```java
import java.util.Scanner;

public class HamiltonianCycle {
    private int V, pathCount;
    private int[] path;
    private int[][] graph;

    /** Function to find cycle **/
    public void findHamiltonianCycle(int[][] g) {
        V = g.length;
        path = new int[V];

        Arrays.fill(path, -1);
        graph = g;
        try {
            path[0] = 0;
            pathCount = 1;
            solve(0);
            System.out.println("No solution");
        }
        catch (Exception e) {
            System.out.println(e.getMessage());
            display();
        }
    }

    /** function to find paths recursively **/
    public void solve(int vertex) throws Exception {
        /** solution **/
        if (graph[vertex][0] == 1 && pathCount == V)
            throw new Exception("Solution found");
        /** last vertex not linked to 0 **/
        if (pathCount == V)
            return;

        for (int v = 0; v < V; v++) {
            if (graph[vertex][v] == 1 ) { //if connected
                path[pathCount++] = v; //add to path
                graph[vertex][v] = 0;
                graph[v][vertex] = 0;

```

```
43                  // if vertex not already selected solve recursively
44                  if (!isPresent(v))
45                      solve(v);
46
47                  graph[vertex][v] = 1; // restore connection
48                  graph[v][vertex] = 1;
49                  path[--pathCount] = -1; // remove path
50              }
51          }
52      }
53
54      // function to check if path is already selected
55      public boolean isPresent(int v) {
56          for (int i = 0; i < pathCount - 1; i++)
57              if (path[i] == v)
58                  return true;
59          return false;
60      }
61
62      public void display() {
63          System.out.print("\nPath : ");
64          for (int i = 0; i <= V; i++)
65              System.out.print(path[i % V] +" ");
66          System.out.println();
67      }
68
69      public static void main (String[] args) {
70          Scanner scan = new Scanner(System.in);
71          System.out.println("HamiltonianCycle Algorithm Test\n");
72          HamiltonianCycle hc = new HamiltonianCycle();
73          System.out.println("Num of vertices:\n");
74          int V = scan.nextInt();
75          System.out.println("\nEnter matrix\n");
76          int[][] graph = new int[V][V];
77          for (int i = 0; i < V; i++)
78              for (int j = 0; j < V; j++)
79                  graph[i][j] = scan.nextInt();
80          hc.findHamiltonianCycle(graph);
81      }
82  }
```