```
import java.util.Arrays;
import java.util.Iterator;
import java.util.LinkedList;
import java.util.Stack;


public class topologicalSort {

    public static void main(String[] args) {
        Graph g = new Graph(6);
        g.add_edge(5, 2);
        g.add_edge(5, 0);
        g.add_edge(4, 0);
        g.add_edge(4, 1);
        g.add_edge(2, 3);
        g.add_edge(3, 1);

        System.out.println("Topological Sort Order :");
        g.topological_sort();
    }


    static class Graph {
        private int V;
        private LinkedList<Integer>[] adj;

        public Graph(int v) {
            V = v;
            adj = new LinkedList[v];

            for (int i = 0; i < v; i++)
                adj[i] = new LinkedList<>();
        }

        void add_edge(int v, int w) {
            adj[v].add(w);
        }

        void ts(int v, boolean[] vis, Stack<Object> s) {
            vis[v] = true;
            Integer i;
            Iterator<Integer> it = adj[v].iterator();

            while (it.hasNext()) {
                i = it.next();
                if (!vis[i])
                    ts(i, vis, s);
            }
            s.push(v);
        }

        void topological_sort() {
            Stack<Object> s = new Stack<>();
            boolean[] vis = new boolean[V];
            Arrays.fill(vis, false);

            for (int i = 0; i < V; i++)
                if (!vis[i])
                    ts(i, vis, s);

            while (!s.isEmpty())
                System.out.println(s.pop() + " ");
        }

    }


}
```