

1. 假定系统有三个并发进程 read, move 和 print 共享缓冲器 B1 和 B2. 进程 read 负责从输入设备上读信息, 每读出一个记录后把它存放到缓冲器 B1 中. 进程 move 从缓冲器 B1 中取出一记录, 加工后存入缓冲器 B2. 进程 print 将 B2 中的记录取出打印输出. 缓冲器 B1 和 B2 每次只能存放一个记录. 要求三个进程协调完成任务, 使打印出来的与读入的记录的个数, 次序完全一样.

请用 PV 操作, 写出它们的并发程序.

解: begin

semaphore : SR, SM<sub>1</sub>, SM<sub>2</sub>, SP

record : B<sub>1</sub>, B<sub>2</sub>

initialization :

SR = 1; // 缓冲区 B<sub>1</sub> 是否空闲

SM<sub>1</sub> = 0; // 缓冲区 B<sub>1</sub> 是否被占用

SM<sub>2</sub> = 1; // 缓冲区 B<sub>2</sub> 是否空闲

SP = 0; // 缓冲区 B<sub>2</sub> 是否被占用

cobegin

process read() {

record X;

while(1) { // 接收来自输入设备上的一个记录

X = 接收的一个记录;

P(SR);

B<sub>1</sub> = X;

V(SM<sub>1</sub>);

}

process move() {

record Y;

while(1) { // 加工并移动一个记录从 B<sub>1</sub> 到 B<sub>2</sub>

P(SM<sub>1</sub>);

Y = B<sub>1</sub>;

V(SR);

加工 Y;

P(SM<sub>2</sub>);

B<sub>2</sub> = Y;

V(SP);

}

process print() {

record Z;

while(1) { // 打印 B<sub>2</sub> 中的一个记录

P(SP);

Z = B<sub>2</sub>;

V(SM<sub>2</sub>);

打印 Z;

}

coend;

end;

T<sub>2</sub>

2. 两个并发执行的进程 A 和 B 的程序如下：

进程 A	进程 B
while(true)	while(true)
{	{
N=N+5;	打印 N 的值；
}	N=0;
	}

19

其中 N 为整数，初值为 4。若进程 A 先执行了三个循环后，进程 A 和进程 B 又并发执行了一个循环，写出可能出现的打印值。正确的打印值应该是多少？请用 P、V 操作进行管理，使进程 A 和 B 并发执行时不会出现与时间有关的错误。

解：A 执行三个循环后  $N = 4 + 5 \times 3 = 19$ ，并发执行有 2 种情况：

1. A 先执行，B 再执行，打印值为 24，执行后  $N = 0$ ；

2. B 先执行，A 再执行，打印值为 19，执行后  $N = 5$

2. 情况是错误的，正确打印值为 24

用 P、V 操作进行管理：

semaphore : S

initialization: S=1

进程 A

while (true) {

P(S);

$N = N + 5$ ;

V(S);

}

进程 B

while (true) {

P(S);

打印 N 的值；

$N = 0$ ;

V(S);

}

3. 假定在单道批处理环境下有 5 个作业，各作业进入系统的时间和估计运行时间如下表所示：

T3.

作业	进入系统时间	估计运行时间 / 分钟
----	--------	-------------

1	8:00	40
2	8:20	30
3	8:30	12
4	9:00	18
5	9:10	5

(1) 如果应用先来先服务的作业调度算法，试将下面表格填写完整。 *Waiting + Running*

作业	进入系统时间	估计运行时间 / 分钟	开始时间	结束时间	<u>周转时间 / 分钟</u>
1	8:00	40	8:00	8:40	40
2	8:20	30	8:40	9:10	50
3	8:30	12	9:10	9:22	52
4	9:00	18	9:22	9:40	40
5	9:10	5	9:40	9:45	35

$$\text{作业平均周转时间 } T = \frac{1}{5}(40+50+52+40+35) = 43.4 \text{ (min)}$$

(2) 如果应用最短作业优先的作业调度算法，试将下面表格填写完整。*(非抢占)*

作业	进入系统时间	估计运行时间 / 分钟	开始时间	结束时间	周转时间 / 分钟
1	8:00	40	8:00	8:40	40
2	8:20	30	8:40	9:10	62
3	8:30	12	9:10	9:22	22
4	9:00	18	9:22	9:40	45
5	9:10	5	9:40	9:45	17

$$\text{作业平均周转时间 } T = \frac{1}{5}(40+62+22+45+17) = 37.2 \text{ (min)}$$

4. 若在一分页存储管理系统中,某作业的页表如下所示.已知页面大小为 1024 字节,试将逻

T4.

辑地址 1011,2148,4000,5012 转化为相应的物理地址.

