

## 2017-2018-2A 答案及评分标准

一、选择题（每题 1 分，共 25 分）          得分：

1. C	2. A	3.A	4.D	5.B	6.C	7.A	8.C	9.D	10.A
11. C	12.C	13.C	14.B	15.A	16.B	17.A	18.C	19.A	20.B
21. A	22.C	23.D	24.A	25.D					

二、综合题：

1、（10 分）

答：

（1）（5 分）什么时候调入页面：可以采用请求调页策略与预调页策略相结合的方式：请求调页策略是不提前调入，当发生缺页时把所缺页面调入内存，保证调入的页面就是要访问的，避免调入无用页面占据内存空间，提高内存利用率；在调入所缺页面的同时，依据进程运行的局部性原理，将该页面后续临近的若干页面（如 8 个页面）同时调入内存，当进程之后要访问这些页面时就不会发生缺页了，从而降低缺页率。

评分标准：策略合理 2 分，解释完整合理 3 分

（2）（5 分）从哪里调入：①如果系统有足够的对换区空间，则在进程开始运行前，所有页面全部从文件区复制到对换区，且尽量连续存放，进程所缺页面全部从对换区调入，以提高调页速度；②如果没有足够的对换区，则仅将进程的可能会被修改的页面存放在对换区，不会被修改的页面保留在文件区，当进程缺页时，可根据页面特征分别从文件区或者对换区调入。换出时，如果不会被修改的页面，则直接淘汰，不必启动磁盘 I/O 操作，以后要使用这些页面时，再从文件区调入。对于已修改的淘汰页，换出到对换区，以后需要时再从对换区调入。③此外，还可以引入缓冲思想，对于淘汰页，可以让其在内存中再驻留一段时间，如果之后很好被访问，则在内存中能找到这个页面，减少磁盘 I/O 启动次数。

评分标准：3 种思路中答对两个就满分。

2、（12 分）

答：

（1）（2 分）J3 页表如下所示：

页号	块号
0	3
1	4
2	6
3	8

（2）（7 分）要访问的逻辑地址有 3 个：500,1500,3100          （1 分）

每个地址转换 2 分

逻辑地址 500: 页号=500/1024=0, 页内位移=500%1024=500

查找页表, 得块号 3, 物理地址=3\*1024+500=3572

逻辑地址 1500: 页号=1500/1024=1, 页内位移=1500%1024=476

查找页表, 得块号 4, 物理地址=4\*1024+476=4572

逻辑地址 3100: 页号=3100/1024=3, 页内位移=3100%1024=28

查找页表, 得块号 8, 物理地址=8\*1024+28=8220

(3) (3 分) 由于没有缺页, 对每个地址的内存访问时间相同, 访问逻辑地址 500:

(1-75%) \*100+100=125ns

所以 3 个逻辑地址的访存时间: 125ns\*3=375ns

### 3、(8 分)

答: 主要是中断, 包括时钟中断, 有了计时功能; 还有其他类型的中断;

还有就是为了实现虚拟存储器引入的页表机制

主要分析以上两个方面的支持, 各占 4 分。

### 4、(12 分) 答:

(1) (2 分) 互斥关系: 三个并发进程 R、M、P 互斥访问可循环使用的缓冲区 B。

同步关系:

(a) 缓冲区有空, R 进程才能存入

(b) 缓冲区有 R 进程读入的字符, M 进程才能处理

(c) 缓冲区有 M 进程处理之后的字符, P 进程才能取出打印

(2) 伪代码描述如下:

semaphore empty=N; //缓冲区互斥操作信号量

semaphore full1=0; // 表示已经读入的字符个数, 初值为 0

semaphore full2=0; // 表示已经处理的字符个数, 初值为 0

semaphore mutex=1; // 互斥访问缓冲区

char buffer[N]; // 缓冲区

int in=0,out1=0,out2=0; // in 指向下一个可用缓冲单元, out1 指向下一个待处理字符, out2 指向下一个待输出的字符。(2 分)

(3)

