

09 年真题

选择题:

【23】单处理机系统中,可并行的是()。

- I.进程与进程 II.处理机与设备
III.处理机与通道 IV.设备与设备
A. I、II和III B. I、II和IV
C. I、III和IV D. II、III和IV

【解析】D。单处理机即只有一个处理机(此处不包含多核的情况),某时刻处理机同时只能处理一个进程,所以进程与进程之间不能并行执行。处理机、通道、设备都能并行执行,比如同时打印(设备)、计算(处理机)、传输数据(通道控制内存与外存间数据交换)。

【24】下列进程调度算法中,综合考虑进程等待时间和执行时间的是()。

- A.时间片轮转调度算法 B.短进程优先调度算法
C.先来先服务调度算法 D.高响应比优先调度算法

【解析】D。高响应比优先调度算法中,计算每个进程,响应比最高的先获得CPU,响应比计算公式:响应比=(进程执行时间+进程等待时间)/进程执行时间。高响应比优先调度算法综合考虑到了进程等待时间和执行时间,对于同时到达的长进程和短进程相比,短进程会优先执行,以提高系统吞吐量;当某进程等待时间较长时,其优先级会提高并很快得到执行,不会产生有进程调度不到的情况。

补充:

时间片轮转调度算法(RR)使每个进程都有固定的执行时间,但对于长进程来说,等待时间也相对较长。

短进程优先调度算法(SJF/SPF)对于短进程有很大的优势,但对于长进程来说,如果不断有短进程请求执行,则会长期得不到调度。

先来先服务(FCFS)实现最简单,但如果有一个长进程到达之后,会长期占用处理机,使后面到达的很多短进程得不到运行。

【25】某计算机系统中有8台打印机,有K个进程竞争使用,每个进程最多需要3台打印机,该系统可能会发生死锁的K的最小值是()。

- A.2 B.3 C.4 D.5

【解析】C。假设 $k=3$,3个进程共享8台打印机,每个进程最多可以请求3台打印机,若3个进程都分别得到2台打印机,系统还剩下2台打印机,接下去无论哪个进程申请打印机,都可以得到满足,3个进程都可以顺利执行完毕,这种情况下不会产生死锁。假设 $k=4$,4个进程共享8台打印机,都得不到满足,产生了互相等待,可能会发生死锁。

补充:

这类型的题通常数字不大,可以凭经验尝试出正确答案,这样会快一些,不过还是给出对应计算公式,以便大家加深理解和满足对于题目的成就感。假设 n 为每个进程所需的资源数, m 为进程数, A 为系统的资源数。则满足 $(n-1) \times m \geq A$ 的最小整数 m 即为可能产生死锁的最小进程数,该公式同样可以用于求出每个进程需要多少资源时可能会产生死锁。

该公式可以这样理解:当所有进程都差一个资源就可以执行,此时系统中所有资源都已经分配,因此死锁。

【26】分区分配内存管理方式的主要保护措施是()。

- A.界地址保护 B.程序代码保护
C.数据保护 D.栈保护

【解析】A、B、C、D 都不是存储保护的措施。使用界限寄存器可以实现分区分配内存管理的存储保护。概念题，记住概念即可。

【27】一个分段存储管理系统中，地址长度为 32 位，其中段号占 8 位，则最大段长是（ ）。
A. 2^8 字节 B. 2^{16} 字节 C. 2^{24} 字节 D. 2^{32} 字节

【解析】C。段式存储管理系统中的逻辑地址分成两部分：段号和位移量。段号 8 位，则位移量用 24 位来表示，位移量的大小决定了每段的长度，所以每段的最大长度是 2^{24} 。通常系统都是以字节作为一个分配单元，因此最大段长为 2^{24} 字节。

【28】下列文件物理结构中，适合随机访问且易于文件扩展的是（ ）。

- A.连续结构 B.索引结构
C.链式结构且磁盘块定长 D.链式结构且磁盘块变长

【解析】B。根据外存储分配方法，链式存储结构将文件按照顺序存储在不同盘块中，因此适合顺序访问，不适合随机访问（需从文件头遍历所有盘块）；连续结构（数据位置可计算得到）和索引存储结构（只需访问索引块即可知道数据位置）适合随机访问，但连续存储结构如果要在中间增加数据，则要整体移动后面的所有数据，因此不适合文件的动态增长，而索引存储结构适合随机访问，因为索引结构可以单独将新增数据放在一个新盘块，只需修改索引块即可。

【29】假设磁头当前位于第 105 道,正在向磁道序号增加的方向移动.现有一个磁道访问请求序列为 35,45,12,68,110,180,170,195,采用 SCAN 调度(电梯调度)算法得到的磁道访问序列是（ ）。

- A.110,170,180,195,68,45,35,12 B.110,68,45,35,12,170,180,195
C.110,170,180,195,12,35,45,68 D.12,35,45,68,110,170,180,195

