

选择题

1. D 2. B 3. C 4. B 5. C 6. A 7. C 8. A 9. C 10. B
11. C 12. D 13. D 14. C 15. C

填空题

- 一次只允许一个进程使用的资源 进程中访问临界资源的那段程序代码
- 并发性 共享性 虚拟性 异步性
- 互斥 占有 非剥夺 循环等待
- 输入井 输出井
- 可供并发进程使用的资源实体数 正在等待使用临界区的进程数

SPOOLING 技术

临界资源

就绪 等待 运行

中断方式、DMA 方式、通道方式

字符流（流式）

.空闲块链、位示图

简答题

1. 答：所谓死锁是指多个进程在运行过程中因争夺资源而造成的一种僵局，当进程处于这种僵持状态时，若无外力作用，它们都无法再向前推进。

产生死锁的原因可归结为如下两点：竞争资源和进程推进顺序非法。

产生死锁的必要条件：互斥条件、请求和保持条件、不剥夺条件、环路等待条件。

2. 答：所谓虚拟存储器，是指具有请求调入功能和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统。其逻辑容量由内存容量和外存容量之和决定，其运行速度接近于内存速度，而每位的成本却又接近于外存。

局部性原理是指程序在执行时将呈现出局部性规律，即在一个较短的时间内，程序的执行仅局限于某个部分：

时间局限性：如果程序中的某条指令一旦执行，则不久以后该指令可能再次执行，如果某数据被访问过，则不久以后该数据可能再次被访问。

空间局限性：一旦程序访问了某个存储单元，在不久之后，其附近的存储单元将被访问，即程序在一段时间内所访问的地址，可能集中在一定的范围之内，其典型便是程序的顺序执行。

3. 答：当用户进程请求打印输出时，Spooling 系统同意打印输出，但并不真正把打印机分配给该用户进程，而只为它做两件事：1，由输出进程在输出井中为之申请一空闲盘块区，并将要打印的数据送入其中；2，输出进程再为用户进程申请一张空白的用户请求打印表，并将用户的打印要求填入表中，再将该表挂到请求打印队列之上。如果还有进程要求打印输出，系统仍可以接受该请求，同样做上面的工作。如果打印机空闲，输出进程将从请求打印队列的队首取出一张请求表，根据表中的要求将要打印的数据从输出井传送到内存缓冲区，再由打印机进行打印。打印完毕，输出进程再查看请求打印队列中是否还有等待要打印的请求表，若有，再取出一张表，并根据其中的要求进行打印，如此下去，直至请求队列为空位置，输出进程才将自己阻塞起来，等待下次再由打印请求时才被唤醒。

编程题

1. Begin

```
Var s,m:semaphore:=1,0;  
Int i,j:=0,0;
```

```

Cobegin
  Process  A()
    Begin
      Repeat
        P(s);
        Pick_up();
        V(m);
        I:=I+1;
      Until  I=n
    End,
  Process  B()
    Begin
      Repeat
        P(m);
        Pick_up();
        V(s);
        J:=j+1;
      Until  j=n
    End;
  Coend;
End.

```

2、 struct semaphore s, sp, so=1, 0, 0;

cobegin

void father (void)

```

{
    while(TRUE) {
        have an apple;
        P(s);
        put an apple;
        V(sp);}
}

```

void mother (void)

```

{
    while(TRUE) {
        have an orange;
        P(s);

```

```

        put an orange;
    V(so);}
void son (void)
{
    while(TRUE) {
        P(so);
        get an orange;
        V(s);
        eat an orange;}
}
void daught (void)
{
    while(TRUE) {
        P(sp);
        get an apple;
        V(s);
        eat an apple;}
}
coend

```

计算题

1. FIFO 物理块数为 3

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 3 | 2 | 1 | 4 | 3 | 5 | 4 | 3 | 2 | 1 | 5 |
| 4 | 4 | 4 | 1 | 1 | 1 | 5 | | | 5 | 5 | |
| | 3 | 3 | 3 | 4 | 4 | 4 | | | 2 | 2 | |
| | | 2 | 2 | 2 | 3 | 3 | | | 3 | 1 | |

缺页率：9/12*100%=75%

被淘汰页面的页号：4 3 2 1 4 3

FIFO 物理块数为 4

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 3 | 2 | 1 | 4 | 3 | 5 | 4 | 3 | 2 | 1 | 5 |
| 4 | 4 | 4 | 4 | | | 5 | 5 | 5 | 5 | 1 | 1 |
| | 3 | 3 | 3 | | | 3 | 4 | 4 | 4 | 4 | 5 |
| | | 2 | 2 | | | 2 | 2 | 3 | 3 | 3 | 3 |
| | | | 1 | | | 1 | 1 | 1 | 2 | 2 | 2 |

