RAMAIAH
Institute of Technology

M.S. Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)
Department of Computer Science and Engineering

Course Name: Distributed Systems
Course Code: CSE20/CSE751
Credits: 3:0:0

Term: Oct 2021-Feb 2022

Faculty:
Sini Anna Alex

Raymond Tree Based Algorithm

Based on Spanning Tree

- Sites are arranged in a logical directed tree. Root: token holder. Edges: directed towards root.
- Every site has a variable holder that points to an immediate neighbour node, on the directed path towards root. (Root's holder point to itself).
- The Raymond Tree follows
  - ✓ Requesting the CS
  - ✓ Executing the CS
  - ✓ Releasing the CS

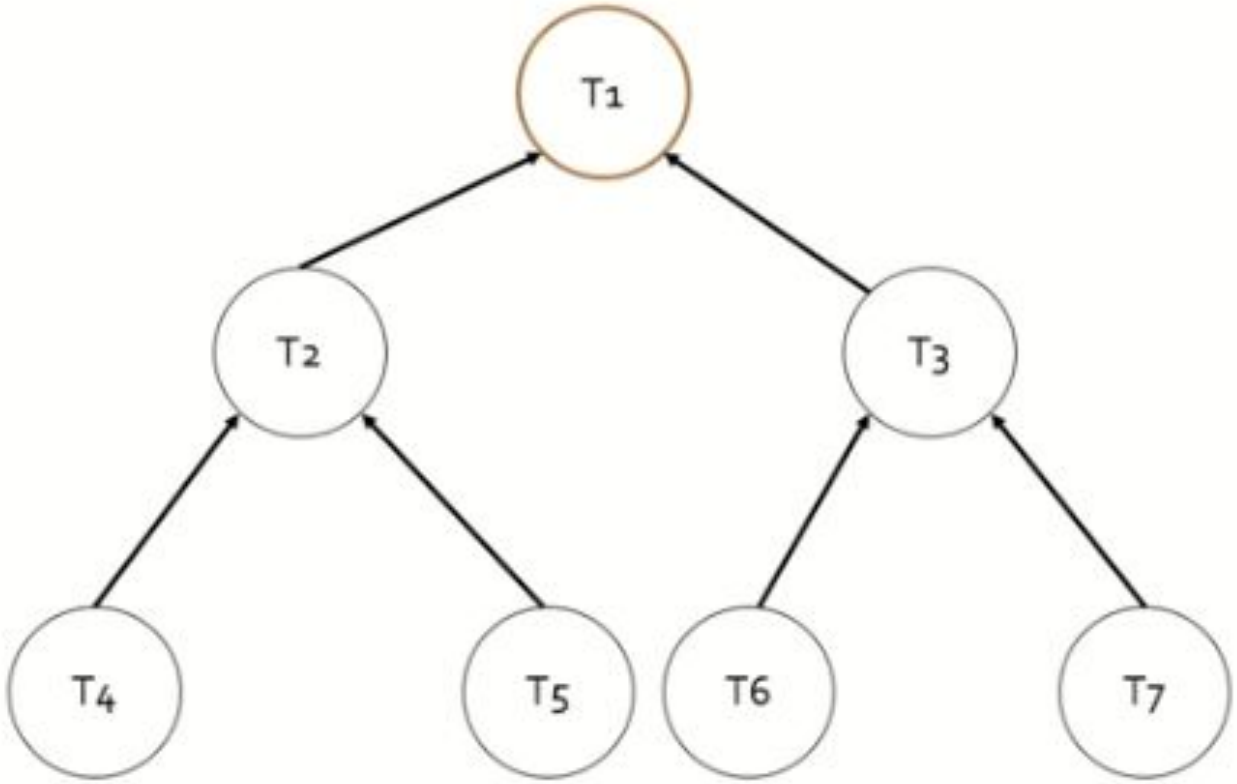
Raymond Tree Based Algorithm

- **R**equesting the CS
  - ✓ If Si does not hold token and request CS, sends REQUEST upwards provided its request_q is empty. It then adds its request to request_q.
  - ✓ Non-empty request_q → REQUEST message for top entry in q.
  - ✓ Site on path to root receiving REQUEST → propagate it up, if its request_q is empty. Add request to request_q.
  - ✓ Root on receiving REQUEST → send token to the site that forwarded the message. Set holder to that forwarding site.
  - ✓ Any Si receiving token, delete top entry from request_q, send token to that site, set holder to point to it. If request_q is non-empty now, send REQUEST message to the holder site.