【解析】A。SCAN 算法的基本思想：磁头从磁盘的一端开始向另一端移动，沿途响应访问请求，直到到达了磁盘的另一端，此时磁头反方向移动并继续响应服务请求，其运动轨迹类似于电梯的运行。根据 SCAN 算法，可以得到访问序列是 A。

【30】文件系统中，文件访问控制信息存储的合理位置是（ ）。

- A.文件控制块 B.文件分配表
C.用户口令表 D.系统注册表

【解析】A。文件系统为实现“按名存取”，为每个文件设置用于描述和控制文件的数据结构，这个数据结构称为文件控制块 FCB。文件控制块 FCB 包含文件名，文件物理地址，访问权限等信息。

补充：

文件分配表（FAT）是系统用来标识盘块的文件归属，即哪些盘块属于哪些文件，并不包含文件的访问权限信息。

用户口令表是多用户系统中，用于记录不同用户口令的存储表，于文件访问权限无关。

系统注册表是用来储存系统相关设置的，例如系统的环境以及软件的设置等等，并不包括文件访问控制信息。

【31】设文件 F1 的当前引用计数值为 1，先建立 F1 的符号链接（软链接）文件 F2，再建立 F1 的硬链接文件 F3，然后删除 F1。此时，F2 和 F3 的引用计数值分别是（ ）。

- A.0、1 B.1、1 C.1、2 D.2、1

【解析】B。在 Unix/Linux 系统中，创建符号链接文件时，该文件有自己的 inode 结构，而硬链接文件和目标文件共享 inode 结构；在删除文件时，引用计数器减 1，当引用计数器为 0 时，才真正删除，并释放 inode 结构。创建文件 F2 后，其文件引用计数器为 1；创建文

件 F3 后，其文件引用计数器为 2，删除文件 F1，引用计数器减 1，此时文件 F3 引用计数器为 1。所以 F2 和 F3 两个文件引用计数器都是 1。

【32】程序员利用系统调用打开 I/O 设备时，通常使用的设备标识是（ ）。

A.逻辑设备名 B.物理设备名 C.主设备号 D.从设备号

【解析】A。在操作系统的设备管理中，用户程序不直接使用物理设备名（或设备的物理地址），而使用逻辑设备名；系统在实际执行时，将逻辑设备名转换为某个具体的物理设备名，实施 I/O 操作。逻辑设备是实际物理设备属性的抽象，它并不限于某个具体设备。

解答题：

【45】三个进程 P1、P2、P3 互斥使用一个包含 $N(N>0)$ 个单元的缓冲区。P1 每次用 `produce()` 生成一个正整数并用 `put()` 送入缓冲区某一空单元中；P2 每次用 `getodd()` 从该缓冲区中取出一个奇数并用 `countodd()` 统计奇数个数；P3 每次用 `geteven()` 从该缓冲区中取出一个偶数并用 `counteven()` 统计偶数个数。请用信号量机制实现这三个进程的同步与互斥活动，并说明所定义的信号量的含义。要求用伪代码描述。

【解析】本题考察了进程的同步与互斥

（1）缓冲区是互斥资源，因此设互斥信号量为 `mutex`。

（2）同步问题：P1，P2 为奇数的放置与取用而同步，设同步信号量为 `odd`；P1，P3 因为偶数的放置与取用而同步，设同步信号量为 `even`；对空闲缓冲区设置资源同步信号量 `empty`，初值为 `N`。伪代码描述如下：

```
semaphore mutex=1;
semaphore odd=0;even=0 ;
semaphore empty=N;
main()
{
    process P1
    while(true)
    {
        number=produce();
        p(empty);           //这里切忌要先进行 p(empty); 再进行 p(mutex)! 不然可能会
        p(mutex);           //导致死锁! (否则当没有空余缓冲区时，P1 将持续占用 mutex)
        put();
        v(mutex);
        if(number%2==0)
            v(even);
        else
            v(odd);
    }
    process P2
    while(true)
    {
        p(odd);
        p(mutex);
        getodd();
        v(mutex);
    }
    process P3
    while(true)
    {
        p(even);
        p(mutex);
        geteven();
        v(mutex);
    }
}
```

```

    v(empty);
    countodd();
}
process P3
while()
{
    p(even);
    p(mutex);
    geteven();
    v(mutex);
    v(empty);
    counteven();
}
}

```

【46】请求分页管理系统中，假设某进程的页表内容如下表所示

页表内容

页号	页框 (Page Frame) 号	有效位 (存在位)
0	101H	1
1	—	0
2	254H	1

页面大小为 4KB，一次内存的访问时间是 100ns，一次快表(TLB)的访问时间是 10ns，处理一次缺页的平均时间为 10^8ns (已含更新 TLB 和页表的时间)，进程的驻留集大小固定为 2，采用最近最少使用置算法(LRU)和局部淘汰策略。假设①TLB 初始为空；②地址转换时先访问 TLB，若 TLB 未命中，再访问页表（忽略访问页表之后的 TLB 更新时间）；③有效位为 0 表示页面不在内存，产生缺页中断，缺页中断处理后，返回到产生缺页中断的指令处重新执行。设虚地址访问序列 2362H、1565H、25A5H，请问：

