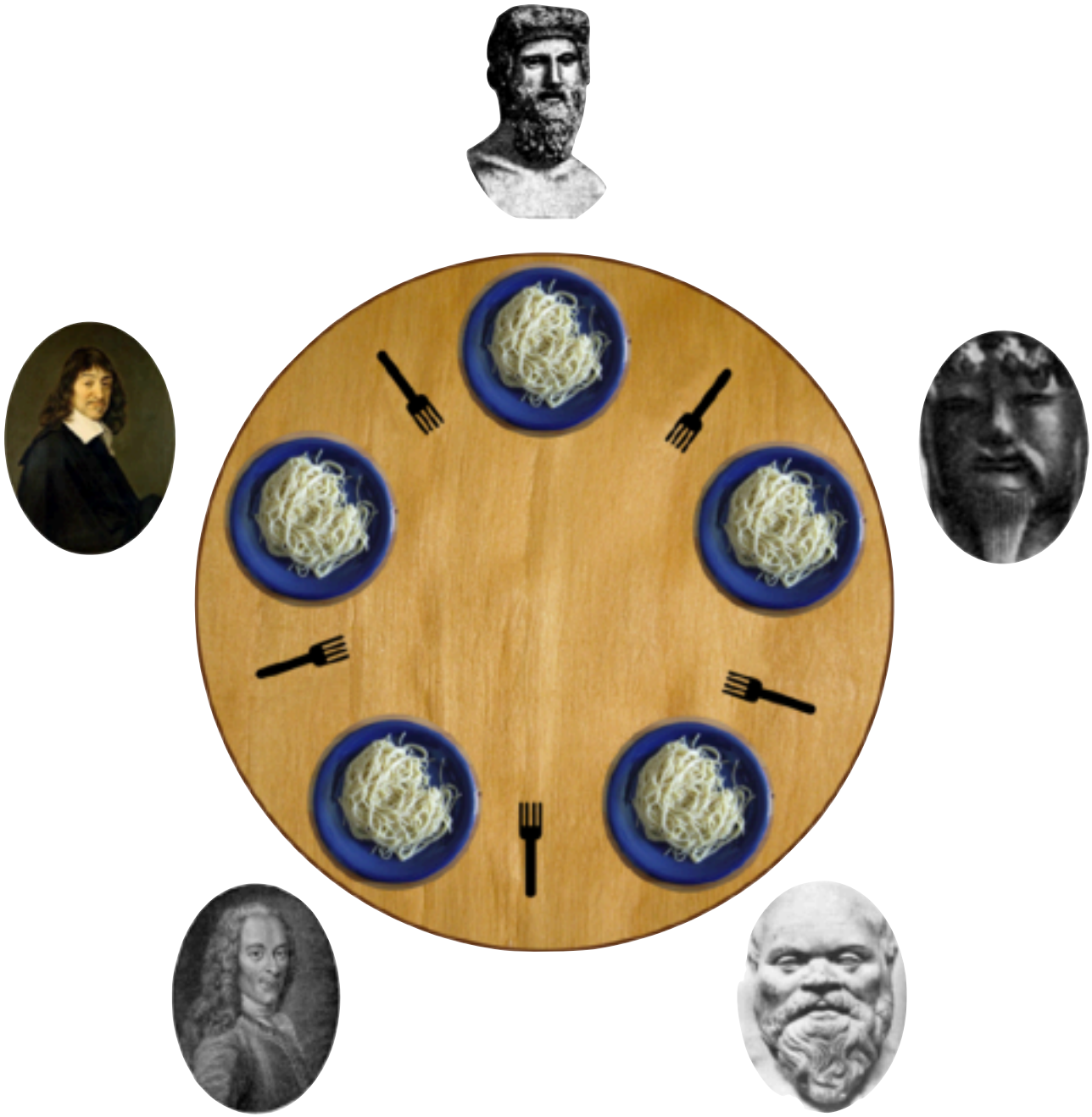
(Bad) Dining Philosophers

image from *wikipedia*

```
void philosopher (i, n) {
  while(1){
    sem_wait(fork[i])
    sem_wait(fork[(i + 1)% n])
    eat & think
    sem_signal(fork[i])
    sem_signal(fork[(i + 1)% n])
  }
}
```

```
for(i=0, i<n, i++){
   sem_init(fork[i], 1)
}
```

- **Deadlock** when each philosopher take the first fork "at the same time"