Raymond Tree Based Algorithm

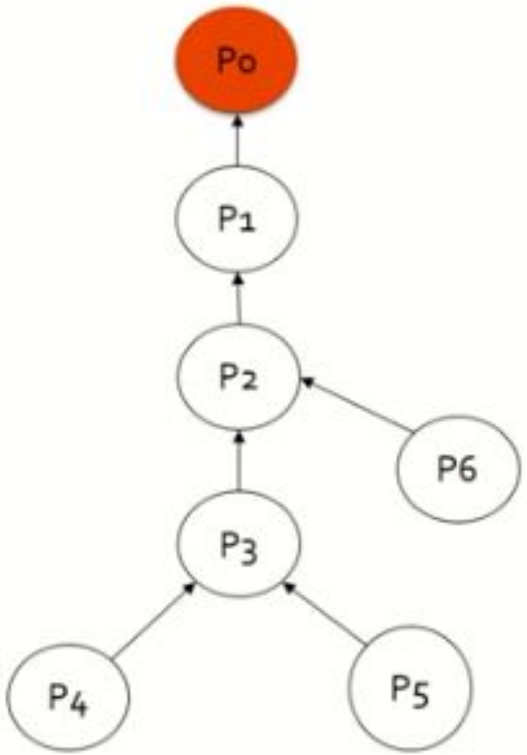
- **E**xecuting the CS: getting token with the site at the top of request_q. Delete top of request_q, enter CS.
- **R**eleasing the CS
  - ✓ If request_q is non-empty, delete top entry from q, send token to that site, set holder to that site.
  - ✓ If request_q is non-empty now, send REQUEST message to the holder site.

Raymond Tree Based Algorithm



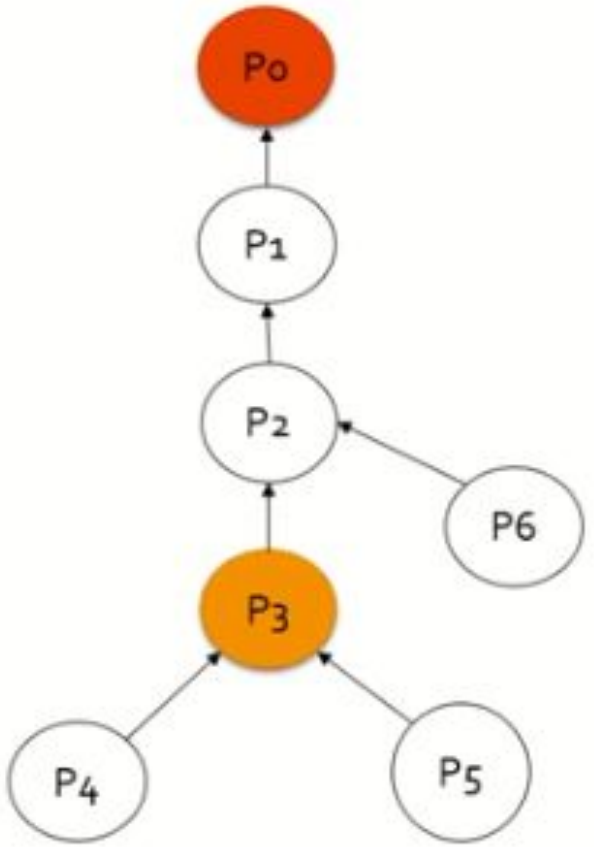
Raymond Tree Based Algorithm

- **P0** is the current node and holds the token.



