# Lab 10

Problem Description:

   Given a Directed Graph and two vertices in it, write a Java program to check if there is a path from the first given vertex to second. For example, in the following graph, there is a path from vertex 1 to 3. As another example, there is no path from 3 to 0. You can either use Breadth First Search (BFS) or Depth First Search (DFS) to find a path between two vertices. Take the first vertex as source in BFS (or DFS), follow the standard BFS (or DFS). If you see the second vertex in your traversal, then return true. Else return false.
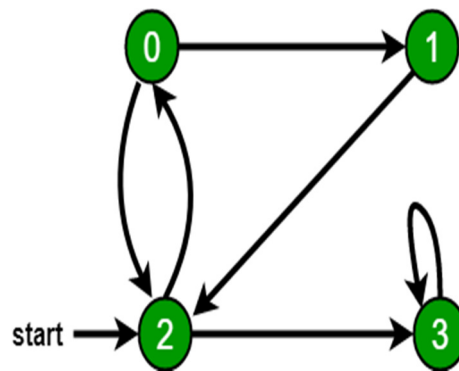
Input: The first line of input n indicates the number of vertices. The second line of input m is the number of edges in the graph and the following m lines are the directed edges in the graph. The next integer p is number of testing inputs and the next p lines will be the test samples. The vertex numbers start from 0.

Output: The output consists of one line of p strings "YES" or "NO" indicating if there is a path between the vertices or not. There is a single space between the output strings.

Test Case	Input	Output
1	4 6 0 1 0 2 1 2 2 0 2 3 3 3 2 1 3 3 1	YES NO