Program:

```
import java.util.ArrayList;
import java.util.HashMap;
import java.util.HashSet;
import java.util.List;
import java.util.Map;
import java.util.Scanner;
import java.util.Set;

class Process {
  int id;
  List<Integer> dependencies;
  boolean isBlocked;

  Process(int id) {
    this.id = id;
    this.dependencies = new ArrayList<>();
    this.isBlocked = false;
  }

  void addDependency(int processId) {
    dependencies.add(processId);
  }
}

class DeadlockDetection {
  private final Map<Integer, Process> processes;

  DeadlockDetection() {
    this.processes = new HashMap<>();
  }

  void addProcess(int id) {
    processes.put(id, new Process(id));
  }

  void addDependency(int from, int to) {
    if (processes.containsKey(from) && processes.containsKey(to)) {
      processes.get(from).addDependency(to);
      processes.get(from).isBlocked = true;
    }
  }

  boolean detectDeadlock(int initiator) {
    Set<Integer> visited = new HashSet<>();
    return detectCycle(initiator, initiator, visited);
  }

  private boolean detectCycle(int current, int initiator, Set<Integer> visited) {
    if (!processes.containsKey(current))
      return false;
    if (visited.contains(current))
      return current == initiator;

    visited.add(current);
    for (int dependent : processes.get(current).dependencies) {
      if (detectCycle(dependent, initiator, visited)) {
        return true;
      }
    }
    visited.remove(current);
    return false;
  }
}
```

```
public class ChandyMisraHaas {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    DeadlockDetection detector = new DeadlockDetection();

    System.out.print("Enter number of processes: ");
    int n = scanner.nextInt();

    for (int i = 1; i <= n; i++) {
      detector.addProcess(i);
    }

    System.out.print("Enter number of dependencies: ");
    int d = scanner.nextInt();

    System.out.println("Enter dependencies (from to):");
    for (int i = 0; i < d; i++) {
      int from = scanner.nextInt();
      int to = scanner.nextInt();
      detector.addDependency(from, to);
    }

    System.out.print("Enter initiator process for deadlock detection: ");
    int initiator = scanner.nextInt();

    if (detector.detectDeadlock(initiator)) {
      System.out.println("Deadlock detected!");
    } else {
      System.out.println("No deadlock detected.");
    }
    scanner.close();
  }
}
```

Output:

```
PS C:\Users\Moin MN\Downloads> java .\ChandyMisraHaas.java
Enter number of processes: 4
Enter number of dependencies: 4
Enter dependencies (from to):
1 2
2 3
3 4
4 1
Enter initiator process for deadlock detection: 1
Deadlock detected!
```

```
PS C:\Users\Moin MN\Downloads> java .\ChandyMisraHaas.java
Enter number of processes: 4
Enter number of dependencies: 3
Enter dependencies (from to):
1 2
2 3
3 4
Enter initiator process for deadlock detection: 1
No deadlock detected.
```