(1) 依次访问上述三个虚地址，各需多少时间？给出计算过程。

(2) 基于上述访问序列，虚地址 1565H 的物理地址是多少？请说明理由。

【解析】本题考查页式存储器管理的相关内容，包括页表项计算及缺页中断处理等。

(1) 因为每页大小为 4KB，页内位移为 12 位（二进制位）或 3 位（十六进制位），逻辑地址 2362H 的页号为第一个 16 进制位（也可转化为二进制，得 0002 0011 0110 0010B，则对应的页号为前四个二进制位），即为 2，通过查询页表得知，该页在内存，但初始 TLB 为空，因此 2362H 的访问时间为 10ns（访问 TLB）+ 100ns（访问页表）+ 100ns（访问内存单元）= 210ns。

同理逻辑地址 1565H 对应的页号为 1，该页不在内存，出现缺页中断，缺页中断处理后，返回到产生缺页中断的指令处重新执行，需要访问一次 TLB。所以，1565H 的访问时间为 10ns（访问 TLB）+ 100ns（访问页表）+ 10^8ns （调页）+ 10ns（访问 TLB）+ 100ns（访问内存单元） $\approx 10^8\text{ns}$ 。

逻辑地址 25A5H 对应的页号为 2，该页在内存，TLB 命中，所以，25A5H 的访问时间=10ns（访问 TLB）+ 100ns（访问内存单元）= 110ns。

(2) 1565H 对应的物理地址是 101565H；在上一题中，当访问 1565H 时产生缺页中断时，内存中已经有 2 页，达到驻留集大小上限，应选出一页换出，由于在之前刚访问过 2362H 地址，2 号页面刚被访问，因此根据 LRU 算法，淘汰 0 号页面，即页框号为 101H 的页面，

因此换入的 1 号页面的页框号为 101H，所以虚地址 1565H 的物理地址为页框号 101H 与页内位移 565H 的组合，即 101565H。

10 年真题

选择题:

【23】下列选项中，操作系统提供给应用程序的接口是（ ）。

- A.系统调用 B.中断 C.库函数 D.原语

【解析】A。操作系统提供两类接口，一类是命令接口，比如用户通过键盘命令和鼠标命令来操作计算机；另一类是程序接口，它提供一组系统调用，用户可以通过运行一些应用程序来访问操作系统的资源。

补充:

实际上，本题四个选项中，只有 A 是操作系统提供的接口。系统提供封装好的系统调用供应用程序使用，应用程序无需考虑系统底层的内容，仅考虑上层的操作即可。

中断是系统内部对于事件响应的机制，对于应用程序来说是透明的，不会提供给应用程序直接使用。

同样库函数和原语都是面对操作系统底层的，不会直接提供给应用程序。

【24】下列选项中，导致创建新进程的操作是（ ）。

- I . 用户登陆成功 II . 设备分配 III . 启动程序执行
A.仅 I 和 II B.仅 II 和 III C.仅 I 和 III D. I 、 II 、 III

【解析】C。分析：用户登录成功就是需要为这个用户创建进程来解释用户的各种命令操作；设备分配由内核自动完成，不需要创建新进程；启动程序执行的目的是创建一个新进程来执行程序。

【25】设与某资源相关联的信号量初值为 3，当前值为 1，若 M 表示该资源的可用个数，N 表示等待该资源的进程数，则 M,N 分别是（ ）。

- A.0,1 B.1,0 C.1,2 D.2,0

【解析】B。分析：信号量可以用来表示某资源的当前可用数量。当信号量 $K > 0$ 时，表示此资源还有 K 个资源可用，此时不会有等待该资源的进程。而当信号量 $K < 0$ 时，表示此资源有 $|K|$ 个进程在等待该资源。此题中信号量 $K=1$ ，因此该资源可用数 $N=1$ ，等待该资源的进程数 $M=0$ 。

【26】下列选项中，降低进程优先权级的合理时机是（ ）。

- A.进程的时间片用完 B.进程刚完成 I/O, 进入就绪队列
C.进程长期处于就绪队列 D.进程从就绪状态转为运行状态

【解析】A。分析：B 选项中，进程完成 I/O 后，进入就绪队列时，已经是优先级最低的进程，不能再降低其优先级，为了让其及时处理 I/O 结果，也应该提高优先级；C 中，进程长期处于就绪队列，也需要增加优先级，使其不至于产生饥饿（所谓饥饿就是进程长期得不到处理机，无法执行）；D 中，当进程处于运行状态时，不可提高或降低其优先级。而 A 选项中，采用时间片算法处理进程调度时，如果进程时间片用完，则需要排到就绪队列的末尾，也就是优先级最低，所以降低优先级的合理时机是时间片用完时。另外如果采用多级反馈调度算法时，当时间片用完，进程还未结束，则放到下一级队列中。

