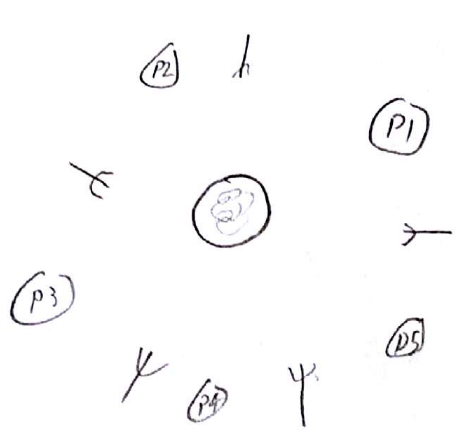
Dining Philosophers



they think()

they eat()

the require 2 forks!

How to do this?

{2 philo can't use same fork at same time (deadlock)}

① Always aquire in same order? Yes for mutex, but can't do this here, for above 5 mutexes (1 per fork), no algorithm to force 5 philosophers to acuire forks in same order

2. pick up fork on left then right. After finished eating put them back down on table!

- all pick up left
- all wait for right? (deadlock)

---

semaphore gets around 'same order' required by mutex, you either derement or increment its counter.

```
semaphore fork[5] = 1
int i
void philosopher (int i) {
    while (true) {

        think()
        wait (fork[i])
        wait (fork[(i+1) mod 5]);
        eat()
        signal (fork[(i+1) mod 5]);
        signal (fork[i]);
    }
}
void main () {
        philosopher (0)
        philosopher (1)
        philosopher (2)
        philosopher (3)
        philosopher (4)
```

Sem

Condvar = CV

mutex = mutex

volatile int = 1

pseudo code

① what happens if

forks = # philosophers? deadlock

forks = # philosophers-1 ? "

forks = # philosophers +1 ? success !!

see

410_Semaphores.Dining_Philosophers

410_Semaphores_Lab