

操作系统教程(第三版) 作者: 孙钟秀

部分课后习题答案

第一章 操作系统概论

二、应用题

1、有一台计算机, 具有 1MB 内存, 操作系统占用 200KB, 每个用户占用 200KB。如果用户进程等待 I/O 的时间为 80%, 若增加 1MB 内存, 则 CPU 的利用率提高多少?

解: 每个进程等待的百分比率为 p , 则 n 个进程同时等待的概率为 p^n , 当 n 个进程同时等待 I/O 期间 CPU 是空闲的, 故 CPU 的利用率是 $1-p^n$

除去操作系统占用的内存, 剩余内存能容纳 4 个用户进程, 由于每个用户进程等待 I/O 的时间为 80%, 故 CPU 的利用率为 $1-(80\%)^4=59\%$

若再增加 1M 内存, 内存就能容纳 9 个用户进程了, CPU 的利用率为 $1-(80\%)^9=87\%$ 利用率提高为

$$(87\%)/(59\%)=147\% \\ 147\% - 100\% = 47\%$$

增加 1M 内存 CPU 利用率 47%。

2、设一计算机系统有输入机一台、打印机两台, 现有二道程序同时投入运行, 且程序 A 先开始运行, 程序 B 后运行。

程序 A 的运行轨迹为: 计算 50ms, 打印信息 100ms, 再计算 50ms, 打印信息 100ms, 结束。程序 B 运行的轨迹为: 计算 50ms, 输入数据 80ms, 再计算 100ms, 结束。

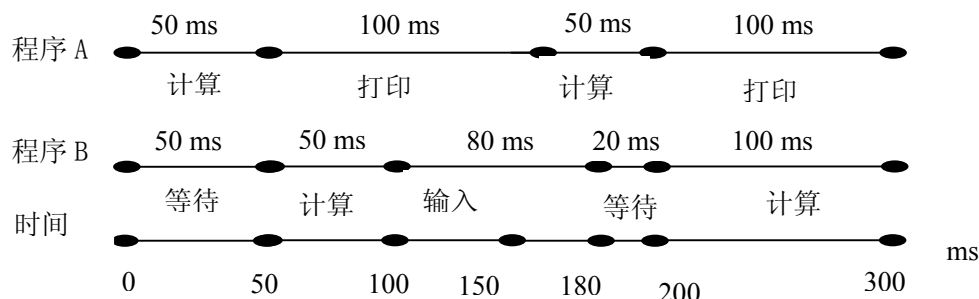
要求:

(1) 用图画出这二道程序并发执行时的工作情况。

(2) 说明在二道程序运行时, CPU 有无空闲等待? 若有, 在哪段时间内等待? 为什么会空闲等待?

(3) 程序 A、B 运行时有无等待现象? 在什么时候会发生等待现象?

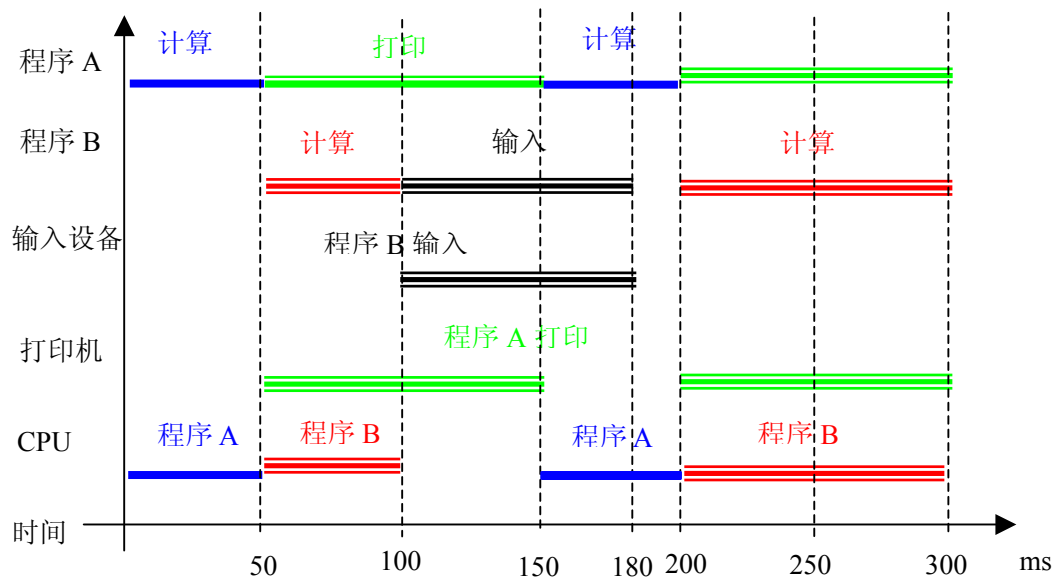
答: (1)工作情况如图。



(2) CPU 有空闲等待, 它发生在 100ms~150ms 时间段内, 此时间段内程序 A 与程序 B 都在进行 I/O 操作。

(3) 程序 A 无等待现象, 程序 B 在 0ms~50ms 时间段与 180ms~200ms 时间段内有等待现象。

工作情况的另一种描述形式如下：



如果将上题的轨迹更改为如下，情况又如何呢？即

一个计算机系统，有一台输入机和一台打印机，现有两道程序投入运行，且程序 A 先开始运行，程序 B 后开始运行。

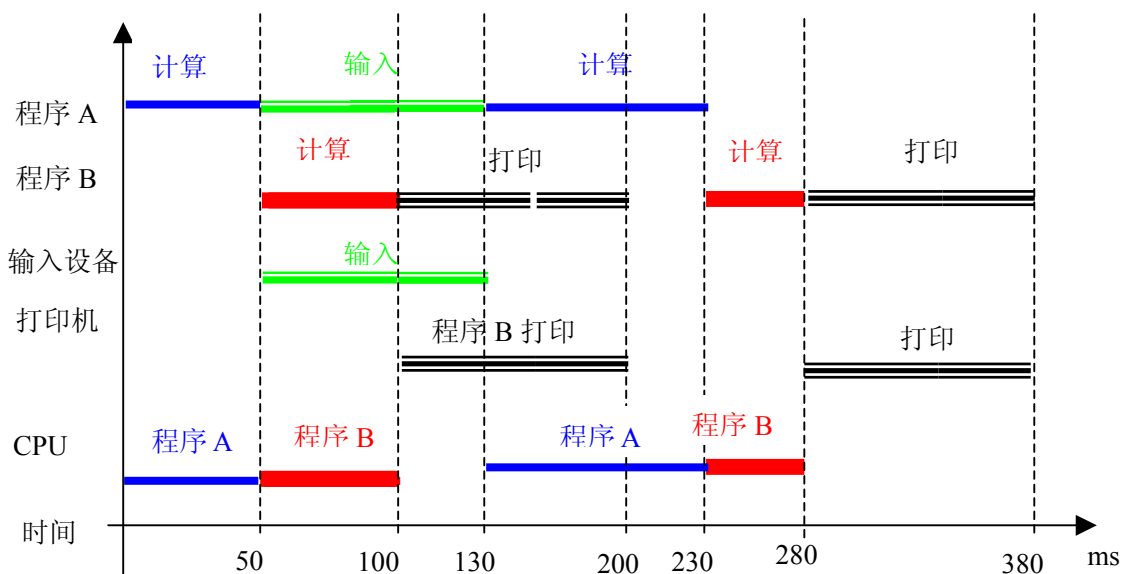
程序 A 的轨迹为：计算 50ms、输入 80ms、再计算 100ms，结束；

程序 B 的运行轨迹为：计算 50ms、打印 100ms、再计算 50ms、打印 100ms，结束。

问题：

- (1) 画出两道程序运行的时间关系图；
- (2) 两道程序运行时，CPU 有无空闲等待？若有，在哪段时间等待？
- (3) 程序 A、B 有无等待 CPU 的情况？若有，在哪段时间等待？

解答：(1) 两道程序运行的时间关系图：

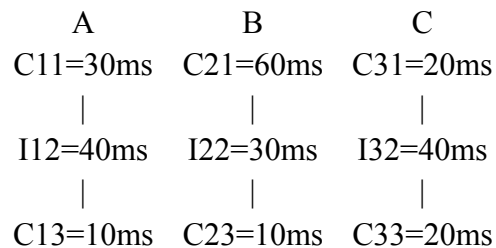


(2) CPU 有空闲等待，它发生在 100ms~130ms 时间段内，此时间段内程序 A 与程序 B

都在进行 I/O 操作。

(3) 程序 A 无等待现象，程序 B 在 0ms~50ms 时间段与 200ms~230ms 时间段内有等待现象。

3、设三道程序，按照 A、B、C 优先次序运行，其内部计算和 I/O 操作时间由图给出。



试画出按多道运行的时间关系图(忽略调度执行时间)。完成三道程序共花多少时间？比单道程序节省了多少时间？若处理器调度程序每次运行程序的转换时间花 1ms，试画出各程序状态转换的时间关系图。

解答：完成三道程序抢占式花费时间是 190 ms，非抢占花费时间是 180 ms，单道花费时间是 260 ms，抢占式比单道节省时间为 70 ms。

单道程序运行时间：260ms

A: $30+40+10=80$ ms

B: $60+30+10=100$ ms

C: $20+40+20=80$ ms

4、在单 CPU 和两台 I/O(I1 和 I2)设备的多道程序设计环境下，同时投入三个作业运行。它们的执行轨迹如下：

Job1: I2(30ms)、CPU(10ms)、I1(30ms)、CPU(10ms)、I2(20ms)

Job2: I1(20ms)、CPU(20ms)、I2(40ms)

Job3: CPU(30ms)、I1(20ms)、CPU(10ms)、I1(10ms)

如果 CPU、I1 和 I2 都能并行工作，优先级从高到低为 Job1、Job2 和 Job3，优先级高的作业可以抢占优先级低的作业的 CPU，但是不抢占 I1 和 I2。试求：

(1)每个作业从投入到完成分别需要多少时间。

(2)从投入到完成 CPU 的利用率。

(3) I/O 设备的利用率。

答：(1)JOB1, JOB2, JOB3 从投入到完成分别所需时间为 110, 90, 110。

(2)每个作业从投入到完成 CPU 的利用率是 72.7%。

(3)I1 的利用率是 72.7%，I2 的利用率是 81.8%。

5、在单 CPU 和两台 I/O(I1 和 I2)设备的多道程序设计环境下，同时投入三个作业运行。它们的执行轨迹如下：

Job1: I2(30ms)、CPU(10ms)、I1(30ms)、CPU(10ms)

Job2: I1(20ms)、CPU(20ms)、I2(40ms)

Job3: CPU(30ms)、I1(20ms)

如果 CPU、I1 和 I2 都能并行工作，优先级从高到低为 Job1、Job2 和 Job3，优先级高的作业可以抢占优先级低的作业的 CPU，但是不抢占 I1 和 I2。试求：

(1)每个作业从投入到完成分别需要多少时间。

(2)从投入到完成 CPU 的利用率。

(3) I/O 设备的利用率。

答：(1)JOB1,JOB2,JOB3 从投入到完成分别所需时间为 80, 90, 90ms。

(2)每个作业从投入到完成 CPU 的利用率是 77.8%。

(3)I1 的利用率是 77.8%，I2 的利用率是 77.8%。

6、若内存中存在 3 道程序 A、B、C，它们按照 A、B、C 的优先次序运行。各程序的计算轨迹为：

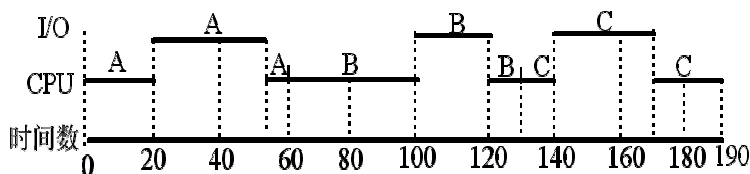
A：计算(20ms)、I/O(30ms)、计算(10ms)

B：计算(40ms)、I/O(20ms)、计算(10ms)

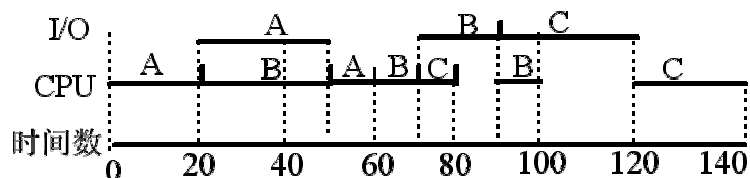
C：计算(10ms)、I/O(30ms)、计算(20ms)

如果三道程序都使用相同的设备进行 I/O(即程序用串行方式使用设备，调度开销忽略不计)。试分别画出单道和多道运行的时间关系图。两种情况下，CPU 的平均利用率各为多少？

答：单道：总运行时间是 190ms，CPU 的利用率是 $110/190=61.3\%$ 。



多道：多道的总运行时间 140ms，CPU 的利用率是 $110/140=78.6\%$ 。



7、若内存中存在 3 道程序 A、B、C，它们按照 A、B、C 的优先次序运行。它们单独运行时的 CPU 和 I/O 占用时间为：

程序 A： 60 20 30 10 40 20 20 (ms)

I/O2 CPU I/O1 CPU I/O1 CPU I/O1

程序 B： 30 40 70 30 30 (ms)

I/O1 CPU I/O2 CPU I/O2

程序 C： 40 60 30 70(ms)