【27】进行 P0 和 P1 的共享变量定义及其初值为（ ）。

```
boolean flag[2];  
int turn=0;  
flag[0]=false; flag[1]=false;
```

若进行 P0 和 P1 访问临界资源的类 C 代码实现如下：

```

void p0() // 进程 p0
{ while (TRUE)
    { flag[0]=TRUE; turn=1;
      While (flag[1]&&(turn==1))
        临界区 ;
      flag[0]=FALSE;
    }
}

void p1() // 进程 p1
{ while (TRUE)
    { flag[0]=TRUE; turn=0;
      While (flag[0]&&(turn==0));
        临界区 ;
      flag[1]=FALSE;
    }
}

```

并发执行进程 P0 和 P1 时产生的情况是

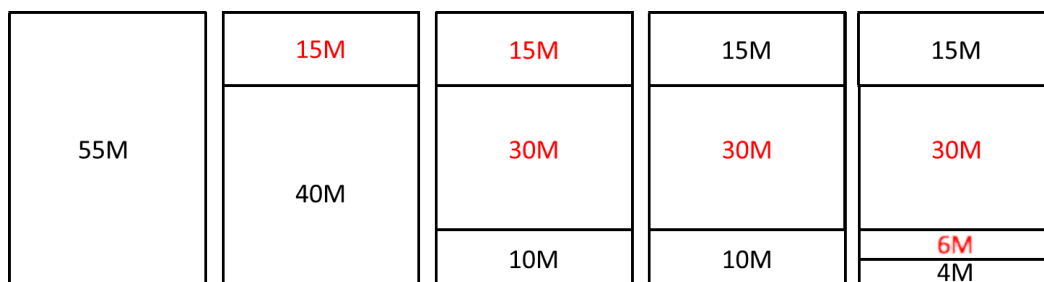
- A.不能保证进程互斥进入临界区，会出现“饥饿”现象
- B.不能保证进程互斥进入临界区，不会出现“饥饿”现象
- C.能保证进程互斥进入临界区，会出现“饥饿”现象
- D.能保证进程互斥进入临界区，不会出现“饥饿”现象

【解析】D。分析：在之前没有了解过该算法的情况下，临场想要判断出来还是很难的，尤其本题的算法是一个可以保证进程护持进入临界区、不会出现“饥饿”现象的良好算法，在考虑的时候，没法找出破绽，却还容易误认为自己的想法不够全面。实际上，该算法满足互斥、空闲让进、有限等待这三条同步机制准则，可以达到比较好的同步效果。对于此种类型的题目，在复习时候可以多分析一些经典的互斥算法。本题为 Peterson 算法，此外还有 Dekker 算法等，读者可以自己找来进行分析理解。

【28】某基于动态分区存储管理的计算机，其主存容量为 55Mb（初始为空），采用最佳适配（Best Fit）算法，分配和释放的顺序为：分配 15Mb，分配 30Mb，释放 15Mb，分配 6Mb，此时主存最大空闲分区的大小是（ ）。

- A.7Mb
- B.9Mb
- C.10Mb
- D.15Mb

【解析】D。分析：采用最佳适配方法，就是每次只找最小且能满足所需分配大小的分区。下图表示整个分区的过程。



（其中红色字体部分为已分配，黑色字体部分为尚未分配）

由此可知最后主存的最大空间为 15MB。

【29】某计算机采用二级页表的分页存储管理方式，按字节编制，页的大小为 2^{10} 字节，页表项大小为 2 字节，逻辑地址结构为：

页目录号	页号	页内偏移量
------	----	-------

逻辑地址空间大小为 2^{16} 页，则表示整个逻辑地址空间的页目录表中包含表项的个数至少是（ ）。

- A.64
- B.128
- C.256
- D.512

【解析】B。分析：在内存中有些页是存储页表项的，相当于页的索引，本题的目的就是要计算包含索引项的页的数量至少是多少。由题可知每页的大小为 $2^{10}B$ ，就是 1024B，每个页表项的大小为 2B，那么每页中可以存放 512 个页表项信息，逻辑地址总共有 2^{16} 页，

那么要储存全部 2^{16} 个页表项所需要的页数量为 $2^{16}/512=2^7=128$ 页，即为答案。

【30】设文件索引节点中有 7 个地址项，其中 4 个地址为直接地址索引，1 个地址项是二级间接地址索引，每个地址项的大小为 4 字节，若磁盘索引块和磁盘数据块大小均为 256 字节，则可表示的单个文件最大长度是（ ）。

- A.33KB B.519KB C.1057KB D.16513KB

【解析】C。分析：根据题目，每个磁盘索引块有 $256/4=64$ 个地址项。直接地址索引指向数据块的大小为 $4 \times 256 = 1\text{KB}$ ；一级间接地址索引数据块的大小为 $2 \times 64 \times 256 = 32\text{KB}$ ；二级间接地址索引数据块的大小为 $64 \times 64 \times 256=1024\text{KB}$ ；所以单个文件的最大长度是： $1\text{KB} + 32\text{KB} + 1024\text{KB} = 1057\text{KB}$ 。