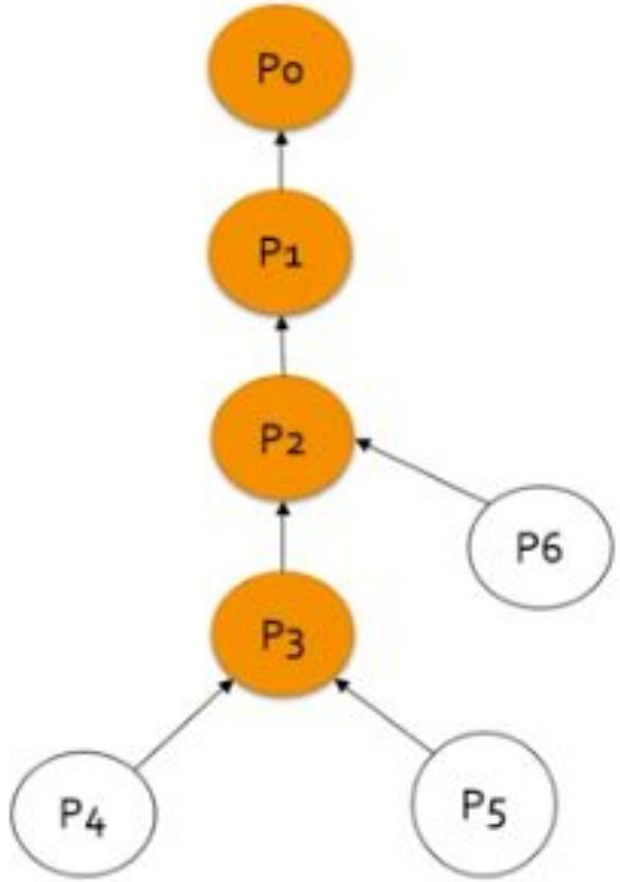
Raymond Tree Based Algorithm

- **P3** wants the token to get into CS.
- **P3** is added to the FIFO queue and request message is sent to parent P2.

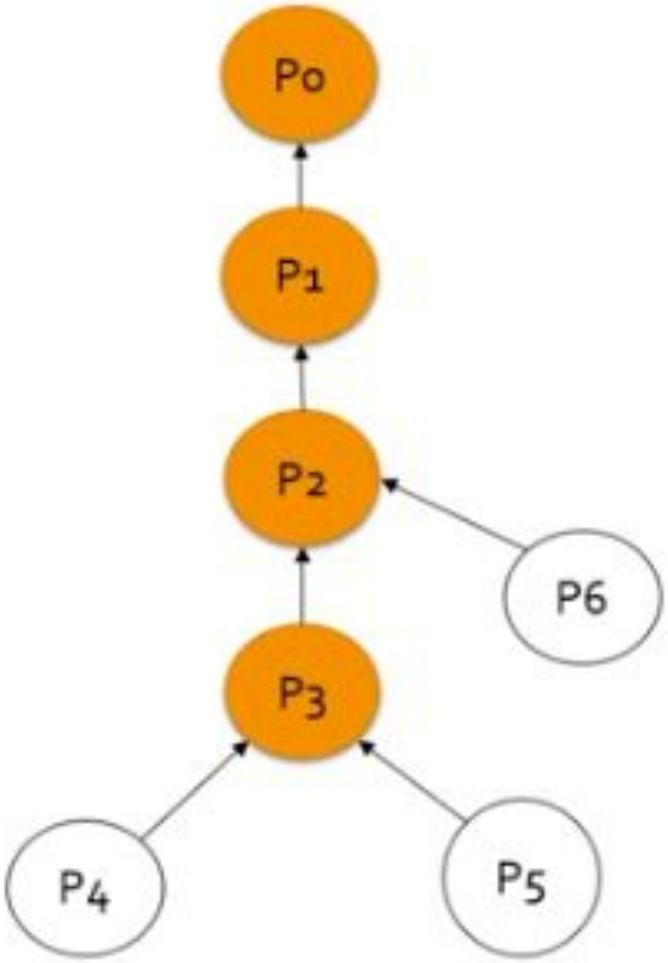


Raymond Tree Based Algorithm

- **P3** wants the token to get into CS.
- **P3** is added to the FIFO queue and request message is sent to parent P2.
- **P2** receives the REQUEST from P3.
- **P3** is added to the FIFO queue of P2 and request message is sent to parent P1.
- **P1** receives the REQUEST from P3.
- **P3** is added to the FIFO queue of P1 and request message is sent to parent P0.