CPU I/O1 CPU I/O2

如果三道程序同时并发执行，调度开销忽略不计，但是优先级高的程序可以中断优先级低的程序，优先级与 I/O 设备无关。试画出多道运行的时间关系图，并问最早与最迟结束的程序是哪个？每道程序执行到结束分别使用了多少时间？计算三个程

序全部运算结束时的 CPU 平均率？

答：最早结束的是 B，最晚的是 C，

A 的运行时间是 250ms，

B 的运行时间是 220ms，

C 的运行时间是 310ms，

CPU 的利用率是 $190/310=61.3\%$ 。

8、若两个程序，A 程序按顺序使用：(CPU)10s，(设备甲)5s，(CPU)10s，(设备乙)10s，(CPU)10s。B 程序按顺序使用：(设备甲)10s，(CPU)10s，(设备乙)5s，(CPU)5s，(设备乙)10s。在顺序环境下先执行 A，在执行 B，求出总的 CPU 利用率为多少？

答：程序 A 的执行了 40 秒，其中 CPU 使用了 25 秒，B 程序执行 40 秒，其中 CPU 使用了 15 秒，而程序共使用了 80 秒，CPU 花 40 秒，CPU 的利用率是

$$40/80=50\%。$$

9、在某计算机系统中，时钟中断处理程序每次执行时间为 2ms(包括进程切换开销)。若中断频率为 60Hz，试问 CPU 用于时钟中断处理的时间比率为多少？

答：因为时钟中断频率是 60HZ，时钟周期是 $1000\text{ms}/60=50/3(\text{ms})$

在每一个时钟周期里，CPU 花 2ms 处理执行任务，所以

CPU 用于时钟中断的时间比例是 $2/(50/3)=6/50=12\%$ 。

第二章 处理机管理

二、应用题

1、下列指令中，哪些只能在核心态运行？

(1)读时钟日期；(2)访管指令；(3)设时钟日期；(4)加载 PSW；(5)置特殊寄存器；(6)改变存储器映象图；(7)启动 I/O 指令。

答：可以在核心态下运行的是：(3)设时钟日期；(4)加载 PSW；(5)置特殊寄存器；(6)改变存储器映象图；(7)启动 I/O 指令。

2、假设有一种低级调度算法是让“最近使用处理器较少的进程”运行，试解释这种算法对“I/O 繁重”型作业有利，但并不是永远不受理“处理器繁重”型作业。

答：因为 I/O 繁忙作业忙于 I/O，所以使用 CPU 较少，按照调度策略算法优先执行。一个进程等待 CPU 时间够长，是最近最少使用 CPU 进程，被优先调度。

3、并发进程之间有什么样的相互制约关系？下列日常生活中的活动属于哪种制约关系？(1)踢足球；(2)吃自助餐；(3)图书馆借书；(4)电视机生产流水线工序

答：并发进程之间的相互制约关系有：互斥和同步。

属于互斥关系的有：(1)踢足球；(3)图书馆借书；

属于同步关系的有：(2)吃自助餐；(4)电视机生产流水线工序

5、若后备作业队列中等待运行的同时有三个作业 J1、J2 和 J3，已知它们各自的运行时间为 a、b、c，且满足 $a < b < c$ 。试证明采用短作业优先算法调度能获得最小平均周转时间。

证：采用短作业优先算法调度时，三个作业的总周转时间为：

$$T1 = a + (a+b) + (a+b+c) = 3a + 2b + c$$

若不按短作业优先算法调度，不失一般性，设调度次序为 J2、J1 和 J3。则三个作业的总周转时间为：

$$T2 = b + (a+b) + (a+b+c) = 3b + 2a + c$$

两个总响应时间之差为： $T2 - T1 = b - a > 0$

可见，采用短作业优先算法调度能获得最小平均周转时间。

7、假定执行表中作业队列，作业号为到达顺序，依次在时刻 0 按次序 1、2、3、4、5 进入单处理器系统。(1) 分别用先来先服务调度算法、时间片轮转调度算法、短作业优先调度算法以及非抢占优先权调度算法算出各作业的执行先后次序(注意优先权高的数值小)；(2) 计算各种情况下作业的平均作业周转时间和平均作业带权周转时间。

作业号	执行时间	优先权
1	10	3
2	1	1
3	2	3
4	1	4
5	5	2

解答：FCFS

作业	执行时间	等待时间	开始时间	完成时间	周转时间	带权周转时间
1	10	0	0	10	10	1
2	1	10	10	11	11	11
3	2	11	11	13	13	6.5
4	1	13	13	14	14	14
5	5	14	14	19	19	3.8
T=13.4 W=7.26						

时间片轮转，时长为 $q=1$

作业	执行时间	提交时间	完成时间	周转时间	带权周转时间
1	10	0	19	19	1.9
2	1	0	2	2	2
3	2	0	7	7	3.5
4	1	0	4	4	4
5	5	0	14	14	2.8
T=9.2 W=2.84					

SJF

次序	执行时间	等待时间	开始时间	完成时间	周转时间	带权周转时间
2	1	0	0	1	1	1
4	1	1	1	2	2	2
3	2	2	2	4	4	2
5	5	4	4	9	9	1.8
1	10	9	9	19	19	1.9
T=7 W=1.74						

非强占优先权调度

次序	执行时间	等待时间	开始时间	完成时间	周转时间	带权周转时间
2	1	0	0	1	1	1
5	5	1	1	6	6	1.2
3	2	6	6	8	8	4
1	10	8	8	18	18	1.8
4	1	18	18	19	19	19
T=10.4 W=5.4						

10、有 5 个批处理作业 A 到 E 均已到达计算中心，其运行时间分别为：2、4、6、8 和 10 分钟；各自的优先级分别被 规定为 1、2、3、4 和 5，这里 5 为最高级。对于 1) 时间片轮转算法、2) 优先数算法、3) 短作业优先算法、4) 先来先服务调度算法(按到达次序 C、D、B、E、A)，在忽略进程切换时间的前提下，计算出平均作业周转时间。(对 1) 每个作业获得相同的 2 分钟的时间片；对 2) 到 4) 采用单道运行，直到结束。)

解答：FCFS

作业	执行时间	等待时间	周转时间	带权周转时间
C	6	0	6	1
D	8	6	14	1.75
B	4	14	18	4.5
E	10	18	28	2.8
A	2	28	30	15
T=19.2 W=5.01				

时间片轮转，时长为 $q=2$

作业	执行时间	等待时间	周转时间	带权周转时间
A	2	0	2	1
B	4	8	12	3
C	6	14	20	3.33
D	8	18	26	3.25
E	10	20	30	3
T=14 W=2				

SJF

次序	执行时间	等待时间	周转时间	带权周转时间
A	2	0	2	1
B	4	2	6	1.5
C	6	6	12	2
D	8	12	20	2.5
E	10	20	30	3
T=14 W=2				

优先权调度

次序	执行时间	等待时间	周转时间	带权周转时间
E	10	0	10	1
D	8	10	18	2.25
C	6	18	24	4
B	4	24	28	7
A	2	28	30	15
T=22 W=5.85				

11、有 5 个批处理作业 A 到 E 均已到达计算中心，其运行时间分别为：10、6、2、4、和 8 分钟；各自的优先级分别被 规定为 3、5、2、1 和 4，这里 5 为最高级。若不考虑系统切换开销，计算出平均作业周转时间。

(1)FCFS(按 A、B、C、D、E)；

(2)优先级调度算法；

(3)时间片轮转算法。

解答: FCFS

作业	执行时间	等待时间	周转时间	带权周转时间
A	10	0	10	1
B	6	10	16	2.66
C	2	16	18	9
D	4	18	22	5.5
E	8	22	30	3.75
T=19.2 W=4.38				

时间片轮转，时长为 $q=2$

作业	执行时间	等待时间	周转时间	带权周转时间
A	10	20	30	3
B	6	16	22	3.66
C	2	4	6	3
D	4	12	16	4
E	8	20	28	3.5
T=20.4 W=3.43				

优先权调度

次序	执行时间	等待时间	周转时间	带权周转时间
B	6	0	6	1
E	8	6	14	1.75
A	10	14	24	2.4
C	2	24	26	13
D	4	26	30	7.5
T=20 W=5.13				

15、单道批处理系统中，下列三个作业采用先来先服务调度算法和最高响应比算法进行调度，哪一种算法性能较好？请完成下表：

作业	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	10:00	2:00				
2	10:10	1:00				
3	10:25	0:25				
平均作业周转时间 $T =$ 平均作业带权周转时间 $W =$						

解答：比较可得 HRRF 优于 FCFS 算法。

FCFS

[illegible]

HRRF

作业	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	10:00	2:00	10:00	12:00	2	120/120
2	10:10	1:00	12:25	13:25	3:15	195/60
3	10:25	0:25	12:00	12:25	2	120/25
T=2.41 W=3.02						

16、若有如表所示四个作业进入系统，分别计算在 FCFS、SJF 和 HRRF 算法下的平均周转时间与带权平均周转时间。(时间以十进制表示)

作业	提交时间	估计运行时间(小时)
1	8.00	2.00
2	8.50	0.50
3	9.00	0.10
4	9.50	0.20

解答：FCFS

作业	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	8.00	2.00	8.00	10.00	2.00	1
2	8.50	0.50	10.00	10.50	2.00	4
3	9.00	0.10	10.50	10.60	1.60	16
4	9.50	0.20	10.60	10.80	1.30	6.5
T=1.725 W=6.875						

SJF

作业	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	8.00	2.00	8.00	10.00	2	1
2	8.50	0.50	10.30	10.80	2.30	4.6
3	9.00	0.10	10.00	10.10	1.10	11
4	9.50	0.20	10.10	10.30	0.80	4
T=1.55 W=5.15						

HRRF

作业	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	8.00	2.00	8.00	10.00	2	1
2	8.50	0.50	10.10	10.60	2.10	4.2
3	9.00	0.10	10.00	10.10	1.10	11
4	9.50	0.20	10.60	10.80	1.30	6.5
T=1.625 W=5.675						

18、若有一个四道作业系统，如果在一段时间内先后有 6 个作业，它们提交和运行时间由下表给出(时间单位为分钟)。系统采用短作业优先(SJF)调度算法，作业被调度进

入系统后中途不会退出，但是作业会被更短的作业抢占。问题：

- (1) 画图给出 6 个作业的执行时间序列；
- (2) 完成各个作业的开始时间、完成时间、周转时间和带权周转时间。
- (3) 计算平均作业周转时间和平均作业带权周转时间。

作业序号	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	8:00	60				
2	8:20	35				
3	8:25	20				
4	8:30	25				
5	8:35	5				
6	8:40	10				

解答： 由下表可知

作业序号	提交时间	运行时间	开始时间	完成时间	周转时间	带权周转时间
1	8:00	60	8:00	10:35	155	$155/60=2.58$
2	8:20	35	8:20	9:55	95	$95/25=3.8$
3	8:25	20	8:25	8:45	20	$20/20=1$
4	8:30	25	9:00	9:25	55	$55/25=2.2$
5	8:35	5	8:45	8:50	15	$15/5=3$
6	8:40	10	8:50	9:00	20	$20/10=2$

1 作业开始于 8:00，结束于 10:35，周转 155 分钟。

2 作业开始于 8:20，结束于 9:55，周转 95 分钟。

3 作业开始于 8:25，结束于 8:45，周转 20 分钟。

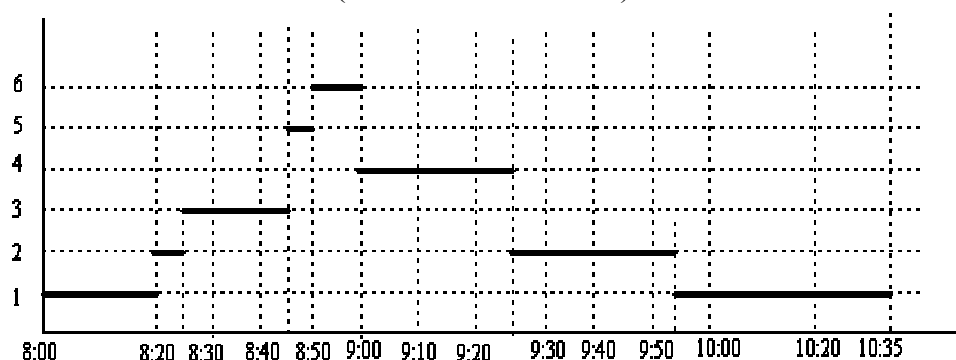
4 作业开始于 9:00，结束于 9:25，周转 55 分钟。

5 作业开始于 8:45，结束于 8:55，周转 15 分钟。

6 作业开始于 8:50，结束于 9:00，周转 20 分钟。

所以平均作业周转时间是 $(155+20+95+55+15+20)/6=60$

平均作业带权周转时间是 $(2.58+3.8+1+2.2+3+2)/6=2.43$



19、有一个具有两道作业的批处理系统，作业调度采用短作业优先的调度算法，进程调度采用以优先数为基础的抢占式调度算法，在下表所示的作业序列，作业优先数即为进程优先数，优先数越小优先级越高。