R( )	M( )	P( )
<pre>{     While(true)     {         char x;         读入一个字符到 x;         P(empty);         P(mutex);         Buffer[in]=x;         in=(in+1)%N;         V(mutex);         V(full1);     } } (3 分)</pre>	<pre>{     char x;     while (true) {         P(full1);         P(mutex);         x=buffer[out1];         if(x==" ") {             x=",";             buffer[out1]=x;         }         out1=(out1+1)%N;         V(mutex);         V(full2);     } } (3 分)</pre>	<pre>{     Char x;     While(true)     {         P(full2);         P(mutex);         x=buffer[out2];         out2=(out2+1)%N;         V(mutex);         V(empty);         输出字符 x;     } } (2 分)</pre>

### 5、(11 分)

答：(1) 系统在选择进入内存就绪的队列的作业时，采用高响应比优先调度算法；当两个作业的进程在内存中进行进程调度时，采用短进程优先调度算法。

10: 00 时，作业 J1 到达，进入内存，进程运行。

10: 00 时，作业 J2 到达，进入内存。这时 J1 还剩 25 分钟，因此 J1 继续运行。

10: 15~10: 30，作业 J3、J4、J5 陆续到达，内存中已经有两道作业，因此这些作业不能进入内存。

10: 35 时，作业 J1 运行完成，再利用作业调度算法选择一个作业进入内存。这时各作业响应比分别为： $R_3=1+20/45$ ; $R_4=1+15/20$ ; $R_5=1+5/30$ 。

因此，J4 的响应比最高，系统作业调度程序选择 J4 进入内存。J2 和 J4 相比，J4 是短作业，因此 J4 先运行。(2 分)

10: 55 时，作业 J4 运行完成。再利用作业调度算法选择一个作业进入内存。这时各作业的响应比分别为： $R_3=1+40/45$ ; $R_5=1+25/30$ 。作业 J3 响应比高，先进入内存。这时内存中有两道作业，即 J2 和 J3 为短作业，先运行。

11: 25 时，作业 J2 运行完成。作业 J5 进入内存。这时内存中有两道作业，即 J3 和 J5。J5 为短作业，先运行。

11: 55 时，作业 J5 运行完成。

12: 40 时，作业 J3 运行完成。(4 分)

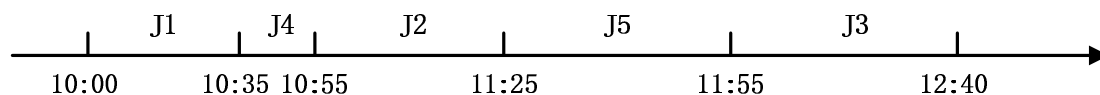


图 (3 分)

(2) 各作业的周转时间分别为：

$T_1=10:35-10:00=35$  分钟

$T_2=11:25-10:10=75$  分钟

$T_3=12:40-10:15=145$  分钟

$T_4=10:55-10:20=35$  分钟

$T_5=11:55-10:30=85$  分钟

平均周转时间= $375/5=75$  分钟 (4 分)

