**作业：利用管程解决哲学家进餐问题**

使用的解决方案：规定奇数号哲学家先拿他左边的筷子， 然后再去拿右边的筷子； 而偶数号哲学家则相反。 按此规定， 将是 1 、 2 号哲学家竞争 1 号筷子： 3 、 4 号哲学家竞争 3 号筷子。 即五位哲学家都先竞争奇数号筷子， 获得后， 再去竞争偶数号筷子， 最后总会有一位哲学家 能获得两只筷子而进餐。设置信号量forks[5] = [1, 1, 1, 1, 1]分别表示筷子是否可用

实现伪码：

class Monitor:

Semaphore forks = [1,1,1,1,1] # 初始化, 1筷子表示可用，0表示筷子不可用

def putUpForks(i):

left = (i - 1 + 5) % 5

right = (i + 1) % 5

if i % 2 == 1:

P(forks[left])

V(forks[right])

else:

P(forks[right])

V(forks[left])

def putDownForks(i):

left = (i - 1 + 5) % 5

right = (i + 1) % 5

V(forks[left])

V(forks[right])

def philosopher(i):

while True:

thinking()

hungry()

Monitor.putUpForks(i)

eating()

Monitor.putDownForks(i)

if \_\_name\_\_ == '\_\_main\_\_':

philosopher(0)

philosopher(1)

philosopher(2)

philosopher(3)

philosopher(4)