作业名	提交时间	估计运行时间	优先数
A	10:00	40 分	5
B	10:20	30 分	3
C	10:30	50 分	4
D	10:50	20 分	6

(1) 列出所有作业进入内存时间及结束时间。

(2) 计算平均周转时间。

解答：每个作业运行将经过两个阶段：作业调度(短作业优先 SJF)和进程调度(优先数抢占式)。另外，批处理系统最多容纳 2 道作业，更多的作业将在后备队列等待。

(1) 10:00 作业 A 到达并投入运行；

(2) 10:20 作业 B 到达，且优先数高于作业 A，故作业 B 投入运行，而作业 A 在就绪队列等待；

(3) 10:30 作业 C 到达，但是内存中已有两道作业，故作业 C 进入作业后备队列等待；

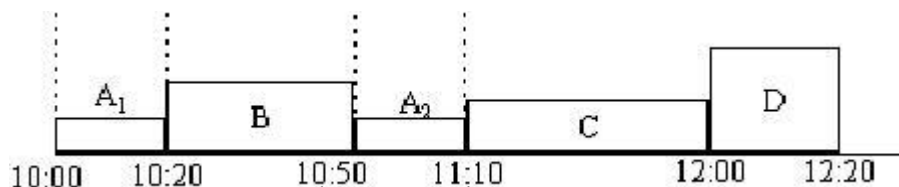
(4) 10:50 作业 B 运行结束，作业 D 到达，按照短作业优先算法，作业 D 被装入内存，进入就绪队列，而由于作业 A 的优先级高于作业 D，作业 A 投入运行；

(5) 11:10 作业 A 运行结束，作业 C 被调入内存，且作业 C 的优先级高于作业 D，作业 C 投入运行；

(6) 12:00 作业 C 运行结束，作业 D 投入运行；

(7) 12:20 作业 D 运行结束；

如图所示



作业 B 运行结束，作业 D 到达，按照短作业优先算法，作业 D 被装入内存，进入就绪队列，而由于作业 A 的优先级高于作业 D，作业 A 投入运行

作业 A 运行结束，作业 C 被调入内存，且作业 C 的优先级高于作业 D，作业 C 投入运行

(2)从图中可以看出

作业名	提交时间	估计运行时间	进入内存时间	运行结束时间
A	10:00	40 分	10:00	11:10
B	10:20	30 分	10:20	10:50
C	10:30	50 分	11:10	12:00
D	10:50	20 分	10:50	12:20

作业 A 的周转时间是 70，作业 B 的周转时间是 30，作业 C 的周转时间是 90，作业 D 的周转时间是 20+70=90，所以平均周转时间是

$$(70+30+90+90)/4=70。$$

24、一个实时系统有 4 个周期性事件，周期分别为 50、100、300 和 250ms，若假设其处理分别需要 35、20、10 和 δ ms，则该系统可调度允许的 δ 最大值为多少 ms。

答：x 应该满足条件 $35/50+20/100+10/300+x/250 \leq 1$, 则 $x \leq 50/3=16.75\text{ms}$,

x 的最大值是 16.75ms.

类比：(1) 一个实时系统有 4 个周期性事件，周期分别为 50、100、250 和 200ms，若假设其处理分别需要 30、20、25 和 δ ms，则该系统可调度允许的 δ 最大值为多少 ms。

答：x 应该满足条件 $30/50+20/100+25/250+x/200 \leq 1$, 则 $x \leq 20\text{ms}$, x 的最大值是 20ms.

(2) 设有周期性实时任务集如下表所示，是否可以调度？

任务	发生周期 T_i	处理时间 C_i
A	30	10
B	40	15
C	50	5

答：由于 $\sum \frac{C_i}{T_i} = \frac{10}{30} + \frac{15}{40} + \frac{5}{50} = \frac{97}{120} = 0.808 < 1$ ，因而可以调度。

第三章 并发进程

二、应用题

4、有一阅览室，共有 100 个座位。读者进入时必须先在一张登记表上登记，该表为每一座位列一表目，包括座号和读者姓名。读者离开时要消掉登记内容。试用 P、V 操作描述读者进程的同步结构。

第一步：确定进程

可以进入阅览室的读者可以有很多，这里设为 n ，即 n 个 Reader（读者）进程

Reader 进程：

登记

进入阅览室

读书

离开阅览室

注销

第二步：确定进程的同步、互斥关系

同步：当教室内有空座位时，读者才可以登记，并进入阅览室

互斥：同时只能有一个读者在入口处进行登记

互斥：同时只能有一个读者在出口处进行注销

第三步：设置信号量

教室内空座位数量， $seat$ ，初值 100

为入口处进行登记设置互斥信号量 Sin ，初值 1，表示当前可用

为出口处进行注销设置互斥信号量 $Sout$ ，初值 1，表示当前可用

第四步：用伪代码描述

begin

$Sin, Sout, seat$: semaphore;

$seat := 100$;

$Sin := 1$;

$Sout := 1$;

cobegin

process Reader- i ($i = 1, 2, \dots, n$);

 begin

$P(seat)$;

$P(Sin)$;

 登记;

$V(Sin)$;

 进入阅览室;

 读书;

 离开阅览室;

$P(Sout)$;

 注销;

$V(Sout)$;

$V(seat)$;

 end

```
coend;  
end;
```

7、设公共汽车上，司机的活动顺序是：启动车辆、正常行车、到站停车；售票员的活动顺序是：关车门、售票、开车门。现假设初始状态为：司机和售票员都已经在车上，汽车处于停止状态，车门处于开的状态。在汽车不断地到站、停车、行驶过程中，请用信号量的 PV 操作实现司机与售票员之间的同步关系。

参考答案：

第一步：确定进程

2 个进程 Driver（司机）、Busman（售票员）

Driver 进程：

启动车辆

正常行车

到站停车

Busman 进程：

关车门

售票

开车门

第二步：确定进程的同步、互斥关系

同步：当售票员将车门关上后，司机才可以启动车辆

同步：当司机到站停车后，售票员打开车门

第三步：设置信号量

车门关上，close，初值 0

到站停车，stop，初值 0

第四步：用伪代码描述

```
begin
```

```
close, stop: semaphore;
```

```
    close := 0;
```

```
    stop := 0;
```

```
cobegin
```

```
Driver ();
```

```
Busman ();
```

```
coend;
```

```
end;
```

```
process Driver ()
```

```
begin
```

```
    L1: P (close);
```

```
    启动车辆;
```

```
        正常行始;
```

```
        到站停车;
```

```
        V (stop);
```

```
        goto L1
```

```
    end;
```

```

process  Busman ( )
begin
    L2: 关车门;
        V (close) ;
        售票;
        P (stop) ;
        开车门;

    goto L2
end;

```

19、设有三个进程 A、B、C，其中 A 与 B 构成一对生产者与消费者（A 为生产者，B 为消费者），共享一个由 n 个缓冲块组成的缓冲池；B 与 C 也构成一对生产者与消费者（此时 B 为生产者，C 为消费者），共享另一个由 m 个缓冲块组成的缓冲池。用 P、V 操作描述它们之间的同步关系。

答：

```

var
    mutexA,emptyA,fullA,mutexC,emptyC,fullC  : semaphore;
    i,j,a,b  : integer;
    bufferA : array 0..n-1 of item
    bufferC : array 0..m-1 of item;

```

procedure produceA: *生产者进程 A*

```

begin
    while ture do
    begin
        Produce next product;
        P(emptyA);
        P(mutexA);
        bufferA(i) :=products;
        i:=(i+1) mod(n) ;
        V(mutexA);
        V(fullA)
    end
end

```

procedure consumer_procedurerB: *消费者和生产者进程 B*

```

begin
    while ture do
    begin
        P(fullA);
        P(mutexA);
        Goods:=buffer(j);
        j:=(j+1)mod(n);

```



```

V(mutexA);
V(emptyA);
Consume goods and Produce next product C;
P(emptyC);
P(mutexC);
BufferC(a):=product C;
a:=(a+1) mod(m);
V(mutexC);
V(fullC)
end
end;

procedure consumerC ; 消费者 C 进程
begin
while ture do
begin
P(fullC);
P(mutexC);
Goods:=bufferC(b);
b:=(b+1) mod(m);
V(mutexC);
V(emptyC);
Consume product;
end
end;
begin
Seminitsal(mutexA.v,1;mutexC.v,1; emptyA.v,n;emptyC.v,m;fullA.V,0;fullC.V,0);
i:=0;j:=0;a:=0;b:=0;
cobegin

produce A
consumer_procedurerB;
consumerC
coend
end.

```

26、设当前的系统状态如下表所示，系统此时 Available=(1,1,2)。

进程	Claim			Allocation		
	R1	R2	R3	R1	R2	R3
P1	3	2	2	1	0	0
P2	6	1	3	5	1	1
P3	3	1	4	2	1	1
P4	4	2	2	0	0	2

问题：(1) 计算各个进程还需要的资源数 Cki-Aki;

- (2) 此时系统是否处于安全状态，为什么？如果安全，则请给出一个安全序列；
- (3) 当 P2 发出资源请求向量 $request_2(1,0,1)$ ，此时系统能否把资源分配给它？为什么？如果能分配给它，则请给出一个安全序列；
- (4) 若在 P2 发出资源请求向量 $request_2(1,0,1)$ 后，若 P1 发出资源请求向量 $request_1(1,0,1)$ ，此时系统能否把资源分配给它？为什么？
- (5) 若在 P1 发出资源请求向量 $request_1(1,0,1)$ 后，若 P3 发出资源请求向量 $request_3(0,0,1)$ ，此时系统能否把资源分配给它？为什么？

解答：(1) 各个进程还需要的资源数 $C_{ki}-A_{ki}$

	C _{ki} -A _{ki}		
进程	R1	R2	R3
P1	2	2	2
P2	1	2	2
P3	1	0	3
P4	4	2	0

- (2) 此时系统是否处于安全状态，安全序列为：P2，P1，P3，P4。
- (3) 当 P2 发出资源请求向量 $request_2(1,0,1)$ ，此时系统可以把资源分配给它，因为系统是安全的，相应的安全序列为：P2，P1，P3，P4。
- (4) 若在 P2 发出资源请求向量 $request_2(1,0,1)$ 后，若 P1 发出资源请求向量 $request_1(1,0,1)$ ，此时系统不能把资源分配给它，因为此时系统资源不足。
- (5) 若在 P1 发出资源请求向量 $request_1(1,0,1)$ 后，若 P3 发出资源请求向量 $request_3(0,0,1)$ ，此时系统不能把资源分配给它，因为如果分配，系统不安全，会导致死锁。

27、设系统中 A、B、C、D 共 4 种资源，在某时刻进程 P0、P1、P2、P3、P4 对资源的占有和需求情况如下表所示。问题：

- (1) 计算各个进程还需要的资源数 $C_{ki}-A_{ki}$ ；
- (2) 此时系统是否处于安全状态，为什么？如果安全，则请给出一个安全序列；
- (3) 若在 P0 发出资源请求向量 $request_0(0,0,1,1)$ 后，此时系统把资源分配给它是否安全？如果安全，则请给出一个安全序列；
- (4) 当 P2 发出资源请求向量 $request_2(1,2,2,2)$ ，此时系统能否把资源分配给它？为什么？
- (5) 若在 P0 发出资源请求向量 $request_0(0,0,1,1)$ 后，P4 若发出资源请求向量 $request_4(0,6,2,2)$ ，此时系统能否把资源分配给它？为什么？

	Allocation				Claim				Available			
进程	A	B	C	D	A	B	C	D	A	B	C	D
P0	0	0	3	2	0	0	4	4	1	6	2	2
P1	1	0	0	0	2	7	5	0				
P2	1	3	5	4	3	6	10	10				
P3	0	3	3	2	0	9	8	4				
P4	0	0	1	4	0	6	6	10				

解答：

- (1) 各个进程还需要的资源数 $C_{ki}-A_{ki}$

	Cki-Aki			
进程	A	B	C	D
P1	0	0	1	2
P2	1	7	5	0
P3	2	3	5	6
P4	0	6	5	2
P5	0	6	5	6

(2) 此时系统是否处于安全状态，安全序列为：P0, P3, P4, P1, P2。

(3) 若在 P0 发出资源请求向量 $request_0(0,0,1,1)$ 后，此时系统把资源分配给它是安全的，安全序列为：P0, P3, P4, P1, P2。

(4) 当 P2 发出资源请求向量 $request_2(1,2,2,2)$ ，此时系统不能把资源分配给它，因为系统将处于不安全状态，会导致死锁。

(5) 若在 P0 发出资源请求向量 $request_0(0,0,1,1)$ 后，P4 若发出资源请求向量 $request_4(0,6,2,2)$ ，此时系统不能把资源分配给它，因为此时系统资源不足。

28、把死锁检测算法用于下面的数据，并请问：

Available = (1, 0, 2, 0)

$$Need = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 \\ 3 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 2 & 1 & 1 & 0 \end{bmatrix} \quad Allocation = \begin{bmatrix} 3 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(1) 此时系统处于安全状态吗？

(2) 若第二个进程提出资源请求 $request_2(0,0,1,0)$ ，系统能分配资源给它吗？

(3) 执行(2)之后，若第五个进程提出资源请求 $request_5(0,0,1,0)$ ，系统能分配资源给它吗？

解：(1)可以找到安全序列，P4,P1,P5,P2,P3。

(2)可以，安全序列是 P1,P4,P5,P2,P3。

(3)不可以，处于不安全状态。

38、桌上有一个空盘子，最多容纳两个水果，每次仅能放入或取出一个水果。爸爸向盘中放苹果，妈妈向盘中放桔子，规定两个儿子专等吃盘中的桔子，两个女儿专等吃盘中的苹果。请用信号量实现爸爸、妈妈、儿子、女儿之间的同步。

