

一、判断题

1. F 2. F 3. F 4. F 5. T 6. F 7. T 8. T 9. F 10. F

二、填空题

1. PCB 2. 局部性 3. 3 4. 动态内存分配 5. 互斥
6. 同步 7. 15 8. 2 9. 发送(send) 接收(receive) 10. 微内核

三、选择题

1. D 2. C 3. B 4. AB 5. A 6. C 7. B 8. B 9. A 10. C

五、程序题

利用生产消费者模型（方法如下边左图）

```
semaphore empty=10    //空椅子数量
semaphore full=0       //人数
semaphore mutex=1     //互斥信号量，取号机的互斥使用
semaphore service=0   //互斥信号量，取号机的互斥使用

process 客户:
while(1){

    p(empty)           //判断是否有椅子
    p(mutex)           //取号
                        //取号
    v(mutex)
    v(full)             //人数+1
    P(service)         //等待被服务
}

process 职员:
while(1){

    p(full)
    v(empty)
    v(service)         //服务
}
}
```

```
int chairs=10    //空椅子数量
int count=0      //人数
semaphore mutexPeople=1 //互斥信号量，人数互斥
semaphore customer=0  //人数
semaphore mutex=1     //互斥信号量，取号机的互斥使用
semaphore service=0   //服务

process 客户:
while(1){

    p(mutexPeople)
    if (wait<chairs):
        waiting++
        v(mutexPeople)
        p(mutex)
        //取号
        v(mutex)
        v(customer)
        P(service) //等待

    else:
        v(mutexPeople)
        //离开
}

process 职员:
while(1){


    p(customer)
    p(mutexPeople)
    waiting--
    v(mutexPeople)
    V(service) //服务
}
}
```

思考：

如果一个客户到来，发现已经没有座位了，可是他还有急事，那么他可以离开吗？按照上面的解决方案，很明显，他需要等。如果这样要求：当客户到来，发现没有空座位，则立即离开。（方法如上边右图）

六、计算题

1. 答案如下



answer

FCFS 调度算法

P1	P2	P3	P4
0...10...11...15...17			

平均等待时间: $(0 + (10-1) + (11-2) + (15-3)) / 4 = 15/2 = 7.5$

平均周转时间: $(10 + (11-1) + (15-2) + (17-3)) / 4 = 47/4 = 11.75$

非抢占的短作业优先

P1	P2	P4	P3
0...10...11...13...17			

平均等待时间: $(0 + (10-1) + (13-2) + (11-3)) / 4 = 7$

平均周转时间: $(10 + (11-1) + (17-2) + (13-3)) / 4 = 45/4 = 11.25$

抢占的短作业优先

P1	P2	P3	P4	P3	P1
0...1...2...3...5...8...17					

平均等待时间: $(0 + (8-1) + (1-1) + (2-2) + (5-3) + (3-3)) / 4 = 9/4 = 2.25$

平均周转时间: $(17 + (2-1) + (8-2) + (5-3)) / 4 = 13/2 = 6.5$

抢占式优先级

P1	P2	P1	P3	P4
0...1...2...11...15...17				

平均等待时间: $((2-1) + (1-1) + (11-2) + (15-3)) / 4 = 22/4 = 5.5$

平均周转时间: $((11-0) + (2-1) + (15-2) + (17-3)) / 4 = 39/4 = 9.75$

轮转法

P1	P2	P3	P4	P1	P3	P1	P1
0...3...4...7...9...12...13...16...17							

平均等待时间: $((9-3+13-12) + (3-1) + (4-2+12-7) + (7-3)) / 4 = 5$

平均周转时间: $((17-0) + (4-1) + (13-2) + (9-3)) / 4 = 37/4 = 9.25$

2. 答案如下:

(1) 系统中资源总量为某时刻系统中可用资源量与各进程已分配资源量之和, 即

$$(2,1,2) + (1,0,0) + (4,1,1) + (2,1,1) + (0,0,2) = (9,3,6)$$

各进程对资源的需求量为各进程对资源的最大需求量与进程已分配资源量的差, 即

$$\begin{pmatrix} 3 & 2 & 2 \\ 6 & 1 & 3 \\ 3 & 1 & 4 \\ 4 & 2 & 2 \end{pmatrix} - \begin{pmatrix} 1 & 0 & 0 \\ 4 & 1 & 1 \\ 2 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 2 & 2 \\ 2 & 0 & 2 \\ 1 & 0 & 3 \\ 4 & 2 & 0 \end{pmatrix}$$

(2) 略

(3) 如果此时 P2 发出资源请求(1,0,1),按照银行家算法进行检查:

$$\text{Request}(1,0,1) \leq \text{Need}(2,0,2)$$

$$\text{Request}(1,0,1) \leq \text{Available}(2,1,2)$$

分配并修改相应数据结构, 由此形成的进程 P2 请求资源后的资源分配情况如下:

Process	Allocation R1 R2 R3	Need R1 R2 R3	Available R1 R2 R3
P1	1 0 0	2 2 2	1 1 1
P2	5 1 2	1 0 1	
P3	2 1 1	1 0 3	
P4	0 0 2	4 2 0	

再利用安全性检查算法检查系统是否安全, 可得到如下表中所示的安全性检测情况。

Process	Work	Allocation R1 R2 R3	Need R1 R2 R3	Available R1 R2 R3
P2	1 1 1	5 1 2	1 0 1	6 2 3
P3	6 2 3	2 1 1	1 0 3	8 3 4
P4	8 3 4	0 0 2	4 2 0	8 3 6

P1	8 3 6	1 0 0	2 2 2	9 3 6
----	-------	-------	-------	-------

从上表看出，故该状态时安全的，可以立即将进程 P2 所申请的资源分配给它，此时存在一个安全序列{P2, P3, P4, P1}

3. 答案如下，由于老师是老司机了，所以可以不画表格用眼睛对齐，同学们请根据自己的情况谨慎行事。

```
(1) 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
7 7 7 2 2 2 4 4 4 0 0 0 7 7 7
0 0 0 3 3 3 2 2 2 1 1 1 0 0 0
1 1 1 0 0 0 3 3 3 2 2 2 2 2 1
```

按照 FIFO 调度算法将产生 15 次缺页中断，缺页中断率为 15/20=75%

(2)

```
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
7 7 7 2 2 2 2 2 2 7
0 0 0 0 4 0 0 0
1 1 3 3 3 1 1
```

按照 OPT 调度算法将产生 9 次缺页中断，缺页中断率为 9/20=45%

(3)

```
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
7 7 7 2 2 4 4 4 0 1 1 1
0 0 0 0 0 0 3 3 3 0 0
1 1 3 3 2 2 2 2 2 7
```

按照 LRU 调度算法将产生 12 次缺页中断，缺页中断率为 12/20=60%

(4) FIFO 算法会产生 Belady 效应。

4. 由于我很懒，省略了步骤，同学们不要学我，步骤应该写全，这个题目应该画图，否则 GAME OVER。

(1) 首次适应，空闲区域为：20KB (312-332) ,102KB(0-102)

(2) 最佳适应，空闲区域为：110KB (312-422) ,12KB(0-12)

(3) 最差适应，空闲区域为：70KB (312-382) ,52KB(0-52)

5. 32 个页面==》页号有 5 位，

每个页面大小是 2KB==》偏移量有 11 位

内存大小是 16KB，每个帧大小是 2KB ==》共有 8 帧==》帧号有 3 位

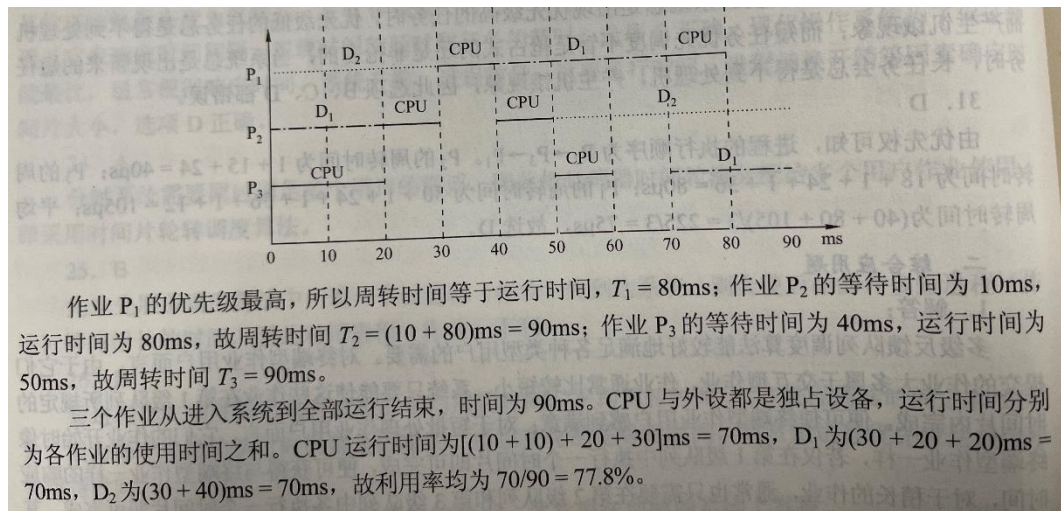
(1) 逻辑地址={页号，偏移量}，所以共 16 位；

物理地址={帧号，偏移量}，所以共 14 位

(2) 物理地址 24C1= 10 0100 1100 0001，红色是偏移量，蓝色是帧号。

所以，帧号是 4，查表得到对应的页号是 2，将页号和偏移量拼接起来，得到逻辑地址=001 0100 1100 0001 = 14C1H （最后一定要写成 16 进制形式）

6. 答案如下



7. (1) $150 \times 2 = 300 \text{ ns}$

(2) $90\% \times (150 + 10) + 10\% \times (150 + 150 + 10) = 175\text{ns}$