Raymond’s Algorithm

Node State:
- Holder: self or neighbor (Indicates direction of token.)
- FIFO queue (Each element is self or a single neighbor. No duplicates.)
- asked: boolean (Indicates whether a request has been sent to current Holder.)

To request token:
1. If Holder = self and queue is empty
   - Enter Critical Section
   - asked = false
2. Else
   - Add self to queue
   - If asked = false
     - Send request to Holder
     - asked = true

On receiving request from neighbor i:
1. If Holder = self
   - If not in Critical Section and queue is empty
     - Send token to node i
     - Holder = i
     - asked = false
   - Else
     - enqueue(request)
2. Else (Holder ≠ self)
   - Add request to queue
   - If asked = false
     - Send request to Holder
     - asked = true

On receiving token:
1. req = dequeue
2. If req = self
   - Holder = self
   - asked = false
   - enter Critical Section
3. Else
   - Send token to req
   - Holder = req
   - If queue not empty
     - Send request to Holder
     - asked = true

On exiting Critical Section:
1. If queue not empty
   - req = dequeue
   - Send token to req
   - Holder = req
   - Asked = false
   - If queue not empty
     - Send request to holder
     - Asked = true