第一步：确定进程

4 个进程 Father（爸爸）、Mother（妈妈）、Son（儿子）、Daughter（女儿）

Father 进程：

将苹果放入盘中

Mother 进程：

将桔子放入盘中

Son 进程：

从盘中取出桔子

吃桔子

Daughter 进程:

从盘中取出苹果

吃苹果

第二步: 确定进程的同步、互斥关系

同步: Father 当盘中无水果时, 才可以将苹果放入盘中

同步: Mother 当盘中无水果时, 才可以将桔子放入盘中

同步: Son 当盘中有桔子时, 才可以从盘中取出桔子

同步: Daughter 当盘中有苹果时, 才可以从盘中取出苹果

第三步: 设置信号量

盘中无水果, Sp, 初值 2;

盘中有桔子, So, 初值 0;

盘中有苹果, Sa, 初值 0;

flag0 = flag1 = false;

mutex = 1;

plate ARRAY[0, 1] of (apple, orange)

第四步: 用伪代码描述

begin

 Sp, So, Sa: semaphore;

 Sp := 1;

 So := 0;

 Sa := 0;

cobegin

 Father ();

 Mother ();

 Son ();

 Daughter ();

coend;

end;

process Father ()

 begin

 L1: P (Sp) ;

 P (mutex) ;

 if (flag0 == false) then

 {

 plate[0] = 苹果;

 flag0 = true;

 }

 else

 {

 plate[1] = 苹果;

 flag1 = true;

 }

 v (mutex) ;

```

        V (Sa) ;
        goto L1;
    end;

process  Mother ( )
begin
L2:
    P (Sp) ;
    P (mutex) ;
    if (flag0 == false) then
    {
        plate[0] = 桔子;
        flag0 = true;
    }
    else
    {
        plate[1] = 桔子;
        flag1 = true;
    }
    v (mutex) ;
    V (So) ;
    goto L2;
end;

process  Son ( )
begin
L3: P (So) ;
    P (mutex) ;
    if (flag0 && plate[0] == 桔子) then
    {
        x = plate[0];
        flag0 = false;
    }
    else
    {
        x = plate[1];
        flag1 = false;
    }
    v (mutex) ;
    V (Sp) ;
    吃桔子;
    goto L3;
end;

```

```

process Daughter ( )
begin
L4: P (Sa) ;
    P (mutex) ;
    if (flag0 && plate[0] == 苹果) then
    {
        x = plate[0];
        flag0 = false;
    }
    else
    {
        x = plate[1];
        flag1 = false;
    }
    v (mutex) ;
    V (Sp)
    吃苹果;
    goto L4;
end;

```

桌上有一个空盘子，只允许放一只水果。爸爸或向盘中放苹果，或放桔子，规定儿子只能吃桔子，女儿只能吃苹果。请用信号量实现爸爸、儿子、女儿之间的同步。

解：这是一个典型的同步问题。

首先确定进程个数：三个进程——爸爸、儿子和女儿。

然后确定同步信号量的个数及含义：设置三个同步信号量，Sp 指示盘子是否为空，其初值为“1”表示开始盘子为空；So 指示盘中是否有桔子，其初值为“0”表示开始盘中无桔子；Sa 指示盘中是否有苹果，其初值为“0”表示开始盘中无苹果。

算法描述如下：

```

begin
    Sp,So,Sa: semaphore;
    Sp :=1;
    So :=0;
    Sa :=0;
cobegin
    Father ( );
    Son ( );
    Daughter ( );
coend;
end;

process Father ( )
begin
    L1: P (Sp) ;
        将水果放入盘中;

```

```

        if (放入的是桔子) then
V (So) ;
        else
            V (Sa) ;
            goto L1;
        end;

process Son ( )
begin
    L2: P (So) ;
        从盘中取出桔子;
        V (Sp)
        吃桔子;
        goto L2;
    end;

process Daughter ( )
begin
    L3: P (Sa) ;
        从盘中取出苹果;
        V (Sp)
        吃苹果;
        goto3;
    end;

```

桌上有一个空盘子，只允许放一只水果。爸爸向盘中放苹果，妈妈向盘中放桔子，规定儿子只能吃桔子，女儿只能吃苹果。请用 PV 操作实现他们之间的同步关系：

参考答案：

(1) 桌上一个盘子，只能放一只水果。爸爸放苹果，妈妈放桔子，儿子只吃桔子，女儿只吃苹果。

第一步：确定进程

4 个进程 Father (爸爸)、Mother (妈妈)、Son (儿子)、Daughter (女儿)

Father 进程：

将苹果放入盘中

Mother 进程：

将桔子放入盘中

Son 进程：

从盘中取出桔子

吃桔子

Daughter 进程：

从盘中取出苹果

吃苹果

第二步：确定进程的同步、互斥关系

同步：Father 当盘中无水果时，才可以将苹果放入盘中

同步: Mother 当盘中无水果时, 才可以将桔子放入盘中

同步: Son 当盘中有桔子时, 才可以从盘中取出桔子

同步: Daughter 当盘中有苹果时, 才可以从盘中取出苹果

第三步: 设置信号量

盘中无水果, Sp, 初值 1

盘中有桔子, So, 初值 0

盘中有苹果, Sa, 初值 0

第四步: 用伪代码描述

begin

 Sp, So, Sa: semaphore;

 Sp := 1;

 So := 0;

 Sa := 0;

cobegin

 Father ();

 Mother ();

 Son ();

 Daughter ();

coend;

end;

process Father ()

 begin

 L1: P (Sp) ;

 将苹果放入盘中;

 V (Sa) ;

 goto L1;

 end;

process Mother ()

 begin

 L2: P (Sp) ;

 将桔子放入盘中;

 V (So) ;

 goto L2;

 end;

process Son ()

 begin

 L3: P (So) ;

 从盘中取出桔子;

 V (Sp)

 吃桔子;

 goto L3;


```

        end;

process  Daughter ( )
begin
    L4: P (Sa) ;
        从盘中取出苹果;
    V (Sp)
        吃苹果;
        goto L4;
end;

```

桌上有一个空盘子，只允许放一只水果。爸爸放苹果，妈妈放桔子，儿子吃桔子、苹果。
 请用 PV 操作实现他们之间的同步关系：

第一步：确定进程

3 个进程 Father（爸爸）、Mother（妈妈）、Son（儿子）

Father 进程：

将苹果放入盘中

Mother 进程：

将桔子放入盘中

Son 进程：

从盘中取出水果（桔子或苹果）

吃水果（桔子或苹果）

第二步：确定进程的同步、互斥关系

同步：Father 当盘中无水果时，才可以将苹果放入盘中

同步：Mother 当盘中无水果时，才可以将桔子放入盘中

同步：Son 当盘中有水果（桔子或苹果）时，才可以从盘中取出水果

第三步：设置信号量

盘中无水果，empty，初值 1

盘中有水果（桔子或苹果），full，初值 0

第四步：用伪代码描述

```

begin
    empty, full: semaphore;
    empty:=1;
    full :=0;
cobegin
    Father ( );
    Mother ( );
    Son ( );
coend;
end;

process  Father ( )
begin
    L1: P (empty) ;

```

```

        将苹果放入盘中;
        V (full) ;
        goto L1;
    end;

process  Mother ( )
begin
    L2:  P (empty) ;
        将桔子放入盘中;
        V (full) ;
        goto L2;
    end;

process  Son ( )
begin
    L3: P (full) ;
        从盘中取出水果;
        V (empty) ;
        吃水果;
        goto L3;
    end;

```

40、假设有三个并发进程 P, Q, R, 其中 P 负责从输入设备上读入信息并传送给 Q, Q 将信息加工后传送给 R, R 则负责将信息打印输出。写出下列条件的并发程序:

(1) 进程 P、Q 共享一个缓冲区, 进程 Q、R 共享另一个缓冲区。

(2) 进程 P、Q 共享一个由 m 个缓冲区组成的缓冲池, 进程 Q、R 共享另一个由 n 个缓冲区组成的缓冲池。

参考答案:

(1) 进程 P、Q 共享一个缓冲区, 进程 Q、R 共享另一个缓冲区。

第一步: 确定进程

3 个进程 P、Q、R

P 进程:

从输入设备上读入信息

将信息放入缓冲区 1

Q 进程:

从缓冲区 1 取出信息

将信息放入缓冲区 2 中

R 进程:

从缓冲区 2 取出信息

将信息打印输出

第二步: 确定进程的同步、互斥关系

同步: P 当缓存区 1 无数据时, 才可以向缓冲区 1 写入信息

同步: Q 当缓存区 1 有数据时, 才可以从缓冲区 1 读取信息

同步: Q 当缓存区 2 无数据时, 才可以向缓冲区 2 写入信息

同步：R 当缓存区 2 有数据时，才可以从缓冲区 2 读取信息

第三步：设置信号量

缓存区 1 无数据，empty1，初值 1

缓存区 1 有数据，full1，初值 0

缓存区 2 无数据，empty2，初值 1

缓存区 2 有数据，full2，初值 0

第四步：用伪代码描述

begin

empty1,empty2,full1,full2: semaphore;

empty1 :=1;

 empty2 :=1;

 full1 :=0;

 full2 :=0;

cobegin

P ();

Q ();

R ();

coend;

end;

process P ()

begin

 L1: 从输入设备上读入信息;

 P (empty1) ;

 将信息放入缓冲区 1;

 V (full1) ;

 goto L1

end;

process Q ()

begin

 L2:P (full1) ;

从缓冲区 1 取出信息;

 V (empty1) ;

 P (empty2) ;

将信息放入缓冲区 2;

 V (full2) ;

 goto L2

end;

process R ()

begin

 L3:P (full2) ;

从缓冲区 2 取出信息;

```

        V (empty2) ;
    将信息打印输出 ;
        goto L3 ;
end;

```

(2) 进程 P、Q 共享一个由 m 个缓冲区组成的缓冲池，进程 Q、R 共享另一个由 n 个缓冲区组成的缓冲池。

第一步：确定进程

3 个进程 P、Q、R

P 进程：

从输入设备上读入信息

将信息放入缓冲池 1 中的一个空缓冲区中

Q 进程：

从缓冲池 1 中的一个非空缓冲区中取出信息

将信息放入缓冲池 2 中的一个空缓冲区中

R 进程：

从缓冲池 2 中的一个非空缓冲区中取出信息

将信息打印输出

第二步：确定进程的同步、互斥关系

同步：P 当缓冲池 1 中有空的缓冲区时，才可以向缓冲池 1 写入信息

同步：Q 当缓冲池 1 中有非空的缓冲区时，才可以从缓冲池 1 读取信息

同步：Q 当缓冲池 2 中有空的缓冲区时，才可以向缓冲池 2 写入信息

同步：R 当缓冲池 2 中有非空的缓冲区时，才可以从缓冲池 2 读取信息

第三步：设置信号量

缓冲池 1 中的空缓冲区的数量，empty1，初值 m

缓冲池 1 中的非空缓冲区的数量，full1，初值 0

缓冲池 2 中的空缓冲区的数量，empty2，初值 n

缓冲池 2 中的非空缓冲区的数量，full2，初值 0

第四步：用伪代码描述

```
begin
```

```
empty1,empty2,full1,full2: semaphore;
```

```
empty1 :=m;
```

```
    empty2 :=n;
```

```
    full1 :=0;
```

```
    full2 :=0;
```

```
cobegin
```

```
P ();
```

```
Q ();
```

```
R ();
```

```
coend;
```

```
end;
```

```
process P ()
```

```
begin
```

```

    L1: 从输入设备上读入信息;
        P (empty1) ;
        将信息放入缓冲池 1 中的一个空缓冲区中;
        V (full1) ;
        goto L1
    end;

process  Q ()
begin
    L2:P (full1) ;
    从缓冲池 1 中的一个非空缓冲区中取出信息;
        V (empty1) ;
        P (empty2) ;
    将信息放入缓冲池 2 中的一个空缓冲区中;
        V (full2) ;
        goto L2

    end;

process  R ()
begin
    L3:P (full2) ;
    从缓冲池 2 中的一个非空缓冲区中取出信息;
        V (empty2) ;
    将信息打印输出 ;
        goto L3 ;

    end;

```

某工厂有一个可以存放设备的仓库，总共可以存放 10 台设备。生产的每一台设备都必须入库，销售部门可从仓库提出设备供应客户。设备的入库和出库都必须借助运输工具。现只有一台运输工具，每次只能运输一台设备。请设计一个能协调工作的自动调度管理系统。

参考答案：

第一步：确定进程

可以为入库（Pin）和出库（Pout）各设置一个进程

Pin 进程：

生产了一台设备

使用运输工具入库

Pout 进程：

使用运输工具出库

提出设备供应客户

第二步：确定进程的同步、互斥关系

同步：当仓库中有空余位置存放设备时，设备才可以入库

同步：当仓库中有存放的设备时，设备才可以出库

互斥：运输工具是临界资源，要互斥访问

第三步：设置信号量

仓库中有空余位置数量，empty，初值 10

仓库中有存放的设备数量，full，初值 0

为运输工具设置互斥信号量 S，初值 1，表示当前可用

第四步：用伪代码描述

```
begin
    empty, full, S: semaphore;
    empty := 10;
    full := 0;
    S := 1;
cobegin
    Pin ();
    Pout ();
coend;
end;

process Pin ()
begin
    L1: 生产了一台设备 ;
    P (empty) ;
        P (S);
    使用运输工具入库;
    V (S) ;
        V (full) ;
        goto L1;
    end;

process Pout ()
begin
    L2: P (full) ;
    P (S);
    使用运输工具出库;
    V (S) ;
    V (empty) ;
    提出设备供应客户;
        goto L2;
    end;
```