【31】设当前工作目录的主要目的是（ ）。

- A.节省外存空间 B.节省内存空间
C.加快文件的检索速度 D.加快文件的读写速度

【解析】C。分析：设置当前目录的作用就是为了加快文件的检索速度，因为设置了当前目录情况下不需要从根目录出发进行检索，只需要从当前目录出发即可。至于 A、B 节省空间和此无关。文件的读写速度仅和硬件本身以及文件存储方式有关，与目录无关。

【32】本地用户通过键盘登陆系统是，首先获得键盘输入信息的程序时（ ）。

- A.命令解释程序 B.中断处理程序 C.系统调用程序 D.用户登录程序

【解析】B。分析：通过键盘登录系统时，必然需要进行键盘的录入，在录入键盘命令时，系统首先会产生中断，对键盘录入的信息进行存储，待键盘录入完毕后，再进行整体登录信息的处理。因此，首先获得键盘输入信息的程序是中断处理程序。

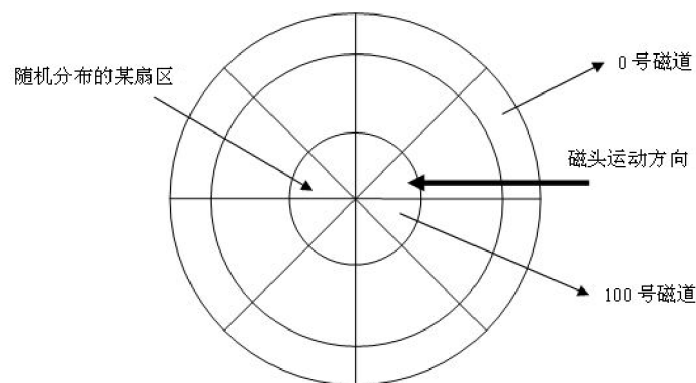
解答题：

【45】假设计算机系统采用 CSCAN（循环扫描）磁盘调度策略，使用 2KB 的内存空间记录 16384 个磁盘的空闲状态。

（1）请说明在上述条件如何进行磁盘块空闲状态的管理。

（2）设某单面磁盘的旋转速度为每分钟 6000 转，每个磁道有 100 个扇区，相邻磁道间的平均移动的时间为 1ms。若在某时刻，磁头位于 100 号磁道处，并沿着磁道号增大的方向移动（如下图所示），磁道号的请求队列为 50、90、30、120，对请求队列中的每个磁道需读取 1 个随机分布的扇区，则读完这个扇区点共需要多少时间？需要给出计算过程。

（3）如果将磁盘替换为随机访问的 Flash 半导体存储器（如 U 盘、SSD 等），是否有比 CSCAN 更高效的磁盘调度策略？若有，给出磁盘调度策略的名称并说明理由；若无，说明理由。



【参考答案】

(1) 用位示图表示磁盘的空闲状态，每一位表示一个磁盘块的空闲状态，共需要 $16384/8=2048$ 字节=2KB。系统提供的 2KB 内存正好能表示着 16384 个磁盘块。

(2) 采用 CSCAN 调度算法，磁头运动方向为磁道增大方向，由于当前磁头位于 100 号磁道处，则访问磁道的顺序为 120、30、50、90，则磁头移动磁道长度为 $20+90+20+40=170$ ，总的移动磁道时间为 $170 \times 1ms=170ms$ 。由于转速为 6000 转/分，则读取一个磁道上的一个扇区的平均时间为 $(60/6000)/100s=0.1ms$ ，总的读取扇区的时间为 $4 \times 0.1ms=0.4ms$ 。

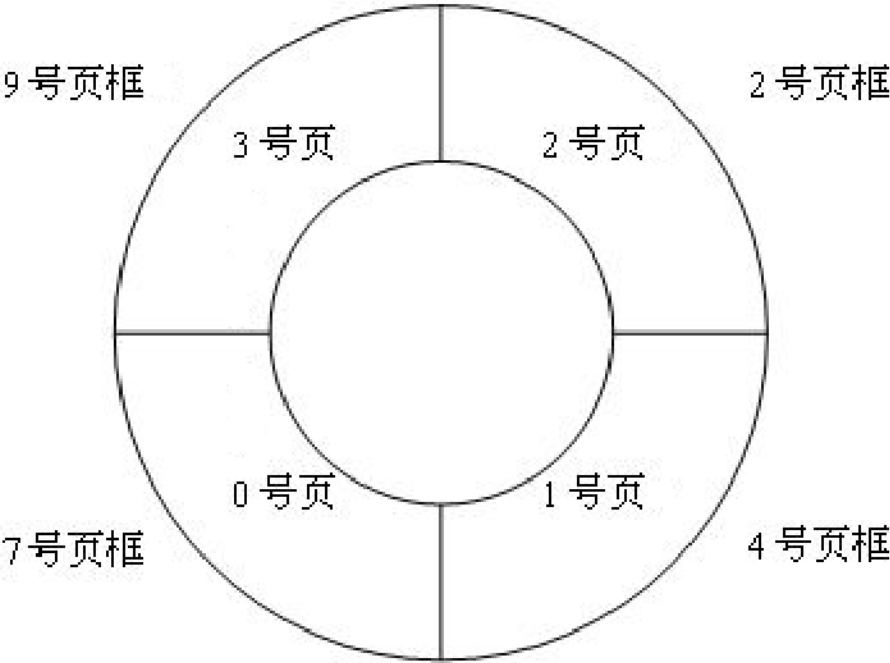
(3) 采用 FCFS (先来先服务) 调度策略更高效。因为 Flash 的半导体存储器的物理结构不需要考虑寻道时间和旋转延迟，可直接按 I/O 请求的先后顺序服务。