(Bad) Dining Philosophers

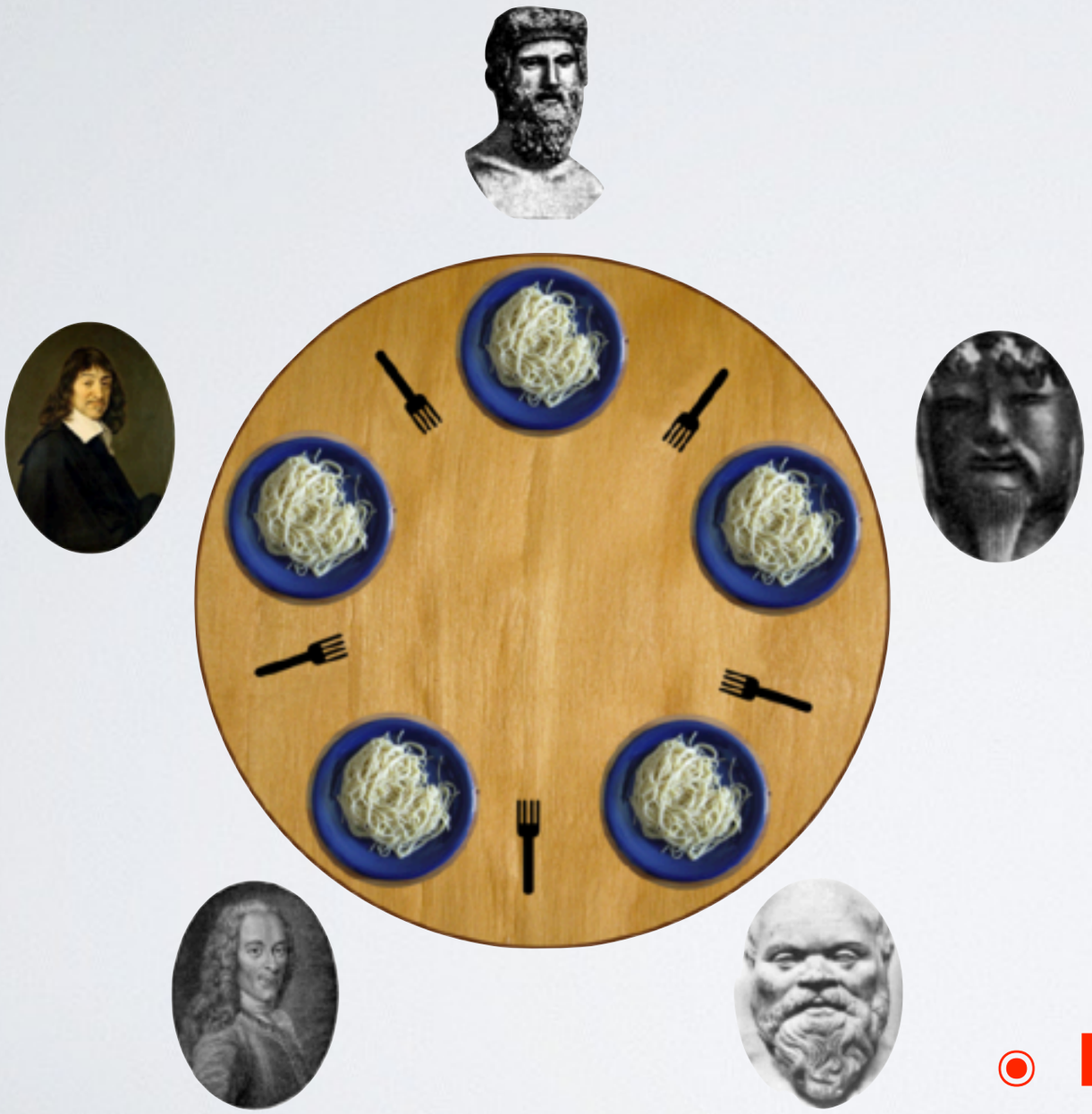


image from *wikipedia*

```
for(i=0, i<n, i++){
   sem_init(fork[i], 1)
}
```

```
void philosopher (i, n) {
  while(1){
     sem_wait(fork[i])
     sem_wait(fork[(i + 1)% n])
     eat & think
     sem_signal(fork[i])
     sem_signal(fork[(i + 1)% n])
   }
}
```

- **Deadlock** when each philosopher take the first fork "at the same time"