有四个并发进程：R1，R2，W1 和 W2，它们共享可以存放一个数的缓冲区。进程 R1 每次从磁盘读入一个数存放到缓冲区中，供进程 W1 打印输出；进程 R2 每次从键盘读一个数存放到缓冲区中，供进程 W2 打印输出。当缓冲区满时，不允许再向缓冲区中存放数据；当缓冲区空时，不允许再从缓冲区中取出数据打印输出。试用 PV 操作实现四个进程的协调运行。

参考答案：

第一步：确定进程

4 个进程 R1、R2、W1、W2

R1 进程：

从磁盘上读入一个数

将数存放到缓冲区中

W1 进程：

将 R1 进程放进缓冲区中的数取出

打印输出

R2 进程：

从键盘读入一个数

将数存放到缓冲区中

W2 进程：

将 R2 进程放进缓冲区中的数取出

打印输出

第二步：确定进程的同步、互斥关系

同步：R1 当缓存区无数据时，才可以向缓冲区写入数据

同步：R2 当缓存区无数据时，才可以向缓冲区写入数据

同步：W1 当缓存区中是 R1 写的的数据时，才可以将数据从缓冲区中读出

同步：W2 当缓存区中是 R2 写的的数据时，才可以将数据从缓冲区中读出

第三步：设置信号量

缓存区无数据，empty，初值 1

缓存区中是 R1 写的的数据，full1，初值 0

缓存区中是 R2 写的的数据，full2，初值 0

第四步：用伪代码描述

begin

empty, full1, full2: semaphore;

empty := 1;

full1 := 0;

full2 := 0;

cobegin

R1 ();

R2 ();

W1 ();

W2 ();

coend;

end;

process R1 ()

begin

L1: 从磁盘上读入一个数;

P (empty) ;

将数存放到缓冲区中;

V (full1) ;

goto L1

```

        end;

process  R2 ()
begin
    L2: 从键盘上读入一个数;
        P (empty) ;
        将数存放到缓冲区中;
        V (full2) ;
        goto L2
    end;

process  W1 ()
begin
    L3:P (full1) ;
    将缓冲区中的数取出;
    V (empty) ;
    打印输出;
        goto L3
    end;

process  W2 ()
begin
    L4:P (full2) ;
    将缓冲区中的数取出;
    V (empty) ;
    打印输出;
        goto L4
    end;

```

系统中只有一台打印机，有三个进程在运行中都需要使用打印机进行打印输出，问：这三个进程间有什么样的制约关系？试用信号量的 PV 操作描述这种关系。

参考答案：由于打印机是临界资源，三个进程共享临界资源，是互斥关系。

为临界资源设置互斥信号量 s，初始值为 1：

```

begin
s : semaphore;
s := 1;
cobegin
process  Pi(i=0,1,2)
begin
    其他工作;
    P (s);
    打印;
    V (s);
end

```



```
coend;  
end;
```

哲学家进餐问题：五位哲学家吃面条，只有五根筷子，每个人必须用一双筷子才能吃面条。请用信号量的 PV 操作描述哲学家之间的关系。

请问如下解答是否正确？如果不正确，错在何处？

第一步：确定进程

5 个进程 P_i ($i=0\sim4$)

P_i 进程：

思考

拿起左边筷子

拿起右边筷子

吃面条

放下右边筷子

放下左边筷子

第二步：确定进程的同步、互斥关系

互斥：筷子是互斥资源，每个人都只能使用他左右的两根筷子

第三步：设置信号量

chopstick[5]：表示 5 根筷子，初值 1

第四步：用伪代码描述

```
begin
```

```
chopstick[0~4] : semaphore;
```

```
chopstick[0~4] := 1;
```

```
cobegin
```

```
process  $P_i(i=0,2,\dots,4)$ 
```

```
begin
```

```
    思考;
```

```
    P ( chopstick[i] ) ;
```

```
    P ( chopstick[i+1%5] ) ;
```

```
    吃面条;
```

```
    V ( chopstick[i+1%5] ) ;
```

```
    V ( chopstick[i] ) ;
```

```
end
```

```
coend;
```

```
end;
```

错误原因：假如所有的哲学家都同时拿起左侧筷子，看到右侧筷子不可用，都在等待右侧筷子，无限期地运行，但是都无法取得任何进展，即出现饥饿，所有哲学家都吃不上饭。

解决方案：

至多只允许四个哲学家同时进餐,以保证至少有一个哲学家能够进餐,最终总会释放出他所使用过的两支筷子,从而可使更多的哲学家进餐。

规定奇数号的哲学家先拿起他左边的筷子,然后再去拿他右边的筷子;而偶数号的哲学家则相反.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.即五个哲学家都竞争奇数号筷子,获得后,再去竞争偶数号筷子,最后总会有一个哲学家能获得两支筷子而进餐。而申请不到的哲学家进入阻塞等待队列，则先申请的哲学家会较先可以吃饭，因此不会出现饿死的哲学家。

将拿筷子的操作做成原子操作，即当一个哲学家正在拿筷子的时候，其它的哲学家不能动筷子，当他那好筷子开始吃饭的时候，其它哲学家才可以拿筷子。

第一步：确定进程

5 个进程 P_i ($i=0\sim4$)

P_i 进程：

思考

拿起左边筷子

拿起右边筷子

吃面条

放下右边筷子

放下左边筷子

第二步：确定进程的同步、互斥关系

互斥：筷子是互斥资源，每个人都只能使用他左右的两根筷子

同步：只能有四个人同时吃饭

第三步：设置信号量

chopstick[5]：表示 5 根筷子，初值 1

num：表示允许吃面的人的个数，初值 4

第四步：用伪代码描述

begin

chopstick[0~4] : semaphore;

num : semaphore;

chopstick[0~4] := 1;

num := 4;

cobegin

process P_i ($i=0,2,\dots,4$)

begin

思考;

P (num);

P (chopstick[i]) ;

P (chopstick[i+1%5]) ;

吃面条;

V (chopstick[i+1%5]) ;

```
        V ( chopstick[i] ) ;  
        V (num);  
end  
coend;  
end;
```

第四章 存储器管理

二、应用题

1、在一个请求式分页虚拟存储管理系统中，一个程序运行的页面走向是：

1、2、3、4、2、1、5、6、2、1、2、3、7、6、3、2、1、2、3、6。

分别使用 FIFO、LRU 和 OPT 算法，对分配给程序 3 个页框、4 个页框、5 个页框和 6 个页框的情况下，分别求出缺页中断次数和缺页中断率。

注意：给定的页块初始均为空，因此首次访问一页时就会发生缺页中断。

解：

页框	FIFO	LRU	OPT
3	16	15	11
4	14	10	8
5	12	8	7
6	9	7	7

缺页中断率 = 缺页中断次数 / 20

具体计算如下

3 个页框情况：

(1) FIFO 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	4	4	4	4	6	6	6	6	3	3	3	3	2	2	2	2	6
		2	2	2	2	1	1	1	2	2	2	2	7	7	7	7	1	1	1	1
			3	3	3	3	5	5	5	1	1	1	1	6	6	6	6	6	3	3
是否缺页	是	是	是	是		是	是	是	是	是		是	是	是		是	是		是	是

缺页中断次数为：16

(2) LRU 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	4	4	1	1	1	2	2	2	2	2	6	6	6	1	1	1	6
		2	2	2	2	2	2	6	6	6	6	3	3	3	3	3	3	3	3	3
			3	3	3	3	5	5	5	1	1	1	7	7	7	2	2	2	2	2
是否缺页	是	是	是	是		是	是	是	是	是		是	是	是		是	是			是

缺页中断次数为：15

4 个页框情况：

(1) FIFO 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	1	1	1	5	5	5	5	5	3	3	3	3	3	1	1	1	1
		2	2	2	2	2	2	6	6	6	6	6	7	7	7	7	7	7	3	3
			3	3	3	3	3	3	2	2	2	2	2	2	6	6	6	6	6	6
				4	4	4	4	4	4	1	1	1	1	1	1	2	2	2	2	2
是否缺页	是	是	是	是			是	是	是	是		是	是	是		是	是		是	

缺页中断次数为:14

(2) LRU 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	1	1	1	1	1	1	1	1	1	1	6	6	6	6	6	6	6
		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
			3	3	3	3	5	5	5	5	5	3	3	3	3	3	3	3	3	3
				4	4	4	4	6	6	6	6	6	7	7	7	7	1	1	1	1
是否缺页	是	是	是	是			是	是				是	是	是			是			

缺页中断次数为:10

2、在一个请求式分页虚拟存储管理系统中，一个作业共有 5 页，执行时其访问页面次序为：

(1) 1、4、3、1、2、5、1、4、2、1、4、5；

(2) 3、2、1、4、4、5、5、3、4、3、2、1、5。

若对分配给该作业 3 个页框，分别采用 FIFO、LRU 页面替换算法，求出各自的缺页中断次数和缺页中断率。

解：(1)

算法	缺页中断次数	缺页中断率
FIFO	9	9/12=75%
LRU	8	8/12=67%

(2)

算法	缺页中断次数	缺页中断率
FIFO	9	9/13=69%
LRU	9	8/13=79%

6、一个地址为 32 位的计算机系统使用二级页表，虚地址被分为 9 位顶级页表，11 位二级页表和偏移。试问：页面长度是多少？虚地址空间共有多少个页面？

解：页面大小为 4K，页面数为 2^{20} 。

11、定义段表如下：

段号	段始址	段长
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

给定地址为段号和位数：1) [0, 430]；2) [3, 400]；3) [1, 1]；4) [2, 500]；5) [2, 42]。试求出对应的内存物理地址。

解：对应的物理块号分别为

- (1). $219+430=649$ 。
- (2). $1327+400=1727$ 。
- (3). $2300+1=2301$ 。
- (4). 因为 $500>100$ ，所以地址越界。
- (5). $1952+42=1994$ 。

12、某计算机系统提供 24 位的虚存空间，主存为 2^{18}B ，采用 分页式虚拟存储管理，页面尺寸为 1KB。假定用户程序产生了虚拟地址 11123456(八进制)，而该页面分得块号为 100(八进制)，说明该系统如何产生相应的物理地址及写出物理地址。

解：主存是 2^{18}B ，页为 1K，主存共有 $2^8=256$ 个块，虚拟地址 11123456(八进制)转换成 2 进制为

001 001 001 010 011 100 101 110，

所以后 10 位是偏移位，用分得的块号替换虚拟地址的块号 001 000 000 1 100 101 110，转换成 8 进制就是 201456。

15、在一个分页存储管理系统中，逻辑地址长度为 16 位，页面大小为 4096 字节。现有一逻辑地址为 2F6AH，且第 0、1、2 页依次存在物理块 10、12、14 号中，问相应的物理地址为多少？

解：页大小位 4096 位 4K，共 12 位，故用 $16-12=4$ 位表示页号，将 2F6AH 转换成 2 进制是

0010 1111 0110 1010，页号位 2 对应的内存块号是 14(E)，

所以对应的物理地址是 EF6AH。

17、一台机器有 48 位虚地址和 32 位物理地址，若页长位 8KB，问页表共有多少个页表项？如果设计一个反置页表，则有多少个页表项？

解：页长 8K 位 13 位，所以

页表项位 35 位，即 2^{35} 个页表，反置也表共有 2^{19} 个，19 位。

20、在一个分页虚存系统中，用户编程空间 32 个页，页长为 1KB，主存为 16KB。如果用户程序有 10 个页长，若已知虚页 0、1、2、3，已分到页框 8、7、4、10。试把虚地址 0AC5H 和 1AC5H 转换成对应的物理地址。

解： $1\text{K}=2^{10}\text{B}$ ，所以用 10 位表示偏移位，其余的均为页号，将 0AC5H 转换成二进制 0000 1010 1100 0101，所以该地址对应的虚拟页号是 2，对应的页框号是 4，

用页框替换虚拟页号的 0001 0010 1100 0101，即物理地址是 12C5H。

同理，将 1AC5H 转换成 2 进制是 0001 1010 1100 0101，它的虚拟页号是 6，没有对应的页框号，此时系统就会发生缺页中断，有系统自行分配内存。

24、假设某虚存的用户空间为 1024KB，页面大小为 4KB。内存空间为 512KB。已知用户的虚页 10、11、12、13 分得内存页框号为 62、78、25、36，求出虚地址 0BEBCH(16

进制)的实地址(16 进制)是多少？

解：页面大小为 $4K=2^{12}$ ，所以用 12 位表示偏移位，将 0BEBC 转换成二进制是 0000 1011 1110 1011 1100，它的页号是 11，对应的页框号是 78(4E)，用页框号替换其虚拟页号得到它的实地址 4EEBC。题中告诉内存大小是 512K，说明内存有 128 个页框。

27、考虑下列的段表：

段号	段始址	段长
0	200	500
1	890	30
2	120	100
3	1250	600
4	1800	88

对下面的逻辑地址，求物理地址，如越界请指明。1) <0, 480>; 2) <1, 25>; 3) <1, 14>; 4) <2, 200>; 5) <3, 500>; 6) <4, 100>。试求出对应的内存物理地址。

