

# OS 第5章作业

徐锦慧\_CS2011\_U202011675

1.

1) FCFS: 143→86→147→91→177→94→150→102→175→130

2) SSTF: 143→147→150→130→102→94→91→86→175→177

3) SCAN: 143→147→150→175→177→130→102→94→91→86

4) C-SCAN: 143→147→150→175→177→86→91→94→102→130

2.

1) 将两个P操作的次序颠倒可能会发生死锁。例如当缓冲区都为满时，生产者仍可顺利执行p(mutex)操作，由于没有空缓冲区而被挂起；但此时缓冲区被生产者占有，消费者无法从缓冲区取出产品，于是产生死锁。

2) 将两个V操作的次序颠倒不会发生死锁。因为V操作只是释放临界资源，不会引起对临界资源的访问。

3.

1) 会产生死锁。当驳船通过A桥，船头到达B桥，驳船请求B桥吊起，而它的尾部占据A桥，若此时AB间弯道已满且有汽车占有B桥，汽车将请求A桥放下，则驳船、汽车均无法前进，产生死锁。

2) 将弯道作为有界缓冲区，只有弯道有空时，车才可以通过B桥。这样当出现（1）中情况时，车将在弯道内等待，驳船可以通过B桥。

```
main{
    int SA=1;    /*A桥的互斥信号量*/
    int SB=1;    /*B桥的互斥信号量*/
    int Sw=n;    /*弯道能容纳的剩余汽车数量*/
    cobegin
        Car1();Car2();...Cari();
        Ship1();Ship2();...Shipi();
    coend
}

Shipi(){
    驳船在河道上行驶；
    P(SA);
    驳船船头行驶至B桥；
    P(SB);
    驳船船尾通过A桥；
    V(SA);
    驳船船尾通过B桥；
    V(SB);
    驳船在河道上行驶；
}
```

```

Cari(){
    汽车在公路上行驶；
    P(SW);
    P(SB);
    汽车通过B桥；
    V(SB);
    汽车通过弯道；
    P(SA);
    汽车上A桥；
    V(SW);
    汽车通过A桥；
    V(SA);
    汽车在公路上行驶；
}

```

4.

1) 系统处于安全状态。安全序列为：P4→P3→P5→P1→P2

2) 不能满足。当P4提出请求后，系统剩余资源为(0,3,2)，P1、P2、P3、P4、P5进程所需的资源分别为(3,4,7)，(1,4,4)，(0,0,6)，(0,2,0)，(5,1,0)，能形成安全队列：P4→P3→P5→P1→P2。但当P1再提出请求时，剩余资源为(0,1,2)，P1、P2、P3、P4、P5进程所需的资源分别为(3,2,7)，(1,4,4)，(0,0,6)，(0,2,0)，(5,1,0)，将产生死锁。

5.

1) 先来先服务调度算法

	进入系统时间	执行时间	开始时间	完成时间	周转时间	带权周转时间
作业1	10.00	2.00	10.00	12.00	2	1
作业2	10.10	1.00	12.00	13.00	2.9	2.9
作业3	10.25	0.25	13.00	13.25	3	12
作业4	9.50	0.20	9.50	9.70	0.2	1

平均周转时间  $t = (2 + 2.9 + 3 + 0.2)/4 = 2.025$

带权平均周转时间  $w = (1 + 2.9 + 12 + 1)/4 = 4.225$

2) 最短作业优先调度算法

	进入系统时间	执行时间	开始时间	完成时间	周转时间	带权周转时间
作业1	10.00	2.00	10	12	2	1
作业2	10.10	1.00	12.25	13.25	3.15	3.15
作业3	10.25	0.25	12	12.25	2	8
作业4	9.50	0.20	9.50	9.70	0.2	1

平均周转时间  $t = (2 + 3.15 + 2 + 0.2)/4 = 1.8375$

带权平均周转时间  $w = (1 + 3.15 + 8 + 1)/4 = 3.2875$

通过计算可以看出，先来先服务算法容易实现，但效率较低，需要更多的周转时间，并且短作业的带权周转时间较高，对短作业不利；最短作业优先调度算法的效率较高，平均周转时间和带权平均周转时间都较低，但是只照顾短作业的利益，对长作业不利。