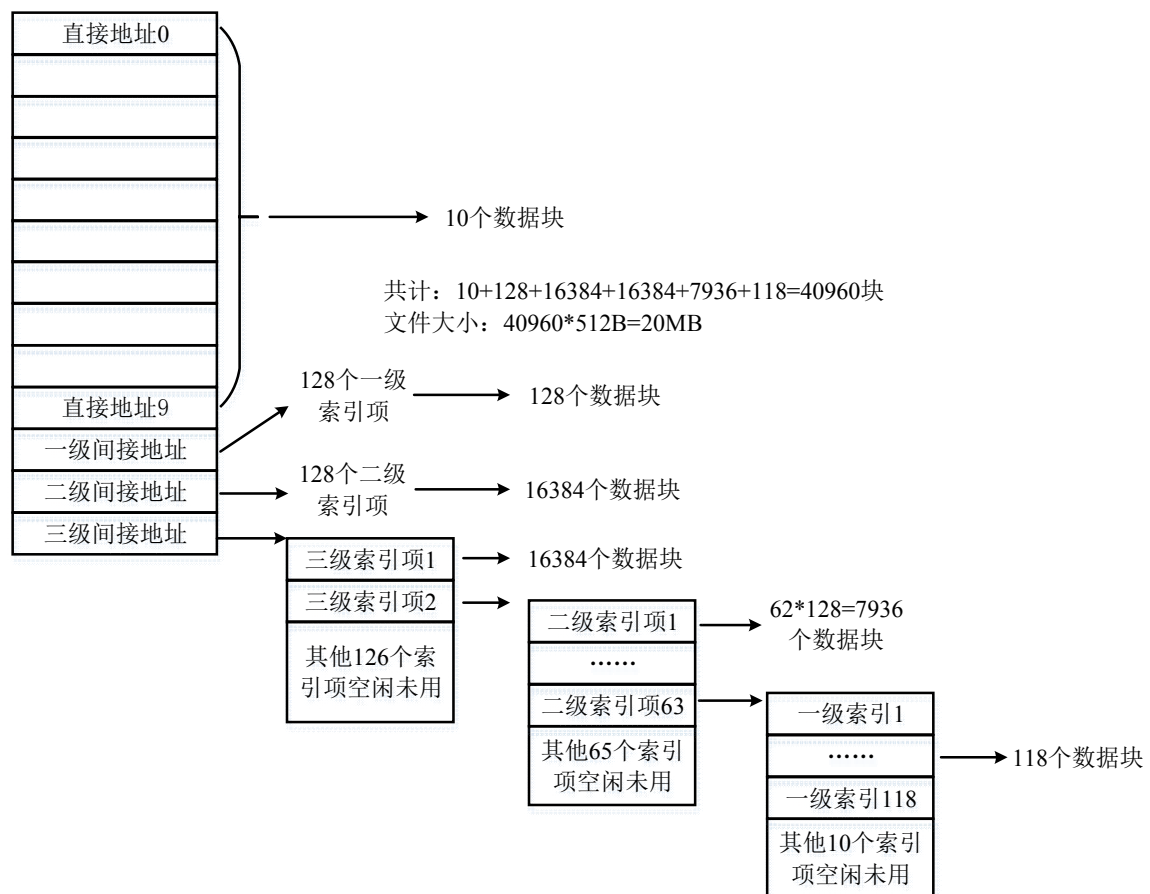
6、(12 分) (1)  $(10+128+128^2+128^3) \times 512=1082201088=1.007\text{GB}$  (1 分)

(2)  $20\text{MB}/512\text{B}=40960$  个盘块 (1 分)，(数据块)

所以使用 10 个直接数据块，128 个一级数据块，16384 个二级数据块， $40960-16522=24438$  个三级数据块 (2 分)。

$24438-16384=8054$ ， $8054/128=62.9 \approx 63$ ，所以用到三级索引地址中两个地址项，第二个地址项的二级索引中用到 63 个地址项，第 63 个二级地址项指向 118 个一级地址项，指向 118 个数据块。(2 分)

图 3 分



(3)  $12345/512=24.11$ ,  $(12345+500)/512=25.08$ , 要读的数据在文件的 25、26 块上。一级间接地址块上第 15、16 索引项指向这两个数据块。因此 read 系统调用需要先访问一级间接索引块, 然后访问两个数据块, 共需要 3 次外存访问。(3 分)

7、(10 分) 答: 目录项 64B, 每个盘块可以放  $512/64=8$  个目录项 (2 分)。在 254 个目录项中查找一个文件需要查找平均 127 个项目 (2 分)。

分离前,  $127/8=16$ , 所以要读取 16 个磁盘块, 即 16 次访问磁盘。(2 分)

分离后,  $512/10=51.2$ , 每个磁盘块可以存放 51 个目录项 (2 分),  $127/51 \approx 3$ , 所以平均访问 2 次磁盘, 再读入 i 节点, 共需访问 3 次磁盘。(2 分)