解：它们的物理地址分别为 680, 915, 904, 越界, 1750, 越界。

- 1) <0, 480> 680;
- 2) <1, 25> 915;
- 3) <1, 14> 904;
- 4) <2, 200> 越界;
- 5) <3, 500> 1750;
- 6) <4, 100> 越界。

28、请页式存储管理系统中，进程访问地址序列为：10, 11, 104, 170, 73, 305, 180, 240, 244, 445, 467, 366。试问(1)如果页面大小为 100，给出页面访问序列。(2) 进程若分得 3 个页框，采用 FIFO 和 LRU 替换算法，求缺页中断率？

解答：(1)页面访问序列为：1, 1, 2, 2, 1, 4, 2, 3, 3, 5, 5, 4。

(2) FIFO 缺页中断为 5 次，缺页中断率为 $5/12=41.6\%$;

	1	1	2	2	1	4	2	3	3	5	5	4
	1	1	1	1	1	1	1	3	3	3	3	3
			2	2	2	2	2	2	2	5	5	5
						4	4	4	4	4	4	4
是否缺页	是		是			是		是		是		

LRU 缺页中断为 6 次，缺页中断率为 $6/12=50\%$;

	1	1	2	2	1	4	2	3	3	5	5	4
	1	1	1	1	1	1	1	3	3	3	3	3
			2	2	2	2	2	2	2	2	2	4
						4	4	4	4	5	5	5
是否缺页	是		是			是		是		是		是

29、假设计算机有 2MB 内存，其中，操作系统占用 512KB，每个用户程序也使用 512KB。如果所有程序都有 70% 的 I/O 等待时间，那么，再增加 1MB 内存，吞吐率增加多少？

解：当计算机有 2M 内存时，最多可容纳 3 个进程，系统的吞吐率为

$$1-(70\%)^3=65.7\%$$

加 1M 内存后，可容纳 5 个进程，系统的吞吐率是

$$1-(70\%)^5=83.2\%$$

系统吞吐量提高了

$$83.2\%/65.7\%-100\%=27\%。$$

35、假设一个任务被划分成 4 个大小相等的段，每段有 8 项的页描述符表，若页面大小为 2KB。试问段页式存储系统中：(a)每段最大尺寸是多少？(b)该任务的逻辑地址空间最大为多少？(c)若该任务访问到逻辑地址空间 5ABCH 中的一个数据，给出逻辑地址的格式。

解答：(a)每段最大尺寸是 16KB；

(b)该任务的逻辑地址空间最大为 64KB；

(c)若该任务访问到逻辑地址空间 5ABCH 中的一个数据，逻辑地址的格式为：
5ABCH，第 1 段 第 1 页 位移由后 11 为给出。

第五章 设备管理

二、应用题

2、现有如下请求队列：8，18，27，129，110，186，78，147，41，10，64，12。试用查找时间最短优先算法计算处理所有请求移动的柱面数。假设磁头当前位置下在磁道100。

解答：处理次序为

100—110—129—147—186—78—64—41—27—18—12—10—8 264

6、有一具有40个磁道的盘面，编号为0~39，当磁头位于第11磁道时，顺序来到如下磁道请求，磁道号是：

1，36，16，34，9，12。

试用：1)先来先服务算法FCFS、2)采用最短寻道时间优先SSTF、3)扫描算法SCAN等三种磁盘驱动调度算法，计算出它们各自要来回穿过多少磁道？

解答：FCFS：11—1—36—16—34—9—12

FCFS穿过的磁道数：111

SSTF：11—12—9—16—1—34—36

SSTF穿过的磁道数：61

SCAN：

朝大方向：11—12—16—34—36—9—1

SCAN朝大方向穿过的磁道数：60

朝小方向：11—9—1—12—16—34—36

SCAN朝小方向穿过的磁道数：45

7、假定某磁盘共有200个柱面，编号为0~199，磁头当前正处在143号柱面，并且刚刚完成了125号柱面的服务请求，如果请求队列的先后访问顺序是：

86，147，91，177，94，150，102，175，130。

试问：为完成上述请求，下列算法存取臂移动的总量是多少？并算出各算法存取臂移动的顺序。

(1) 采用最短寻道时间优先(Shortest Seek Time First Algorithm)

(2) 先来先服务算法(First Come First Server Algorithm)

(3) 电梯调度算法(Elevator Algorithm)

解答：

(1)采用最短寻道时间优先

次序为：143—147—150—130—102—94—91—86—175—177

存取臂移动的总量是162柱面

(2) 先来先服务算法

次序为：143—86—147—91—177—94—150—102—175—130

存取臂移动的总量是565柱面

(3) 电梯调度算法

次序为：143—147—150—175—177—130—102—94—91—86

存取臂移动的总量是 125 柱面

16、假定某磁盘共有 100 个柱面，编号为 0~99，磁头当前正处在 20 号柱面，并且刚刚完成了 15 号柱面的服务请求，如果请求队列的先后访问顺序是:10、22、20、2、40、6、38。若查找移过每个柱面要花 6ms。试问:为完成上述请求，下列算法存取臂移动的总量是多少？并算出各算法分的查找时间。

(1) 采用最短寻道时间优先(Shortest Seek Time First Algorithm)

(2) 先来先服务算法(First Come First Server Algorithm)

(3) 电梯调度算法(Elevator Algorithm)正向柱面大的方向运动

解答：

(1)采用最短寻道时间优先

次序为:20—20—22—10—6—2—38—40

存取臂移动的总量是 60 柱面，时间 360ms

(2) 先来先服务算法

次序为:20—10—22—20—2—40—6—38

存取臂移动的总量是 146 柱面，时间 876ms

(3) 电梯调度算法

次序为:20—20—22—38—40—10—6—2

存取臂移动的总量是 58 柱面，时间 348ms

17、假定在某移动臂磁盘上，刚刚处理了访问 75 号柱面的请求，目前正在 80 号柱面读信息，并且具有下面的请求序列将访问磁盘

请求次序	1	2	3	4	5	6	7	8
欲访问的柱面号	160	40	190	188	90	58	32	102

试用：1) 电梯调度算法、2)采用最短寻道时间优先算法，分别列出实际处理上述请求的次序。

解答：1)电梯调度算法

次序为:80—90—102—160—188—190—50—40—32 268

2)采用最短寻道时间优先

次序为:80—90—102—58—40—32—160—188—190 250

18、计算机系统中，屏幕显示分辨率为 600×480 ，若要存储一屏 256 彩色的图象，需要多少字节的存储空间？

解答：屏幕显示分辨率为 600×480 ，有 $600 \times 480 = 300 \times 2^{10}$ 个像素

对于的 256 彩色图象，一个像素需要 8bits，即 1byte 表示

所以对于一屏 256 彩色的图象则需要的字节数为 300×2^{10} ，即 300KB

第六章 文件管理

二、应用题

5、在 UNIX 中，如果一个盘块的大小为 1KB，每个盘块号为 4 个字节，即每块可放 256 个地址。请转换下列文件的字节偏移量为物理地址：(1)9999；(2)18000；(3)420000。

解答：(1)9999: $L1 = \text{INT}(9999, 1024) = 9$ $B1 = \text{MOD}(9999, 1024) = 783$
(2)18000: $L2 = \text{INT}(18000, 1024) = 17$ $B2 = \text{MOD}(18000, 1024) = 592$
(3)420000: $L3 = \text{INT}(420000, 1024) = 410$ $B3 = \text{MOD}(420000, 1024) = 160$

6、在 UNIX/Linux 系统中，如果当前目录是 /usr/wang，那么，相对路径为 ../ast/xxx 文件的绝对路径名是什么？

解答：相对路径为 ../ast/xxx 文件的绝对路径名是 /usr/ast/xxx

7、一个 UNIX 文件 F 的存取权限为：rwxr-x---，该文件的文件主 uid=12，gid=1，另一个用户的 uid=6，gid=1，是否允许该用户执行文件 F？

解答：可以，因为它们同组。

8、某文件是链接文件，由 5 个逻辑记录组成，每个逻辑记录的大小与磁盘块大小相等，均为 512B，依次存放在 50，121，75，80，63 号盘块上。若要存取文件的第 1569 逻辑地址处的信息，需要访问哪一个磁盘块？

解答：逻辑盘块号 = $1569 \text{ DIV } 512 = 3$

块内地址 = $1569 \text{ MOD } 512 = 33$

逻辑盘块号 3 对应的物理盘块号为 80，所以需要访问 80 这个磁盘块的第 33 字节。

类似的：某文件是链接文件，由 5 个逻辑记录组成，每个逻辑记录的大小与磁盘块大小一致，均为 512B，依次存放在 25，30，32，60，78 号盘块上。若要存取文件的第 1518 逻辑地址处的信息，需要访问哪一个磁盘块？

解答：逻辑盘块号 = $1518 \text{ DIV } 512 = 2$

块内地址 = $1518 \text{ MOD } 512 = 494$

逻辑盘块号 2 对应的物理盘块号为 32，所以需要访问 32 这个磁盘块。

14、有一磁盘组共有 10 个盘面，每个盘面上有 100 个磁道，每个磁道有 16 个扇区。若以扇区为分配单位，现问：若使用位示图管理磁盘空间，则位示图需要占用多少空间？

解答：盘块数 = 磁盘组的总空间大小 / (字节数/盘块)
= (盘面数 × 磁道数/盘面 × 扇区数/磁道 × 字节数/扇区) / (字节数/盘块)
= $10 \times 100 \times 16 \times 512 / 512$
= 16000

所以位示图共占用 16000bit (2000B)。

类似的：有一磁盘组共有 10 个盘面，每个盘面上有 120 个磁道，每个磁道有 20 个扇区。假定分配以 2KB 为单位，若使用位示图管理磁盘空间，则位示图需要占用多少空间？

$$\begin{aligned}\text{解答：盘块数} &= \text{磁盘组的总空间大小} / (\text{字节数/盘块}) \\ &= (\text{盘面数} \times \text{磁道数/盘面} \times \text{扇区数/磁道} \times \text{字节数/扇区}) / (\text{字节数/盘块}) \\ &= 10 \times 120 \times 20 \times 512 / (2 \times 1024) \\ &= 6000\end{aligned}$$

所以位示图共占用 6000bit（750B）。

操作系统课堂习题

1、证明：在非抢占式调度算法中，最短作业优先具有最小平均周转时间。

证：假设某一时刻，有 n 个作业到达，其服务时间分别为

$$t_1 \leq t_2 \leq t_3 \leq \dots \leq t_n,$$

所以响应时间是

$$T(t_1) = t_1,$$

$$T(t_2) = t_1 + t_2,$$

.....

$$T(t_n) = \sum_{i=1}^n t_i$$

所以平均响应时间是

$$T_w = \frac{1}{n} \sum_{i=1}^n (n-i+1)t_i,$$

再假设作业 a 和作业 b 交换执行的顺序，且 $a < b$ ，则新的响应时间如下：

$$T'_{av} = \frac{1}{n} [nt_1 + (n-1)t_2 + \dots + (n-a+1)t_b + \dots + (n-b+1)t_a + \dots t_n]$$

$$T'_{av} - T_w = \frac{1}{n} [(n-a+1)t_b - (n-a+1)t_a + (n-b+1)t_a + (n-b+1)t_a - (n-b+1)t_b] = \frac{1}{n} (t_a - t_b)(b-a)$$

因为

$$a < b,$$

所以

$$t_a < t_b,$$

因而

$$T'_{av} - T_w > 0$$

即按最短作业优先算法调度的平均响应时间最小。

2、一个处理机被所有的就绪队列的进程所复用，进程以无限的速度运行(这是一个理想的时间电轮转调度，它的使用时间片和要求服务的平均时间相比非常小)。

试证明：对于一个无穷的泊松分布输入，一个进程的平均响应时间 t_{Rx} 和服务时间 t_s ，有

$$t_{Rx} = \frac{t_s}{1-p}。$$

证：在理想情况下，按时间电轮转调度算法，一个服务时间为 t_x 的进程在创建后立即被加入就绪队列中，并且系统有 n 个进程，则此进程在系统中时间为 nt_x ，因为该进

[illegible]

HRRN

作业	执行时间	达到时间	开始时间	完成时间	周转时间	带权周转时间
1	3	0	0	3	3	1
2	5	1	3	8	7	1.4
3	2	3	8	10	7	3.5
4	5	9	10	15	6	1.2
5	5	12	15	20	8	1.6
T=6.2 W=1.74						

RR (q=1)

作业	执行时间	达到时间	开始时间	完成时间	周转时间	带权周转时间
1	3	0	0	6	6	2
2	5	1	1	11	10	2
3	2	3	3	8	5	2.5
4	5	9	9	18	9	1.8
5	5	12	12	20	8	1.6
T=7.6 W=1.98						

5、对于给定段表

段	基址	长度
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

下面逻辑地址所对应的物理地址是什么？

- 1) 0, 430 $219+430=649$
- 2) 0, 10 $2300+10=2310$
- 3) 1, 11 $2300+11=2311$
- 4) 2, 500 非法地址访问，越界，由操作系统处理
- 5) 3, 400 $1327+400=1727$
- 6) 4, 112 越界

