```
id: 0
                         frame.node.setIndex(frame.g.time);
                       frame.node.setLowLink(frame.g.time);
                                 frame.g.time += 1;
                          frame.g.stack.push(frame.node);
                            frame.node.setInStack(true);
            frame.iterator = frame.g.getNeighbours(frame.node).iterator();
                                  frame.block = 1:
                                        break:
                                        id: 1
                            if (frame.iterator.hasNext()) {
                           frame.n = frame.iterator.next();
                             if (!frame.n.wasVisited()) {
                  stack.push(new DfsCTCFrame(frame.g,frame.n));
                                  frame.block = 7;
                                        break:
                                       } else {
                               if (frame.n.isInStack()) {
  frame.node.setLowLink(Math.min(frame.node.getLowLink(), frame.n.getIndex()));
                                  frame.block = 1:
                                        break:
          }} else { if (frame.node.getLowLink() == frame.node.getIndex()) {
                      final List<Node> ctc = new ArrayList<>();
                                       Node n:
                                         do {
                               n = frame.g.stack.pop();
                                 n.setInStack(false);
                                      ctc.add(n):
                           } while (!n.equals(frame.node));
                                 frame.g.ctc.add(ctc);
                                        return;
                                        id: 7
frame.node.setLowLink(Math.min(frame.node.getLowLink(), frame.n.getLowLink()));
                                  frame.block = 1;
                                       break;
```