缺页率：10/12*100%=83.33%

被淘汰页面的页号：4 3 2 1 5 4

LRU 物理块数为 3

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 3 | 2 | 1 | 4 | 3 | 5 | 4 | 3 | 2 | 1 | 5 |
| 4 | 4 | 4 | 1 | 1 | 1 | 5 | | | 2 | 2 | 2 |
| | 3 | 3 | 3 | 4 | 4 | 4 | | | 4 | 1 | 1 |
| | | 2 | 2 | 2 | 3 | 3 | | | 3 | 3 | 5 |

缺页率：10/12*100%=83.33%

被淘汰页面的页号：4 3 2 1 5 4 3

LRU 物理块数为 4

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 3 | 2 | 1 | 4 | 3 | 5 | 4 | 3 | 2 | 1 | 5 |
| 4 | 4 | 4 | 4 | | | 4 | | | 4 | 4 | 5 |
| | 3 | 3 | 3 | | | 3 | | | 3 | 3 | 3 |
| | | 2 | 2 | | | 5 | | | 5 | 1 | 1 |
| | | | 1 | | | 1 | | | 2 | 2 | 2 |

缺页率：8/12*100%=66.67%

被淘汰页面的页号：2 1 5 4

2、FIFO：14 次

LRU：16 次

3、

4、先来先服务

| 执行次序 | 运行时间 | 优先数 | 等待时间 | 周转时间 |
|------|------|-----|------|------|
| A | 10 | 3 | 0 | 10 |
| B | 6 | 5 | 10 | 16 |
| C | 2 | 2 | 16 | 18 |
| D | 4 | 1 | 18 | 22 |
| E | 8 | 4 | 22 | 30 |

平均周转时间为：T=（10+16+18+22+30）/5=19.2min

最高级优先调度：

| 执行次序 | 运行时间 | 优先数 | 等待时间 | 周转时间 |
|------|------|-----|------|------|
| A | 10 | 3 | 0 | 6 |
| B | 6 | 5 | 6 | 14 |
| C | 2 | 2 | 14 | 24 |
| D | 4 | 1 | 24 | 26 |
| E | 8 | 4 | 26 | 30 |

平均周转时间为：T=（6+14+24+26+30）/5=20min

时间片轮转

平均周转时间为：T=（30+22+6+16+28）/5=20.4min

5、1.最短寻道时间优先算法 SSTF：248/9=27.56 道

2.电梯算法：250/9=27.78

6、先来先服务

| 被访问的下一磁道号 | 移动距离 |
|-----------|------|
| 55 | 45 |
| 58 | 3 |
| 39 | 19 |
| 18 | 21 |
| 90 | 72 |
| 160 | 70 |
| 150 | 10 |
| 38 | 112 |
| 184 | 146 |

平均寻道长度：55.3

最短寻道时间优先

| 被访问的下一磁道号 | 移动距离 |
|-----------|------|
| 90 | 10 |
| 58 | 32 |
| 55 | 3 |
| 39 | 16 |
| 38 | 1 |
| 18 | 20 |
| 150 | 132 |
| 160 | 10 |
| 184 | 24 |

