对N个共享缓冲区读写问题

设empty表空缓冲区的个数，empty=0时N个缓冲区全满生产者不能放入数据 occupied 表示被占用的缓冲区的个数，occupied=0时N个缓冲区全空，消费者不能从中读取数据

empty + occupied = N

初始化 empty = N, occupied = 0, mutex = 0

i = j= 0 //i指向缓冲区写入数据的位置， j指向缓冲区读出数据的位置

def read () :

while(True)

p(occupied)

p(mutex)

item = remove(buffer[j])

j = (j – 1) % N

v(mutex)

v(empty)

print(item)

def write ():

while(True):

item = produce()

p(empty)

p(mutex)

buffer[i] = item

i = (i + 1) % N

v(mutex)

v(occupied)

def main():

coexecute : read(); write();

盘中可放N个水果的吃水果问题PV互斥同步

apple表示放入盘子的苹果的数量，orange 表示放入盘子的橘子的数量, empty 表示盘子空位的数量apple +orange +empty = N

初始化 apple = orange = 0, empty = N, mutex = 0

def father():

while(True):

p(empty)

p(mutex)

fruit = put()

v(mutex)

if fruit == 苹果:

v(apple)

else:

v(orange)

def son():

while(True):

p(orange)

p(mutex)

pick(orange)

v(mutex)

v(empty)

eat()

def daughter():

while(True):

p(apple)

p(mutex)

pick(apple)

v(mutex)

v(empty)

eat()

def main():

coexecute: father(), son(), daughter()