(Good) Dining Philosophers

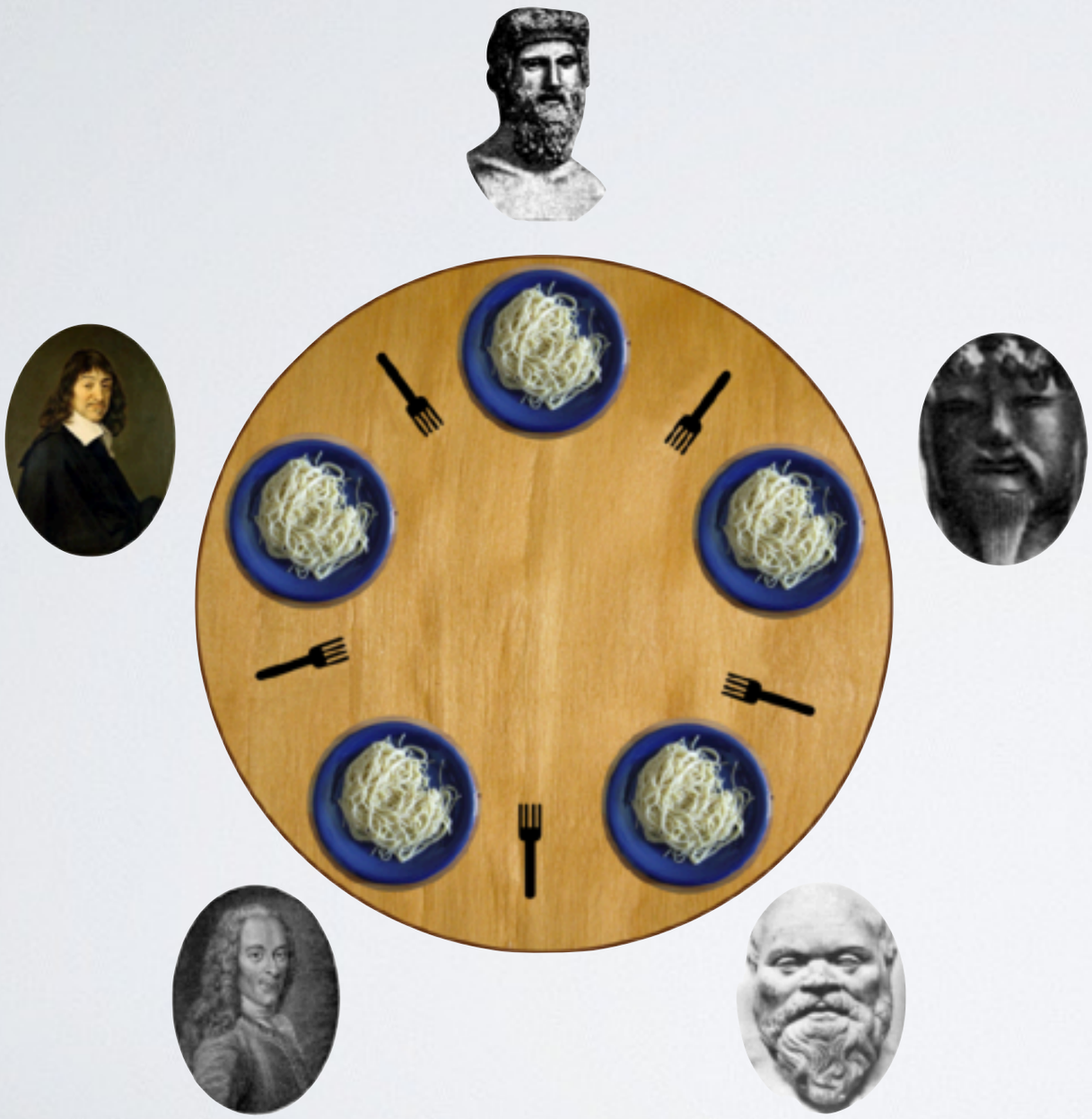


image from *wikipedia*

```
for(i=0, i<n, i++){
   init(fork[i], 1)
}
```

```
void philosopher (i, n) {
  while(1){
    if ((i+1) == n){
      sem_wait(fork[(i + 1)% n])
      sem_wait(fork[i])
    }else{
      sem_wait(fork[i])
      sem_wait(fork[(i + 1)% n])
    }
    eat & think
    sem_signal(fork[i])
    sem_signal(fork[(i + 1)% n])
  }
}
```