【46】设某计算机的逻辑地址空间和物理地址空间均为 64KB，按字节编址。某进程最多需要 6 页数据存储空间，页的大小为 1KB，操作系统采用固定分配局部置换策略为此进程分配 4 个页框。

页号	页框号	装入时间	访问位
0	7	130	1
1	4	230	1
2	2	200	1
3	9	160	1

当该进程执行到时刻 260 时，要访问逻辑地址为 17CAH 的数据。请回答下列问题：

- (1) 该逻辑地址对应的页号是多少？
(2) 若采用先进先出(FIFO)置换算法，该逻辑地址对应的物理地址？要求给出计算过程。
(3) 采用时钟(Clock)置换算法，该逻辑地址对应的物理地址是多少？要求给出计算过程。(设搜索下一页的指针按顺时针方向移动，且指向当前 2 号页框，示意图如下)



【参考答案】

(1) 因为 $17CAH = 0001\ 0111\ 1100\ 1010\ B$ ，由于页的大小为 1KB，则页内位移有 10 位，表示的页号的为左边 6 位，即 00101B，所以页号为 5。

(2) 根据 FIFO 算法，需要替换装入时间最早的页，故需要置换装入时间最早的 0 号页，即将 5 号页装入到 7 号页框中，即页框号为 111，凑成 6 位为 000111，与页内位移拼接得到

对应的物理地址为 0001 1111 1100 1010B = 1FCAH。

(3) 根据 CLOCK 算法, 如果当前指针所指页框的使用位为 0 时, 则替换该页; 否则将使用位清 0, 并将指针指向下一个页框, 继续查找。根据题设和示意图, 将从 2 号页框开始查找, 前 4 次查找页框号的顺序为 2-→4-→7-→9, 并将对应页框使用位清 0。在第 5 次查找中, 指针指向 2 号页框, 这时 2 号页框的使用位为 0, 故替换 2 号页框对应的 2 号页, 将 5 号页转入 2 号页框中, 并将对应使用位设置为 1, 所以对应的物理地址为 0000 1011 1100 1010 B=0BCAH。

【分析】45、46 题所考察的知识点: 外存储空间的管理方法 (位图表 bitmap), 磁盘调度算法 (CSCAN), 磁盘的结构 (平均旋转延时的计算), 逻辑地址到物理地址的映射 (物理地址计算), 页面置换算法 (FIFO 算法和 clock 算法)。这些知识点都包含到操作系统原理的本科教学大纲中, 要求考生必须掌握。但是在实际考试中, 考生这两题的得分少, 且得 0 分考生较多。在学习操作系统原理时, 考生需要充分理解和掌握操作系统的概念、原理和算法, 并且能够灵活应用。

11 年真题

选择题：

【23】下列选项中，满足短任务优先且不会发生饥饿现象的调度算法是（ ）。

- A. 先来先服务
- B. 高响应比优先
- C. 时间片轮转
- D. 非抢占式短任务优先

【解答】B。这里考察的是多种作业调度算法的特点。响应比=作业响应时间/作业执行时间=（作业执行时间+作业等待时间）/作业执行时间。高响应比算法，在等待时间相同情况下，作业执行的时间越短，响应比越高，满足短任务优先。同时响应比会随着等待时间增加而变大，优先级会提高，能够避免饥饿现象。

下面给出几种常见的进程调度算法特点的总结，读者要在理解的基础上识记。

	先来先服务	短作业优先	高响应比优先	时间片轮转	多级反馈队列
能否是可抢占	否	能	能	能	队列内算法不一定
能否是不可抢占	能	能	能	否	队列内算法不一定
优点	公平，实现简单	平均等待时间最少，效率最高	兼顾长短作业	兼顾长短作业	兼顾长短作业，有较好的响应时间，可行性强
缺点	不利于短作业	长作业会饥饿，估计时间不易确定	计算响应比的开销大	平均等待时间较长，上下文切换浪费时间	无
尤其适用于	无	作业调度，批处理系统	无	分时系统	相当通用
决策模式	非抢占	非抢占	非抢占	抢占	抢占

