 0, 0, 0, 1, 0, 0},  {1, 0, 1, 0, 1, 0, 0, 0, 1, 0},  {0, 0, 1, 1, 0, 1, 0, 0, 0, 0},  {1, 0, 0, 0, 0, 1, 1, 0, 1, 0},  {0, 0, 0, 0, 0, 1, 0, 1, 0, 1},  {1, 0, 1, 0, 1, 0, 0, 0, 1, 0} };  int n = 10;  vector<vector<int>> graph(n, vector<int>());  for (int i = 0; i < n; ++i) {  for (int j = 0; j < n; ++j) {  if (init\_graph[i][j]) graph[i].push\_back(j);  }  }  bfs(graph, 0); | 0 1 2 4 6 8 3 7 5 9 |