6、考虑下面页访问串

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6

分别页块是 3 和 4，求下面调度算法各有多少缺页中断？ LRU，FIFO，最佳算法；给定条件是：给定的页框初始为空，因此首次访问就会发生缺页中断。

解： 页块号	LRU	FIFO	最佳算法
3	15	16	11
4	10	14	8

3 个页框情况：

(1) FIFO 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	4	4	4	4	6	6	6	6	3	3	3	3	2	2	2	2	6
		2	2	2	2	1	1	1	2	2	2	2	7	7	7	7	1	1	1	1
			3	3	3	3	5	5	5	1	1	1	1	6	6	6	6	6	3	3
是否缺页	是	是	是	是		是	是	是	是	是		是	是	是		是	是		是	是

缺页中断次数为：16

(2) LRU 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	4	4	1	1	1	2	2	2	2	2	6	6	6	1	1	1	6
		2	2	2	2	2	2	6	6	6	6	3	3	3	3	3	3	3	3	3
			3	3	3	3	5	5	5	1	1	1	7	7	7	2	2	2	2	2
是否缺页	是	是	是	是		是	是	是	是	是		是	是	是		是	是			是

缺页中断次数为：15

4 个页框情况：

(1) FIFO 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	1	1	1	5	5	5	5	5	3	3	3	3	3	1	1	1	1
		2	2	2	2	2	2	6	6	6	6	6	7	7	7	7	7	7	3	3
			3	3	3	3	3	3	2	2	2	2	2	6	6	6	6	6	6	6
				4	4	4	4	4	4	1	1	1	1	1	1	2	2	2	2	2
是否缺页	是	是	是	是			是	是	是	是		是	是	是		是	是		是	

缺页中断次数为：14

(2) LRU 算法

	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
	1	1	1	1	1	1	1	1	1	1	1	1	1	6	6	6	6	6	6	6
		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
			3	3	3	3	5	5	5	5	5	3	3	3	3	3	3	3	3	3
				4	4	4	4	6	6	6	6	6	7	7	7	7	1	1	1	1
是否缺页	是	是	是	是			是	是				是	是	是			是			

缺页中断次数为：10

7、思考：在页式管理中，设贮存大小为 3 页，已知页面走向为 4，3，2，1，4，3，5，4，3，2，1，5。调页时，分别采用先进先出和最佳算法，问缺页率各是多少？

解：FIFO 算法

页面	4	3	2	1	4	3	5	4	3	2	1	5
	4	4	4	1	1	1	5	5	5	5	5	5
		3	3	3	4	4	4	4	4	2	2	2
			2	2	2	3	3	3	3	3	1	1
是否缺页	是	是	是	是	是	是	是			是	是	

缺页中断次数为：9 缺页率=9/12=75%

OPT 算法

页面	4	3	2	1	4	3	5	4	3	2	1	5
	4	4	4	4	4	4	4	4	4	5	5	5
		3	3	3	3	3	3	3	3	3	1	1
			2	1	1	1	5	5	5	2	2	2
是否缺页	是	是	是	是			是			是	是	

缺页中断次数为 7 缺页率=7/12=58.3%

8、思考：考虑页面走向为：4，3，2，1，4，3，5，4，3，2，1，5。当内存块数分别为 3 和 4 时，试问 LRU，FIFO，及 OPT 算法的缺页数各是多少？(假设内存块起始为空，访问时发生缺页中断)。

解：	LRU	FIFO	OPT
3 块	10	9	7
4 块	8	10	6

9、固定分配策略分得 3 个页框,页面走向 2 3 2 1 5 2 4 5 3 2 5 2,分别用 OPT,LRU,FIFO,时钟算法计算它们得缺页中断次数.

解：OPT 算法

页面	2	3	2	1	5	2	4	5	3	2	5	2
	2	2	2	2	2	2	4	4	4	2	2	2
		3	3	3	3	3	3	3	3	3	3	3
				1	5	5	5	5	5	5	5	5
是否缺页	是	是		是	是		是			是		

缺页中断次数为 6

LRU 算法

页面	2	3	2	1	5	2	4	5	3	2	5	2
	2	2	2	2	2	2	2	2	3	3	3	3
		3	3	3	5	5	5	5	5	5	5	5
				1	1	1	4	4	4	2	2	2
是否缺页	是	是		是	是		是		是	是		

缺页中断次数为 7

FIFO 算法

页面	2	3	2	1	5	2	4	5	3	2	5	2
	2	2	2	2	5	5	5	5	3	3	3	3
		3	3	3	3	2	2	2	2	2	5	5
				1	1	1	4	4	4	4	4	2
是否缺页	是	是		是	是	是	是		是		是	是

缺页中断次数为 9

CLOCK 算法

页面	2	3	2	1	5	2	4	5	3	2	5	2
	2	2	2	2	5	5	5	5	3	3	3	3
		3	3	3	3	2	2	2	2	2	2	2
				1	1	1	4	4	4	4	5	5
是否缺页	是	是		是	是	是	是		是		是	

缺页中断次数为 8

10、设某移动磁头磁盘有 256 个柱面，编号为 0~255，磁头当前正处在 95 柱面，对于如下请求访问序列：90, 88, 148, 190, 92, 210, 179, 245, 45, 250, 56, 70, 80 试问：在采用 SSTF(最短寻道时间优先)和 CSCAN(循环扫描，磁头向高柱面号运动)两种不同调度算法下，各算法的平均寻道长度(即平均每次磁头移动距离)为多少？(给出访问顺序，并计算)

SSTF:92, 90, 88, 80, 70, 56, 45, 148, 179, 190, 210, 245, 250

总道数: $3+2+2+8+10+14+11+103+31+11+20+35+5=257$

平均寻道长度=257/13=19.8

CSCAN: 148, 179, 190, 210, 245, 250, 45, 56, 70, 80, 88, 90, 92

总道数：53+31+11+20+35+5+205+11+14+10+8+2+2=407

平均寻道长度=407/13=31.3

11、某文件是链接文件，由 5 个逻辑记录组成，每个逻辑记录的大小与磁盘块大小一致，均为 512B，依次存放在 25，30，32，60，78 号盘块上。若要存取文件的第 1518 逻辑地址处的信息，需要访问哪一个磁盘块？

解答: 逻辑盘块号 = $1518 \text{ MOD } 512 = 2$ 块内地址 = 494

逻辑盘块号 2 对应的物理盘块号为 32，所以需要访问 32 这个磁盘块。

12、根据下表，利用先来先服务算法、短作业优先算法以及最高响应比算法，计算各作业的周转时间和带权周转时间，平均周转时间和平均带权周转时间，比较各算法的性能。

作业名称	提交时间	要求运行时间
1	0	5
2	1	9
3	2	6
4	3	3

解答：先来先服务算法

[illegible]

平均周转时间 $T=14$ 平均带权周转时间 $W=3.3$

短作业优先算法

作业名称	提交时间	要求运行时间	开始运行时刻	完成时刻	T_i	W_i
1	0	5	0	5	5	1
2	1	9	14	23	20	2.2
3	2	6	9	14	12	2
4	3	3	5	8	5	1.6
平均周转时间 $T=10.5$ 平均带权周转时间 $W=1.7$						

最高响应比算法

作业名称	提交时间	要求运行时间	开始运行时刻	完成时刻	T_i	W_i
1	0	5	0	5	5	1
2	1	9	14	23	20	2.2
3	2	6	9	14	12	2
4	3	3	5	8	5	1.6
平均周转时间 $T=10.5$ 平均带权周转时间 $W=1.7$						

比较：先来先服务算法性能最差，算法简单；短作业优先算法性能最好；最高响应比算法不优于短作业优先算法。

13、一分段系统中，某个已装入内存的作业的段表如表所示。

段号	段始址	段长
0	2100	630
1	3500	180
2	200	150
3	1200	300
4	5400	1120

计算下列逻辑地址所对应的物理地址：(0, 311), (1, 120), (2, 230), (4, 800)。

解答：(1)逻辑地址(0, 311)

使用段号 0 查找段表，找到对应的表项后，用段内地址 311 与段长进行比较，没有越界，计算物理地址=段始址+段内地址=2100+311=2411

(2)逻辑地址(1, 120)

使用段号 1 查找段表，找到对应的表项后，用段内地址 120 与段长进行比较，没有越界，计算物理地址=段始址+段内地址=3500+120=3620

(3)逻辑地址(2, 230)

使用段号 2 查找段表，找到对应的表项后，用段内地址 230 与段长进行比较，发现越界，发出越界中断信号，终止程序运行。

(4)逻辑地址(4, 800)

使用段号 4 查找段表，找到对应的表项后，用段内地址 800 与段长进行比较，没有越界，计算物理地址=段始址+段内地址=5400+800=6200

14、设有五个进程，它们到达就绪队列的时刻和运行时间如表所示。若分别采用先来先服务算法和短进程优先算法，试给出各进程的调度顺序以及平均周转时间。

进程	到达时刻	运行时间
P1	10.1	0.3
P2	10.3	0.9
P3	10.4	0.5
P4	10.5	0.1
P5	10.8	0.4

解答：(1)先来先服务(FCFS)

调度顺序	进程	到达时刻	运行时间	开始时间	完成时间	周转时间
1	P1	10.1	0.3	10.1	10.4	0.3
2	P2	10.3	0.9	10.4	11.3	1.0
3	P3	10.4	0.5	11.3	11.8	1.4
4	P4	10.5	0.1	11.8	11.9	1.4
5	P5	10.8	0.4	11.9	12.3	1.5
平均周转时间：T=(0.3 + 1.0 + 1.4 + 1.4 + 1.5)/ 5 = 1.12						

(2) 短进程优先(SPF)

调度顺序	进程	到达时刻	运行时间	开始时间	完成时间	周转时间
1	P1	10.1	0.3	10.1	10.4	0.3
2	P3	10.4	0.5	10.4	10.9	0.5
3	P4	10.5	0.1	10.9	11.0	0.5
4	P5	10.8	0.4	11.0	11.4	0.6
5	P2	10.3	0.9	11.4	12.3	2.0
平均周转时间：T=(0.3 + 0.5 + 0.5 + 0.6 + 2.0)/ 5 = 0.78						

15、设有周期性实时任务集如下表所示，是否可以调度？

任务	发生周期 T_i	处理时间 C_i
A	30	10
B	40	15
C	50	5

答：由于 $\sum \frac{C_i}{T_i} = \frac{10}{30} + \frac{15}{40} + \frac{5}{50} = \frac{97}{120} = 0.808 < 1$ ，因而可以调度

16、在一个请求分页系统中，假设一个作业的页面走向为 4, 3, 2, 1, 4, 3, 5, 4, 3, 2, 1, 5，若分配给该作业的物理块数为 4，假设当前没有任何页面在内存，分别采用 FIFO 和 LRU 页面置换算法，试计算在运行过程中发生的缺页次数和缺页率，并比较所得结果。

解答：(1)采用 FIFO 页面置换算法：

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
缺页	是	是	是	是	否	否	是	是	是	是	是	是
内存块	4	4	4	4	4	4	5	5	5	5	1	1
		3	3	3	3	3	3	4	4	4	4	5
			2	2	2	2	2	2	3	3	3	3
				1	1	1	1	1	1	2	2	2
换页							4	3	2	1	5	4

缺页次数是：10 次，缺页率=缺页次数/访问次数=10/12=83.3%

(2)采用 LRU 页面置换算法：

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
缺页	是	是	是	是	否	否	是	否	否	是	是	是
内存块	4	4	4	4	4	4	4	4	4	4	4	5
		3	3	3	3	3	3	3	3	3	3	3
			2	2	2	2	5	5	5	5	1	1
				1	1	1	1	1	1	2	2	2
换页							2			1	5	4

缺页次数是：8 次，缺页率=缺页次数/访问次数=8/12=66.7%

17、一分页系统中，页面大小为 4KB，某个已装入内存的作业的页表如表所示。请计算下列逻辑地址所对应的物理地址：378，15034，5700，30000。

页号	块号
0	3
1	9
2	10
3	6
4	15

解答：(1)逻辑地址 378：

$$\text{页号} = 378 / 4096 = 0$$

$$\text{页内地址} = 378 \text{MOD} 4096 = 378$$

用页号 0 查找页表，找到对应的块号为 3，则物理地址为：

$$\text{物理地址} = \text{块号} \times \text{页面大小} + \text{页内地址} = 3 \times 4096 + 378 = 12666$$

(2)逻辑地址 15034：

$$\text{页号} = 15034 / 4096 = 3$$

$$\text{页内地址} = 15034 \text{MOD} 4096 = 2746$$

用页号 3 查找页表，找到对应的块号为 6，则物理地址为：

$$\text{物理地址} = \text{块号} \times \text{页面大小} + \text{页内地址} = 6 \times 4096 + 2746 = 27322$$

(3)逻辑地址 5700：

$$\text{页号} = 5700 / 4096 = 1$$

$$\text{页内地址} = 5700 \text{MOD} 4096 = 1604$$

用页号 1 查找页表，找到对应的块号为 9，则物理地址为：

$$\text{物理地址} = \text{块号} \times \text{页面大小} + \text{页内地址} = 9 \times 4096 + 1604 = 38468$$

(4)逻辑地址 30000：

页号 = $30000 / 4096 = 7$ 页内地址 = $30000 \text{MOD} 4096 = 1328$
 用页号 3 查找页表，发现越界，发出越界中断信号，终止程序运行。