页号	物理块号
0	2
1	3
2	1
3	6

解: ①  $1011 \div 1024 = 0 \cdots \cdots 1011$ , 页号为 0, 物理块号 2, 偏移 1011

$$\text{物理地址: } 2 \times 1024 + 1011 = 3059$$

②  $2148 \div 1024 = 2 \cdots \cdots 100$ , 页号为 2, 物理块号 1, 偏移 100

$$\text{物理地址: } 1 \times 1024 + 100 = 1124$$

③  $4000 \div 1024 = 3 \cdots \cdots 928$ , 页号为 3, 物理块号 6, 偏移 928

$$\text{物理地址: } 6 \times 1024 + 928 = 7072$$

④  $5012 \div 1024 = 4 \cdots \cdots 916$ , 页号为 4, 超出页表长度, 产生越界.

- T5. 5. 在一个采用页式虚拟存储管理的系统中,有一用户作业,它依次要访问的字地址序列是:115,228,120,88,446,102,321,432,260,167,若该作业的第0页已经装入主存,现分配给该作业的主存共300字,页的大小为100字,请回答下列问题:

(1)按 FIFO 调度算法将产生 次缺页中断,依次淘汰的页号为 ,缺页中断率为 .

(2)按 LRU 调度算法将产生 次缺页中断,依次淘汰的页号为 ,缺页中断率为 .

解: 页大小为100字, 故需访问的地址页面号为

1, 2, 1, 0, 4, 1, 3, 4, 2, 1

主存共300字, 帧数为  $300 \div 100 = 3$  帧

初始时第0页已装入.

(1) FIFO :

1	2	1	0	4	1	3	4	2	1
0	0	0	0	4	4	4	4	4	4
1	1	1	1	1	1	3	3	3	3

缺页 缺页

缺页 缺页

缺页 缺页

调出0 调出1

调出2

产生5次缺页中断, 依次淘汰的页面为 0, 1, 2, 缺页中断率为 50%.

(2) LRU :

1	2	1	0	4	1	3	4	2	1
0	0	0	0	0	0	3	3	3	1
1	1	1	1	1	1	1	1	2	2

缺页 缺页

缺页 缺页

缺页 缺页

缺页 缺页

调出2

调出0

调出1

调出3

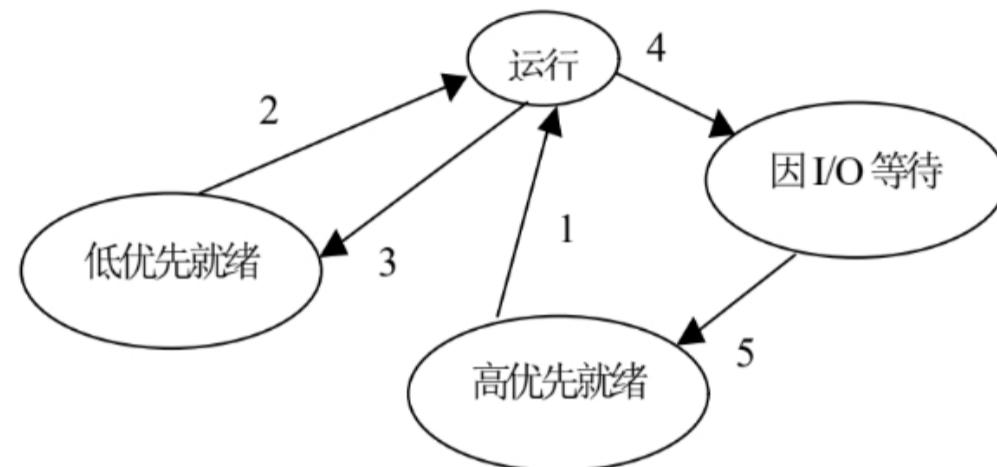
产生6次缺页中断, 依次淘汰的页面为 2, 0, 1, 3, 缺页中断率为 60%.

## T6. 某系统的进程状态图如下

(1)说明一个进程发生变迁1、3、4的原因是什么?

(2)下述因果变迁是否会发生?如果有可能的话,在什么情况下发生?

- A) 1-> 3    B) 2->4 ✗   C) 4->1    D) 5->1    E) 3->2



答：(1) 1: CPU空闲且高优先就绪队列中有进程，则从高优先就绪队列中调一个进程到CPU上执行。

3: 当一个正在CPU上运行的进程用完其时间片且未完成运行，则立即退出CPU进入低优先就绪队列。

4: 当前正在CPU上运行的进程需要输入或输出数据时，退出CPU进入等待队列。

(2) A, B 因果变迁不可能发生, C, D, E 可能发生, 原因分别为:

C (4→1): 当前正在CPU上运行的进程需要输入或输出数据时，退出CPU进入等待队列，CPU空闲。

此时若高优先就绪队列中有进程，则发生1。

D (5→1): 当高优先就绪队列和CPU都处于空闲状态时，一个处于等待队列的进程被唤醒进入高优先就绪队列后立即被调度到CPU上执行。

E (3→2): 当一个正在CPU上运行的进程用完其时间片且未完成运行，立即退出CPU进入低优先就绪队列，

若此时高优先就绪队列为空，则立即从低优先就绪队列调度一个进程到CPU中执行。即2。