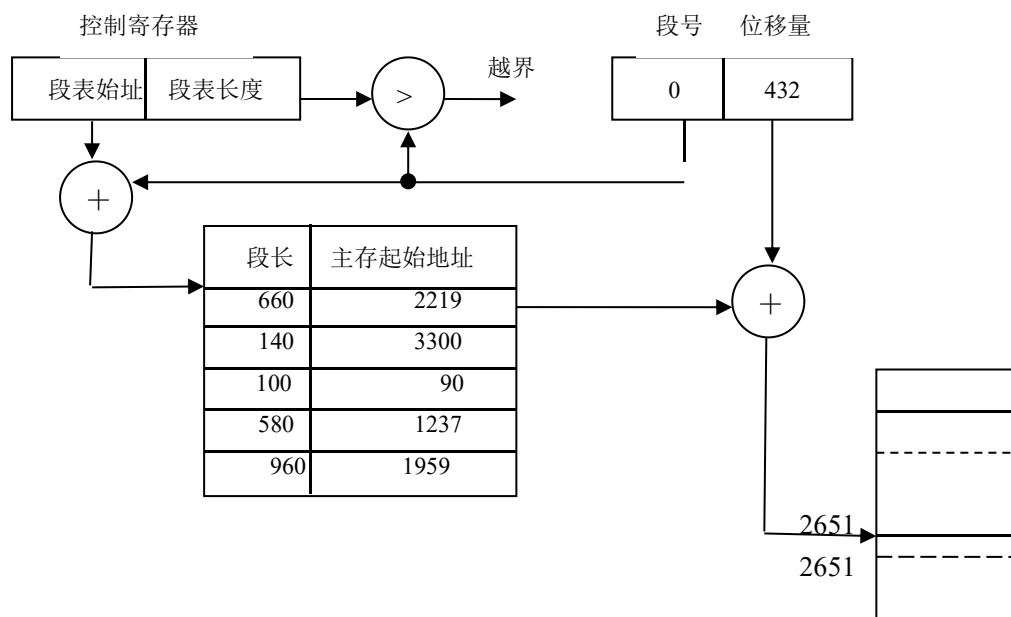
平均寻道长度：27.5

扫描算法

| 被访问的下一磁道号 | 移动距离 |
|-----------|------|
| 150 | 50 |
| 160 | 10 |
| 184 | 24 |
| 90 | 94 |
| 58 | 32 |
| 55 | 3 |
| 39 | 16 |
| 38 | 1 |
| 18 | 20 |

平均寻道长度：27.8

7、



2). [0, 432] 物理地址为 2651

[1, 10] 物理地址为 3310

[2, 500] 地址不合法, 产生地址越界

[3, 400] 物理地址为 1637

(3) 存取主存中的一条指令或数据至少要访问 2 次内存.

判断是否为安全状态, 关键在于能否找到一个安全序列. 这与进程剩余需求量有关, 列表如下

8、物理地址为 13057

综合题

1、

| 进程 | 最大资源数量 | | | 已分配资源数量 | | | 还需要资源数量 | | | 剩余资源数量 | | |
|----|--------|---|----|---------|---|---|---------|---|---|--------|---|---|
| | A | B | C | A | B | C | A | B | C | A | B | C |
| P1 | 5 | 5 | 9 | 2 | 1 | 2 | 3 | 4 | 7 | 2 | 3 | 3 |
| P2 | 5 | 3 | 6 | 4 | 0 | 2 | 1 | 3 | 4 | | | |
| P3 | 4 | 0 | 11 | 4 | 0 | 5 | 0 | 0 | 6 | | | |
| P4 | 4 | 2 | 5 | 2 | 0 | 4 | 2 | 2 | 1 | | | |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 | | | |

(2) Request4 (0, 3, 4) > Available (2, 3, 3), 系统不能给予满足。

(3) a: Request4 (2, 0, 1) < Need4 (2, 2, 1)

b: Request4 (2, 0, 1) < Available (2, 3, 3)

C: 系统试探将 Request4 (2, 0, 1) 分配出去并修改数据结构中的值

| 进程 | 最大资源数量 | | | 已分配资源数量 | | | 还需要资源数量 | | | 剩余资源数量 | | |
|----|--------|---|---|---------|---|---|---------|---|---|--------|---|---|
| | A | B | C | A | B | C | A | B | C | A | B | C |
| P1 | 5 | 5 | 9 | 2 | 1 | 2 | 3 | 3 | 4 | 7 | 0 | 3 |
| P2 | 5 | 3 | 6 | 4 | 0 | 2 | 1 | 3 | 4 | | | |

| | | | | | | | | | |
|----|---|---|----|---|---|---|---|---|---|
| P3 | 4 | 0 | 11 | 4 | 0 | 5 | 0 | 0 | 6 |
| P4 | 4 | 2 | 5 | 4 | 0 | 5 | 0 | 2 | 0 |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 |

此时可以寻找到安全序列<p4,p5,p1,p2,p3>,此次请求可以分配

(4) a: Request1 (0, 2, 0) < Available(0,3,2)

b: Request1(0,2,0)<Need(3,4,7)

C:试探分配，此时系统中的情况如下表所示：

| 进程 | 最大资源数量 | | | 已分配资源数量 | | | 还需要资源数量 | | | 剩余资源数量 | | |
|----|--------|---|----|---------|---|---|---------|---|---|--------|---|---|
| | A | B | C | A | B | C | A | B | C | A | B | C |
| P1 | 5 | 5 | 9 | 2 | 3 | 2 | 3 | 2 | 7 | 0 | 1 | 2 |
| P2 | 5 | 3 | 6 | 4 | 0 | 2 | 1 | 3 | 4 | | | |
| P3 | 4 | 0 | 11 | 4 | 0 | 5 | 0 | 0 | 6 | | | |
| P4 | 4 | 2 | 5 | 4 | 0 | 5 | 0 | 2 | 0 | | | |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 | | | |

d: 剩余资源不足，所以不存在安全序列。此次分配不予满足。

2、

1) P4、P5、P1、P2、P3

2) 因可用的资源的数量不够，只能推迟分配。

3) P4、P5、P1、P2、P3

4) 系统不予以分配，因为分配出去后，找不到一个安全序列。