18、设有四个进程，它们到达就绪队列的时刻、运行时间及优先级(此处优先级 1 为最低优先级，优先级 4 为最高优先级)如表所示。若分别采用非抢占式优先级调度算法和可抢占式优先级调度算法，试给出各进程的调度顺序以及平均周转时间。

进程	到达时刻	运行时间	优先级
P1	0	8	1
P2	1	3	3
P3	2	7	2
P4	3	12	4

解答：

(1) 非抢占式优先级调度算法

调度顺序	进程	优先级	到达时刻	运行时间	开始时间	完成时间	周转时间
1	P1	1	0	8	0	8	8
2	P4	4	3	12	8	20	17
3	P2	3	1	3	20	23	22
4	P3	2	2	7	23	30	28
平均周转时间： $T = (8 + 17 + 22 + 28) / 4 = 18.75$							

(2) 抢占式优先级调度算法

调度顺序	进程	优先级	到达时刻	剩余运行时间	开始时间	停止时间	共完成时间	状态	周转时间
1	P1	1	0	8	0	1	1	未完成	
2	P2	3	1	3	1	3	2	未完成	
3	P4	4	3	12	3	15	12	完成	12
4	P2	3	1	1	15	16	3	完成	15
5	P3	2	2	7	16	23	7	完成	21
6	P1	1	0	7	23	30	8	完成	30
平均周转时间： $T = (12 + 15 + 21 + 30) / 4 = 19.5$									

19、一个请求分页系统中，内存容量为 1MB，被划分为 256 块，每块为 4KB。有一作业，其页表如表所示。

页号	块号	状态
0	15	1
1	20	1
2	33	1
3	—	0
4	—	0

(1)计算逻辑地址 9016 所对应的物理地址；(2)对逻辑地址 12300，试给出其物理地址的转换过程。

解答：(1)逻辑地址 9016:

$$\text{页号} = 9016 / 4096 = 2$$

$$\text{页内地址} = 9016 \text{MOD} 4096 = 824$$

用页号 2 查找页表，找到对应的块号为 33，则物理地址为：

$$\text{物理地址} = \text{块号} \times \text{页面大小} + \text{页内地址} = 33 \times 4096 + 824 = 135992$$

(2)逻辑地址 12300:

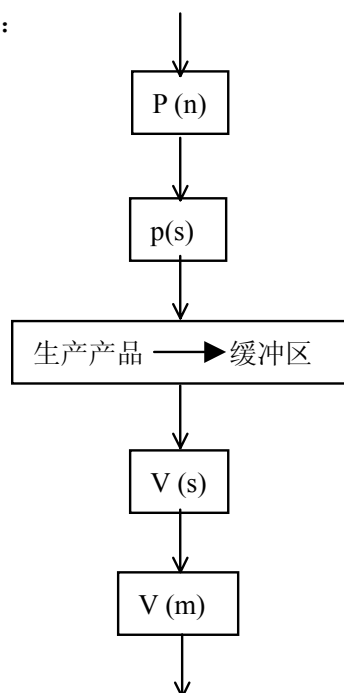
$$\text{页号} = 12300 / 4096 = 3$$

$$\text{页内地址} = 12300 \text{MOD} 4096 = 12$$

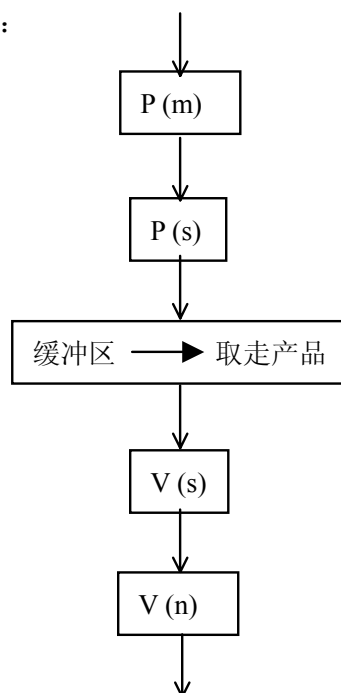
用页号 3 查找页表，发现该页不在内存，发生缺页中断，等把页面调进内存后再重新进行地址转换工作。

20、如图，在生产者——消费者问题中，交换生产者两个 P 操作的次序会有什么结果？交换消费者两个 V 操作的次序呢？说明理由(其中：s 为有界缓冲区的信号量；n 为有界缓冲区中可以存放产品的个数；m 为有界缓冲区中已经存放产品的个数)。
如果在生产者与消费者问题的算法中，若消费速度较生产速度快得多，下图可以简化吗？若可以，如何简化？

生产者：



消费者：

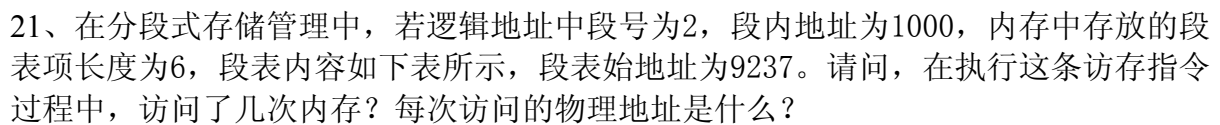


答：交换生产者两 P 操作的次序有可能造成死锁。

例如，当缓冲区中放满产品时，如果此时生产者先做互斥操作，即：P(s), 然后才做同步操作 P(n)，由于此时 $n=-1$ 造成生产者被阻塞。当消费者执行到互斥操作 P(s) 时，由于生产者已执行过 P(s) 并未作释放，所以此时 $s=-1$ ，造成消费者也被阻塞，生产者等消费者取走产品，释放空缓冲区，而消费者则等待生产者释放临界资源缓冲区的使用权，所以两个进程都无法向前推进而造成死锁。

交换消费者两个 V 操作的次序不会发生死锁。因为 V 操作是释放资源，没有被阻

可以简化。简化后的结果为,



答：访问了2次内存。

第二次访问该逻辑地址所对应的具体物理地址： $43556+1000=44556$ 。

作业名称	提交时间	要求运行时间
1	0	8
2	1	4
3	2	9
4	3	5

先来先服务算法

[illegible]

短作业优先算法

作业名称	提交时间	要求运行时间	开始运行时刻	完成时刻	T_i	W_i
1	0	8	0	8	8	1
2	1	4	8	12	11	2.75
3	2	9	17	26	24	2.66
4	3	5	12	17	14	2.8
平均周转时间 $T=14.25$ 平均带权周转时间 $W=2.3$						

最高响应比算法

作业名称	提交时间	要求运行时间	开始运行时刻	完成时刻	T_i	W_i
1	0	8	0	8	8	1
2	1	4	8	12	11	2.75
3	2	9	17	26	24	2.66
4	3	5	12	17	14	2.8
平均周转时间 $T=14.25$ 平均带权周转时间 $W=2.3$						

比较：短作业优先算法性能最好；最高响应比算法不优于短作业优先算法。

23、某一时刻，系统中各进程的资源分配状态如下：

	已分配资源				尚需资源				系统中可用资源			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	0	0	0	2	0	0	1	0	1	5	2	0
P1	1	0	0	0	0	7	5	0				
P2	1	3	5	4	1	0	0	2				
P3	0	6	3	2	0	0	2	0				
P4	0	0	1	4	0	6	4	2				

使用银行家算法回答（给出分析过程）：

（1）该状态下，系统是否安全？

（2）如果进程 P1 请求 (0, 4, 2, 0)，系统能否满足进程的要求？

答：（1）利用安全算法对该时刻资源分配情况进行分析，如下图所示：

	Work	Need	Allocation	Work+Allocation	Finish
P0	1 5 2 0	0 0 1 0	0 0 0 2	1 5 3 2	true
P2	1 5 3 2	1 0 0 2	1 3 5 4	2 8 8 6	true
P3	2 8 8 6	0 0 2 0	0 6 3 2	2 14 11 8	true
P4	2 14 11 8	0 6 4 2	0 0 1 4	2 14 12 12	true
P1	2 14 12 12	0 7 5 0	1 0 0 0	3 14 12 12	true

由以上分析可知，在该时刻存在着一个安全序列 {P0, P2, P3, P4, P1}，故系统是安全的。

或：安全序列 {P0, P3, P2, P4, P1} 等等

（2）如果进程 P1 要求 (0, 4, 2, 0)，

要求 (0, 4, 2, 0) < 尚需 (0, 7, 5, 0)；

要求 (0, 4, 2, 0) < 系统中可用资源 (1, 5, 2, 0)；

系统可为 P1 做试分配，由此形成的资源变化情况如图示：

	已分配资源矩阵	需求资源矩阵	可用资源向量
P1	1 4 2 0	0 3 3 0	1 1 0 0

利用安全算法对该时刻资源分配情况进行分析，如下图所示：

	Work				Need				Allocation				Work+Allocation				Finish
P0	1	1	0	0	0	0	0	0	0	0	1	2	1	1	1	2	true
P2	1	1	1	2	1	0	0	2	1	3	5	4	2	4	6	6	true
P3	2	4	6	6	0	0	2	0	0	6	3	2	2	10	9	8	true
P4	2	10	9	8	0	6	4	2	0	0	1	4	2	10	10	12	true
P1	2	10	10	12	0	3	3	0	1	4	2	0	3	14	12	12	true

由以上分析可知，可找到一个安全序列{P0, P2, P3, P4, P1}，故系统能立即满足进程的要求。

24、已知某程序执行时的页面访问顺序为：0、1、4、2、0、2、6、5、1、2、3。如果该程序有3个物理块可用且使用下列置换算法，求发生页面置换的次数。（1）FIFO算法；（2）LRU算法

答：（1）FIFO 算法

	0	1	4	2	0	2	6	5	1	2	3
0	0	0	0	2	2	2	2	5	5	5	3
	1	1	1	0	0	0	0	1	1	1	1
			4	4	4	4	6	6	6	2	2

页面置换的次数：7 次。

（2）LRU 算法

	0	1	4	2	0	2	6	5	1	2	3
0	0	0	0	2	2	2	2	2	1	5	3
	1	1	1	0	0	0	5	5	1	1	1
			4	4	4	4	6	6	6	2	2

页面置换的次数：7 次。

25、某一时刻，系统中各进程的资源分配状态如下：

	已分配资源矩阵				最多资源矩阵				可用资源向量			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	0	0	1	2	0	0	1	2	1	5	2	0
P1	1	0	0	0	1	7	5	0				
P2	1	3	5	4	2	3	5	6				
P3	0	6	3	2	0	6	5	2				
P4	0	0	1	4	0	6	5	6				

使用银行家算法回答：

（1）该状态下，系统是否安全？

（2）如果进程 P1 请求(0, 4, 2, 0)，系统能否满足进程的要求？

答：（1）利用安全算法对该时刻资源分配情况进行分析，如下图所示：

	Work				Need				Allocation				Work+Allocation				Finish
P0	1	5	2	0	0	0	1	0	0	0	0	2	1	5	3	2	true
P2	1	5	3	2	1	0	0	2	1	3	5	4	2	8	8	6	true
P3	2	8	8	6	0	0	2	0	0	6	3	2	2	14	11	8	true
P4	2	14	11	8	0	6	4	2	0	0	1	4	2	14	12	12	true
P1	2	14	12	12	0	7	5	0	1	0	0	0	3	14	12	12	true

由以上分析可知，在该时刻存在着一个安全序列{P0, P2, P3, P4, P1}，故系统是安全的。

安全的。

或：安全序列 {P0, P3, P2, P4, P1} 等等

(2) 如果进程 P1 要求 (0, 4, 2, 0),

要求 (0, 4, 2, 0) < 尚需 (0, 7, 5, 0);

要求 (0, 4, 2, 0) < 系统中可用资源 (1, 5, 2, 0);

系统可为 P1 做试分配, 由此形成的资源变化情况如图示:

已分配资源矩阵 需求资源矩阵 可用资源向量

P1	1	4	2	0	0	3	3	0	1	1	0	0
----	---	---	---	---	---	---	---	---	---	---	---	---

利用安全算法对该时刻资源分配情况进行分析, 如下图所示:

	Work				Need				Allocation				Work+Allocation				Finish
P0	1	1	0	0	0	0	0	0	0	0	1	2	1	1	1	2	true
P2	1	1	1	2	1	0	0	2	1	3	5	4	2	4	6	6	true
P3	2	4	6	6	0	0	2	0	0	6	3	2	2	10	9	8	true
P4	2	10	9	8	0	6	4	2	0	0	1	4	2	10	10	12	true
P1	2	10	10	12	0	3	3	0	1	4	2	0	3	14	12	12	true

由以上分析可知, 可找到一个安全序列 {P0, P2, P3, P4, P1}, 故系统能立即满足进程的要求。

26、设某移动磁头磁盘有 200 个柱面, 编号为 0~199, 磁头当前正处在 144 柱面, 对于如下请求访问序列: 88, 148, 92, 179, 90, 试问: 在采用 SSTF(最短寻道时间优先) 和 CSCAN(循环扫描, 磁头向高柱面号运动) 两种不同调度算法下, 各算法的平均寻道长度 (即平均每次磁头移动距离) 为多少? (给出访问顺序, 并计算)

答: SSTF: 访问顺序 148、179、92、90、88,

平均寻道长度 = $(4+31+87+2+2) / 5 = 25.2$;

SCAN: 访问顺序 148、179、88、90、92,

平均寻道长度 = $(4+31+91+2+2) / 5 = 26$