```java
import java.util.*;

public class TopologicalSort {
    private Stack<Integer> stack;
    public TopologicalSort() {
        stack = new Stack<Integer>();
    }

    public int[] topological(int matrix[][], int s) throws
 NullPointerException {
        int numNodes = matrix[s].length - 1;
        int[] topological_sort = new int[numNodes + 1];
        int pos = 1;
        int j;
        int visited[] = new int[numNodes + 1];
        int element = s;
        int i = s;
        visited[s] = 1;
        stack.push(s);

        while (!stack.isEmpty()) {
            element = stack.peek();
            while (i <= numNodes) {
                if (matrix[element][i] == 1 && visited[i] == 1) {
                    if (stack.contains(i)) {
                        System.out.println("NOT POSSIBLE");
                        return null;
                    }
                }
                if (matrix[element][i] == 1 && visited[i] == 0) {
                    stack.push(i);
                    visited[i] = 1;
                    element = i;
                    i = 1;
                    continue;
                }
                i++;
            }
            j = stack.pop();
            topological_sort[pos++] = j;
            i = ++j;
```

```
41          }
42          return topological_sort;
43      }
44
45      public static void main(String...arg) {
46          int numNodes, source;
47          Scanner scanner = null;
48          int tsort[]  = null;
49          try {
50              System.out.println("Enter numNodes:");
51              scanner = new Scanner(System.in);
52              numNodes = scanner.nextInt();
53
54              int adjacency_matrix[][] = new int[numNodes + 1]
    [numNodes + 1];
55              System.out.println("Enter the adjacency matrix");
56
57              for (int i = 1; i <= numNodes; i++)
58                  for (int j = 1; j <= numNodes; j++)
59                      adjacency_matrix[i][j] = scanner.nextInt();
60
61              System.out.println("Enter source:");
62              source = scanner.nextInt();
63
64              System.out.println("Topological sort: ");
65              TopologicalSort toposort = new TopologicalSort();
66              tsort = toposort.topological(adjacency_matrix, source);
67              System.out.println();
68              for (int i = tsort.length - 1; i > 0; i--) {
69                  if (tsort[i] != 0)
70                      System.out.print(tsort[i]+"\t");
71              }
72          }
73          catch(InputMismatchException inputMismatch) {
74               System.out.println("Wrong Input format");
75          }
76          catch(NullPointerException nullPointer) {
77          }
78          scanner.close();
79      }
80  }
```