【24】下列选项中，在用户态执行的是（ ）。

- A. 命令解释程序
- B. 缺页处理程序
- C. 进程调度程序
- D. 时钟中断处理程序

【解析】A。CPU 状态分为管态和目态，管态又称特权状态、系统态或核心态。通常，操作系统在管态下运行，CPU 在管态下可以执行指令系统的全集。目态又称常态或用户态，机器处于目态时，程序只能执行非特权指令，用户程序只能在目态下运行。

CPU 将指令分为特权指令和非特权指令，对于那些危险的指令，只允许操作系统及其相关模块使用，普通的应用程序不能使用。

缺页处理与时钟中断都属于中断，会对系统造成影响，因此只能在核心态执行。进程调度属于系统的一部分，也只能在核心态执行。命令解释程序属于命令接口，是操作系统提供给用户所使用的接口，因此可以用在用户态执行。

补充：常见的特权指令有以下几种

- (1) 有关对 I/O 设备使用的指令。如启动 I/O 设备指令、测试 I/O 设备工作状态和控制 I/O 设备动作的指令等。
- (2) 有关访问程序状态的指令。如对程序状态字 (PSW) 的指令等。
- (3) 存取特殊寄存器指令。如存取中断寄存器、时钟寄存器等指令。
- (4) 其他指令

本题中 B、D 都是要修改中断寄存器，C 要修改程序状态字 (PSW)。

【25】在支持多线程的系统中，进程 P 创建的若干个线程不能共享的是 ()。

- A. 进程 P 的代码段
- B. 进程 P 中打开的文件
- C. 进程 P 的全局变量
- D. 进程 P 中某线程的栈指针

【解析】D。进程是资源分配的基本单元，进程下的各线程可以并行执行，它们共享进程的虚地址空间，但各个进程有自己的栈，各自的栈指针对其他线程是透明的，因此进程 P 中某线程的栈指针是不能共享的。

【26】用户程序发出磁盘 I/O 请求后，系统的正确处理流程是 ()。

- A. 用户程序→系统调用处理程序→中断处理程序→设备驱动程序
- B. 用户程序→系统调用处理程序→设备驱动程序→中断处理程序
- C. 用户程序→设备驱动程序→系统调用处理程序→中断处理程序
- D. 用户程序→设备驱动程序→中断处理程序→系统调用处理程序

【解析】B。首先用户程序 (目态) 是不能直接调用设备驱动程序的，有关对 I/O 设备使用的指令是特权指令，通过系统调用，把进程的状态从用户态变为核心态，故 C、D 错误。I/O 软件一般从上到下分为四个层次：用户层、与设备无关软件层、设备驱动程序及中断处理程序。与设备无关软件也就是系统调用的处理程序。因此正确处理流程为 B。

【27】某时刻进程的资源使用情况如下表所示：

进程	已分配资源			仍需分配			可用资源		
	R1	R2	R3	R1	R2	R3	R1	R2	R3
P1	2	0	0	0	0	1	0	2	1
P2	1	2	0	1	3	2			
P3	0	1	1	1	3	1			
P4	0	0	1	2	0	0			

此时的安全序列是 ()。

- A. P1, P2, P3, P4
- B. P1, P3, P2, P4
- C. P1, P4, P3, P2
- D. 不存在

【解析】D。使用银行家算法可知，不存在安全序列。由于初始 R1 资源没有剩余，只能分配资源给 P1 执行，P1 完成之后释放资源，这时由于 R2 只有 2 个剩余，因此只能分配对应资源给 P4 执行，P4 完成之后释放资源，但此时 R2 仍然只有 2 个剩余，无法满足 P2、P3 的要求，无法分配。因此产生死锁状态。

【28】在缺页处理过程中，操作系统执行的操作可能是（ ）。

I 修改页表 II 磁盘 I/O III 分配页框

A. 仅 I, II B. 仅 II C. 仅 III D. I, II 和 III

【解析】D。这些情况都可能发生，当产生缺页中断时，肯定会修改页表项并分配页框（分配页框出现在有空余页面的情况下），并且会从外存将所缺页调入，产生磁盘 I/O。

【29】当系统发生抖动（trashing）时，可以采取的有效措施是（ ）。

I. 撤销部分进程

II. 增加磁盘交换区的容量

III. 提高用户进程的优先级

A. 仅 I B. 仅 II C. 仅 III D. 仅 I, II

【解析】A。在具有对换功能的操作系统中，通常把外存分为文件区和对换区。前者用于存放文件，后者用于存放从内存中换出的进程。抖动现象是指刚刚被换出的内容又要被访问，因此马上又要换入这种系统频繁置换页面的现象。发生抖动时系统会将大部分时间用于处理页面置换上，降低系统效率。撤销部分进程可以减少系统页面数，可以有效防止系统抖动。改变优先级与增大交换区容量对减少抖动没有帮助。

【30】在虚拟内存管理中，地址变换机构将逻辑地址变为物理地址，形成该逻辑地址的阶段是（ ）。

A. 编辑

B. 编译

C. 链接

D. 装载

【解析】B。编译过程指编译程序将用户源代码编译成目标模块，在编译源代码的过程中，编译程序会将程序所使用的变量地址信息转化为逻辑地址。编辑过程是指编辑源代码的过程，此时还没有地址的概念。而链接和装载都是对编译好的程序进行处理，包括将逻辑地址转化为物理地址。

