```
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
import java.util.Queue;

public class RaymondTree {
    static List<Node> list = new ArrayList<>();
    public static void main(String[] args) {
        int n = 8;
        Node n1 = new Node(1,1);
        Node n2 = new Node(2,1);
        Node n3 = new Node(3,1);
        Node n4 = new Node(4,2);
        Node n5 = new Node(5,2);
        Node n6 = new Node(6,3);
        Node n7 = new Node(7,3);
        Node n8 = new Node(8,3);

list.add(n1);list.add(n2);list.add(n3);list.add(n4);list.add(n5);list.add(n6);list.add(n7);list.add(n8);
        List<Integer> req=new LinkedList<>();
        req.add(8);//req.add(6);req.add(2);
        System.out.println("Initially");
        printFunction();
        for(int a:req)
        {
            System.out.println("--------- Process "+a+" -----------");
            request(a,a);
        }
    }

    private static void request(int holder,int req){
        Node n = list.get(holder-1); //holder = 3 => 2nd node in the list
        n.q.add(req);//8
        System.out.println("Queue of "+n.id);
        for(Integer i:n.q)
            System.out.print(i+" ");
        System.out.println("");
        if(n.holder!=n.id)
            request(n.holder, n.id); //3,8...1,3
        else
            giveToken(n.id);//1
    }

    private static void giveToken(int nid){ Node n = list.get(nid-1);
```

```
        int next = n.q.remove();//3
        n.holder = next;
        if(next!=nid)
            giveToken(next);
        else
            printFunction();
    }

    private static void printFunction(){
        for(Node n:list){
            System.out.println("id: "+ n.id+"  Holder: "+ n.holder+" ");
        }
    }
}

class Node{
    int id;
    int holder;
    Queue<Integer> q;
    Node(int id, int holder){
        this.id = id;
        this.holder = holder;
        this.q = new LinkedList<>();
    }
}
```