

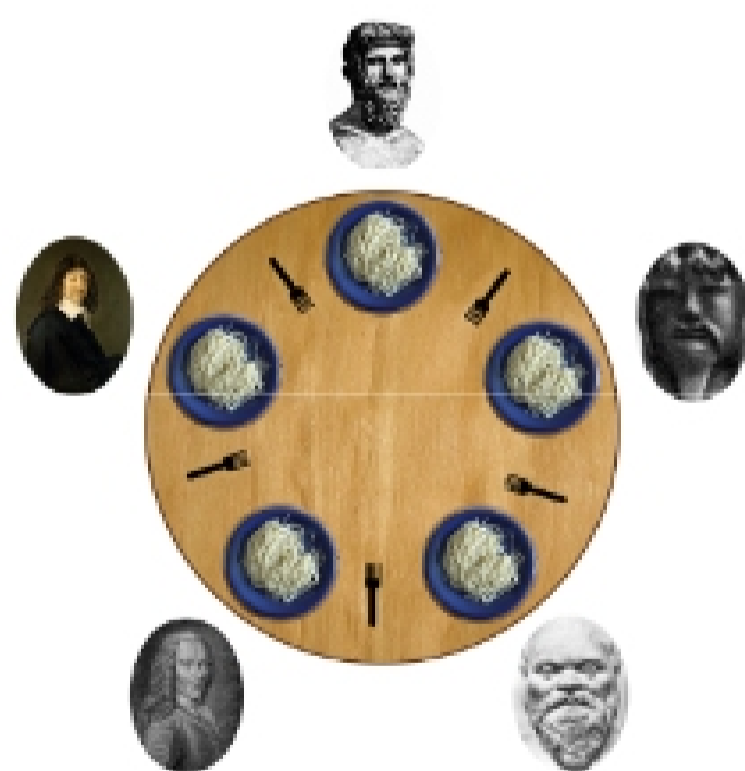
# CMSC 330: Organization of Programming Languages

---

## Threads Classic Concurrency Problems

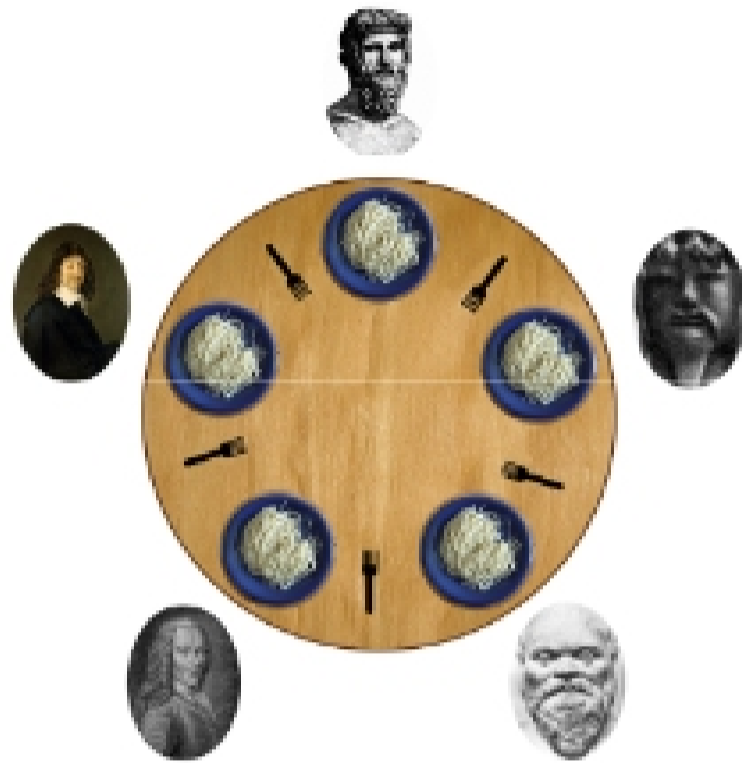
### The Dining Philosophers Problem

---



- Philosophers either eat or think
- They must have two forks to eat
- Can only use forks on either side of their plate
- Avoid deadlock and starvation!

## Bad Dining Philosophers Solution 1

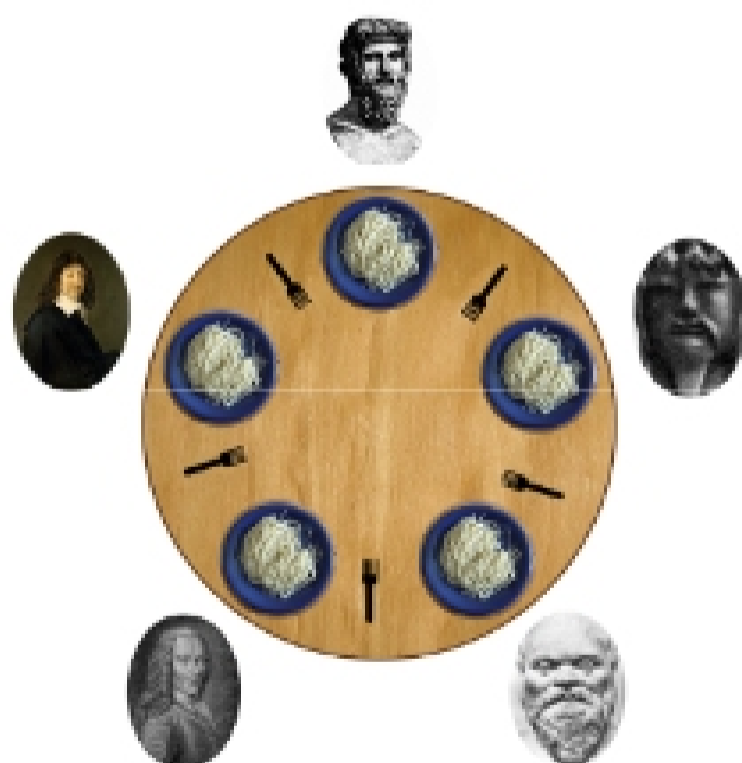


- Philosophers all pick up the left fork first
- **Deadlock!**
  - all are holding the left fork and waiting for the right fork

CMSC 330

3

## Bad Dining Philosophers Solution 2



- Philosophers all pick up the left fork first
- Philosophers put down a fork after waiting for 5 minutes, then wait 5 minutes before picking it up again
- **Starvation!**

CMSC 330

4

## Dining Philosophers Solution

you try!

- Number the philosophers
- Start by giving the fork to the philosopher with lower number. Initially, all forks are dirty.
- When a philosopher wants both forks, he sends a message to his neighbors
- When a philosopher with a fork receives a message if his fork is clean, he keeps it, otherwise he cleans it and gives it up.
- After a philosopher eats, his forks are dirty. If a philosopher had requested his fork, he cleans it and sends it.

CMSC 330

5

## Dining Philosophers Example



Each philosopher begins with the forks shown.

All are dirty.

CMSC 330

6