Department of Computer Science and Engineering

FACULTY OF ENGINEERING AND TECHNOLOGY
UNIVERSITY OF LUCKNOW
LUCKNOW

CS-501

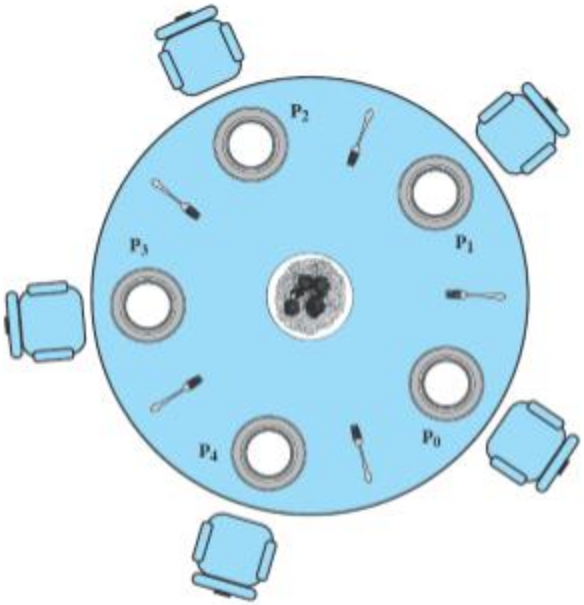
Dr. Zeeshan Ali Siddiqui
Assistant Professor
Deptt. of C.S.E.

CLASSICAL PROBLEMS OF SYNCHRONIZATION

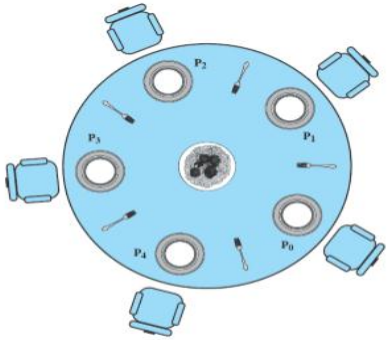
(**The Dining-Philosophers Problem**)

The Dining-Philosophers Problem 1/2

- Consider five philosophers who spend their lives *thinking* and *eating*.
- The *philosophers* share a circular table surrounded by five chairs, each belonging to one philosopher.
- In the center of the table is a bowl of rice, and the table is laid with five single *chopsticks*.



The Dining-Philosophers Problem 2/2



- When a philosopher *thinks*, she does not interact with her colleagues.

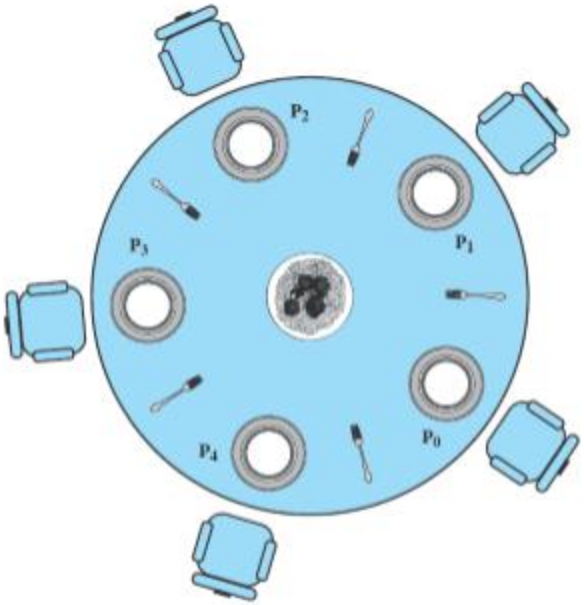
- From time to time, a philosopher gets hungry and tries to pick up the two chopsticks that are *closest* to her (*the chopsticks that are between her and, her left and right neighbors*).

- A philosopher may pick up only one *chopstick* at a time. Obviously, she cannot pick up a chopstick that is already in the hand of a neighbor.

- When a hungry philosopher has both her chopsticks at the same time, she *eats* without releasing the chopsticks. When she is finished eating, she puts down both chopsticks and starts *thinking* again.

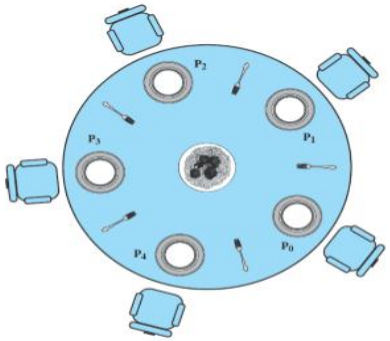
Solution 1/3

- One solution is to represent each chopstick with a *semaphore*.
- A philosopher tries to grab a chopstick by executing a *wait()* operation on that semaphore.
- She releases her chopsticks by executing the *signal()* operation on the appropriate semaphores.



Solution 2/3

- The structure of philosopher i.

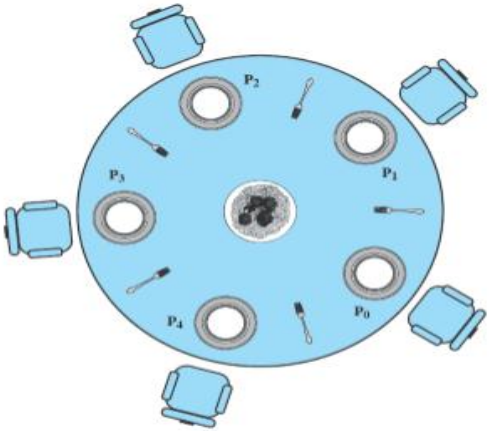


```
semaphore chopstick[5];
do {
    wait(chopstick[i]);
    wait(chopstick[(i+1) % 5]);
        . . .
    /* eat for awhile */
        . . .
    signal(chopstick[i]);
    signal(chopstick[(i+1) % 5]);
        . . .
    /* think for awhile */
        . . .
    } while (true);
```

- Each philosopher picks up first the fork on the *left* and then the fork on the *right*.
- After the philosopher is finished eating, the two forks are *replaced* on the table.

Solution 3/3

- This solution, alas, leads to *deadlock*:
  - If all of the philosophers are hungry at the *same time*, they all sit down, they all pick up the fork on their left, and they all reach out for the other fork, which is not there.
  - In this undignified position, all philosophers *starve*.



Homework

- Possible *remedies* to the deadlock problem.

- **Sleeping Barber Problem**

References

1. Silberschatz, Galvin and Gagne, “Operating Systems Concepts”, Wiley.

2. William Stallings, “Operating Systems: Internals and Design Principles”, 6th Edition, Pearson Education.

3. D M Dhamdhere, “Operating Systems: A Concept based Approach”, 2nd Edition, TMH.

Thank You.