【31】某文件占 10 个磁盘块，现要把文件磁盘块逐个读入主存缓冲区，并送用户区进行分析，假设一个缓冲区与一个磁盘块大小相同，把一个磁盘块读入缓冲区的时间为 $100\mu\text{s}$ ，将缓冲区的数据传送到用户区的时间是 $50\mu\text{s}$ ，CPU 对一块数据进行分析的时间为 $50\mu\text{s}$ 。在单缓冲区和双缓冲区结构下，读入并分析完该文件的时间分别是（ ）。

A. $1500\mu\text{s}$, $1000\mu\text{s}$

B. $1550\mu\text{s}$, $1100\mu\text{s}$

C. $1550\mu\text{s}$, $1550\mu\text{s}$

D. $2000\mu\text{s}$, $2000\mu\text{s}$

【解析】B。单缓冲区当上一个磁盘块从缓冲区读入用户区完成时下一磁盘块才能开始读入，也就是当最后一块磁盘块读入用户区完毕时所用时间为 $150 \times 10 = 1500$ ，加上处理最后一个磁盘块的时间 50，得 1550。双缓冲区情况下，不存在等待磁盘块从缓冲区读入用户区的问题，因此传输数据全部传输到缓冲区的时间为 $100 \times 10 = 1000$ ，再加上将双缓冲区的数据传输到用户区并处理完的时间 $2 \times 50 = 100$ ，得 1100。

【32】有两个并发执行的进程 P1 和 P2，共享初值为 1 的变量 x。P1 对 x 加 1，P2 对 x 减 1。加 1 和减 1 操作的指令序列分别如下所示。

//加 1 操作

load R1, x ①//取 x 到寄存器 R1 中

inc R1 ②

store x, R1 ③//将 R1 的内容存入 x

//减 1 操作

load R2, x ④

dec R2 ⑤

store x, R2 ⑥

两个操作完成后，x 的值（ ）。

- A. 可能为-1 或 3 B. 只能为 1
C. 可能为 0、1 或 2 D. 可能为-1、0、1 或 2

【解答】C。执行①②③④⑤⑥结果为 1，执行①②④⑤⑥③结果为 2，执行④⑤①②③⑥结果为 0，结果-1 无法得到。

解答题：

45.某银行提供 1 个服务窗口和 10 个供顾客等待的作为。顾客到达银行时，若有空座位，则到取号机上领取一个号，等待叫号。取号机每次仅允许一位顾客使用。当营业员空闲时，通过叫号选取一位顾客，并为其服务。顾客和营业员的活动过程描述如下；

```
cobegin
{
    Process 顾客;
    {
        从取号机获取一个号码;
        等待叫号;
        获取服务;
    }
    Process 营业员
    {
        While (TRUE)
        {
            叫号;
            为顾客服务;
        }
    }
}coend
```

请添加必要的信号量和 P、V（或 wait()、signal()）操作，实现上述过程中的互斥与同步。要求写出完整的过程，说明信号量的含义并赋初值。

【解析】

Semaphore seats = 10; //表示空余座位数量的资源信号量，初值为 10

Semaphore mutex = 1; //管理取号机的互斥信号量，初值为 1，表示取号机空闲。

Semaphore custom = 0; //表示顾客数量的资源信号量，初值为 0

```
Process 顾客
{
    P(seats);           //找个空座位先
    P(mutex);           //再看看取号机是否空闲
    从取号机上取号;
    V(mutex);           //放开那个取号机！
    V(custom);           //取到号，告诉营业员有顾客
    等待叫号;
    V(seats);           //被叫号，离开座位
    接受服务;
```

```

}

Process 营业员
{
    While(true)
    {
        P(custom);    //看看有没有等待的顾客
        叫号;
        为顾客服务;
    }
}

```

46.某文件系统为一级目录结构，文件的数据一次性写入磁盘，已写入的文件不可修改，但可多次创建新文件。请回答如下问题。

（1）在连续、链式、索引三种文件的数据块组织方式中，哪种更合适？要求说明理由。为定位文件数据块，需要 FCB 中设计哪些相关描述字段？

（2）为快速找到文件，对于 FCB，是集中存储好，还是与对应的文件数据块连续存储好？要求说明理由。

【解析】

（1）连续更合适。因为一次写入不存在插入问题，而且写入文件之后不需要修改，连续的数据块组织方式很适合一次性写入磁盘不再修改的情况。同时连续存储相对于链式和索引省去了指针的空间开销，支持随机查找，查找速度最快。

（2）FCB 集中存储较好。FCB 存储有文件的很多重要信息，同时是文件目录的重要组成部分，在检索时，通常会访问对应文件的 FCB。如果将 FCB 集中储存，则可以减少在检索过程中产生的访盘次数，提高检索的速度。