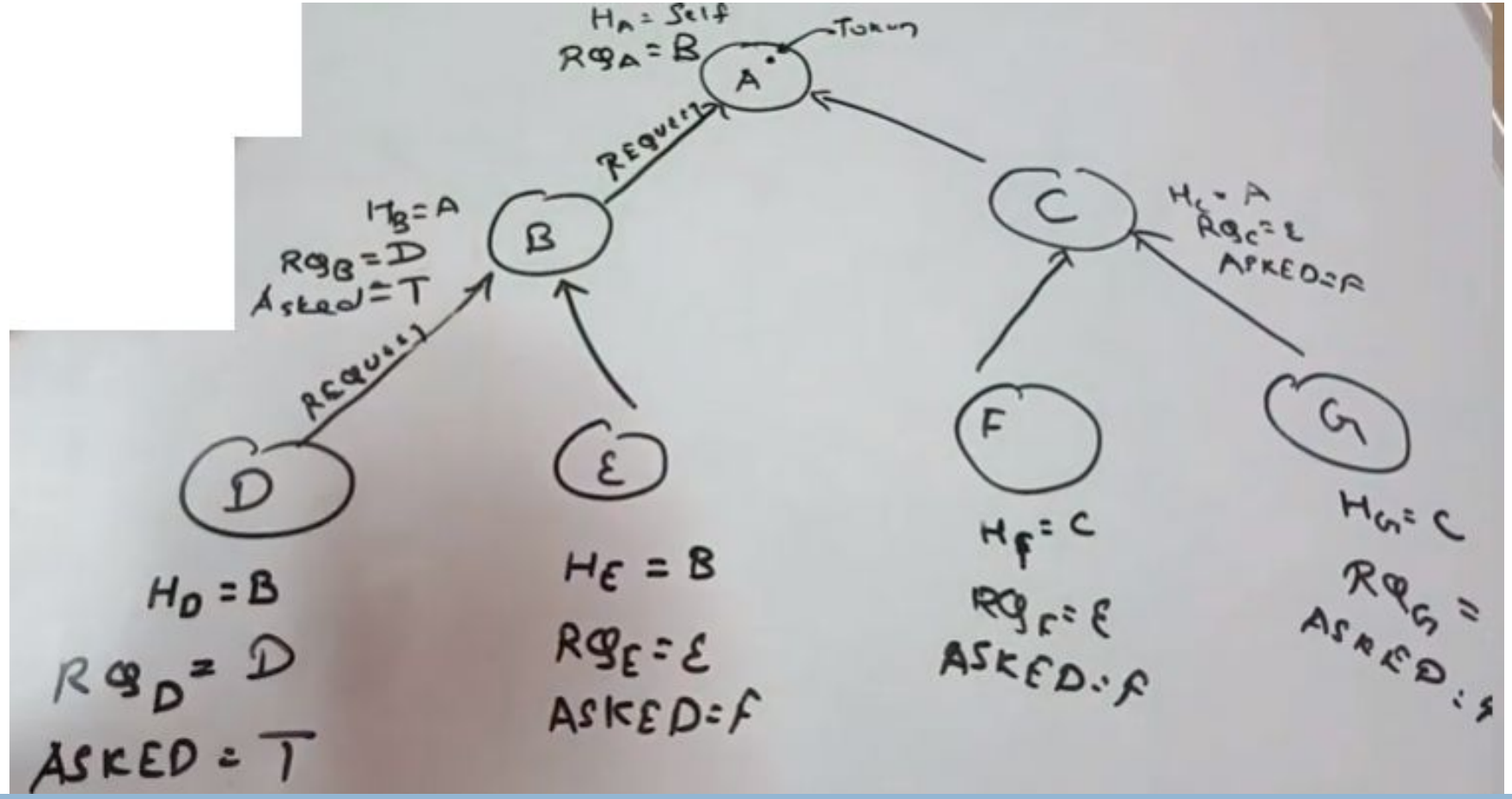


Raymond Tree Based Algorithm

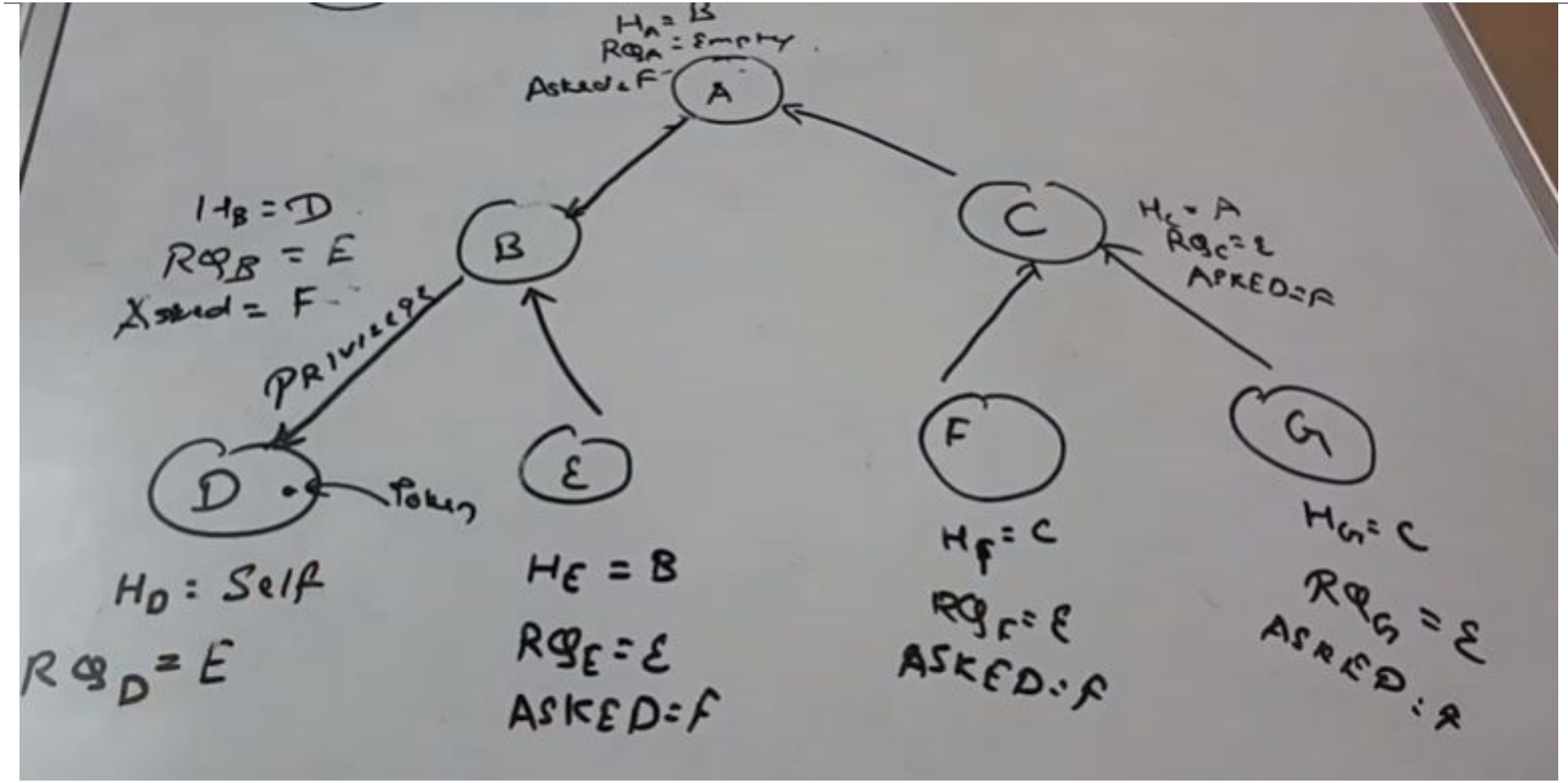


- **P0** receives the REQUEST from P3 through P1
- It surrenders the token and passes it to P1.
- It also changes the direction and makes P1 the root temporarily.
- **P2** removes the first element from the queue to see which node requested the token.
- Token was requested by P3, P2 surrenders the token and passes it to P3.
- **P3** completes the CS, checks the FIFO queue for the request of the token, if not holds it with itself, till it is demanded by any process.

Raymond Tree Based Algorithm



Raymond Tree Based Algorithm



Thank you