

2011 计算机 考研试题及参考答案

1、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_____。C

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

解析：本题是对典型进程调度算法的考察，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，作业执行时间越少，响应比越高，优先执行，满足短任务优先。随着等待时间增加，响应比也会变大，执行机会就增大，所以不会产生饥饿现象。先来先服务和时间片轮转不符合短任务优先，非抢占式短任务优先会产生饥饿现象。

以下哪一些是基于时间片的调度算法_____。AB

A. 时间片轮转 B. 多级反馈队列调度算法
C. 抢占式调度算法 D. 先来先服务调度算法

解析：本题考察进程调度算法中的时间片调度算法。其中的时间片轮转法以及多级反馈队列调度算法是基于时间片的调度算法。至于其他的算法均不是基于时间片的调度算法。

2、下列选项中，在用户态执行的是_____。A

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：本题涉及的考点是OS的概念、特征、功能和提供的服务，具体考查的是处理机的状态，以及在不同的状态下执行的程序。缺页处理程序和时钟中断都属于中断，在核心态执行进程调度属于系统调用在核心态执行，命令解释程序属于命令接口，它在用户态执行。

在一般OS中必不可少的调度是_____。D

A. 高级调度 B. 中级调度 C. 作业调度 D. 进程调度

解析：高级调度也就是作业调度，用于决定把外存上处于后备队列中的哪些作业调入内存并为它们创建进程、分配必要的资源，然后，再将新创建的进程排在就绪队列上，准备执行在批处理系统中，需要有高级调度，但在分时系统和实时系统中通常不需要作业调度。所以不是必不可少的调度。中级调度它按照一定的算法将外存中已具备运行条件的进程换入内存，将内存中处于阻塞状态的某些进程换出到外存。中级调度的目的是为了解决内存紧张问题，它常用于分时系统及具有虚拟存储器的系统中，也不是必不可少的调度。低级调度也称进程调度，用来决定就绪队列中哪个进程应先获得处理机，并将处理机分配给选中的进程。进程调度是最基本的调度，一般的OS中都必须配置它。

3、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_____。D

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：本题考查的是多线程模型中的特点，进程中某线程的栈指针，对其他线程透明，不能与其他线程共享。

线程是进程中某个单一顺序的控制流，也被称为轻量进程，它是进程中的一个实体，是被系统独立调度和分派的基本单位。

线程的属性：

（1）轻型实体。线程除了拥有运行中必不可少的资源（如线程控制块TCB、程序计数器、寄存器组、堆栈等）外基本上不拥有系统资源。

- (2) 独立调度和分派的基本单位。
- (3) 可并发执行。
- (4) 共享进程资源。

多线程模型包括多对一模型，即多个用户级线程映射到一个内核级线程；一对一模型将每个用户级线程映射到一个内核级线程；多对多模型将 n 个用户级线程映射到 m 个内核级线程上（要求 $m \leq n$ ）

进程与线程可以从四个方面来考查区别：

- (1) 调度方面：线程是调度和分派的基本单位；
- (2) 并发性方面：进程之间可以并发执行，一个进程中的若干线程也可以并发执行；
- (3) 拥有资源方面：进程作为拥有资源的基本单位，线程除少量必不可少的资源外，基本上不拥有资源，但它可以访问其隶属进程的资源；
- (4) 系统开销：进程间切换时，要涉及进程环境的切换，开销比较大。而线程间切换只需保存和设置少量的寄存器内容，因此进程切换的系统开销远大于线程间切换的开销。

4、用户程序发出磁盘 I/O 请求后，系统的正确处理流程是_____。B

- 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：本题考核 IO 控制方式，要求考生理解 OS 处理 IO 请求的流程。IO 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、设备驱动程序以及中断处理程序。与设备无关软件层也就是系统调用的处理程序。

IO 控制方式包括有程序 IO 方式、中断驱动 IO 控制方式、直接存储器访问 IO 控制方式和 IO 通道控制方式。需要理解记忆这些内容。

IO 控制方式有四种：程序 IO 控制方式、中断控制方式、DMA 方式和通道控制方式。它们各自的优缺点：

(1) 程序 IO 控制方式。优点是**控制简单**，也不需要很多硬件支持。缺点是**CPU 和外设之间只能串行工作**，且 CPU 大部分时间处于循环测试状态，这使得 CPU 的利用率大大降低，CPU 在一段时间内只能和一台外设交换数据信息，从而不能实现设备之间的并行工作：由于程序 IO 方式靠测试设备状态标志来控制数据传送，因此无法发现和处理因设备或其他硬件所产生的错误。所以程序 IO 控制方式只适用于那些 CPU 执行速度较慢且外设较少的系统。

(2) 中断控制方式。优点是能实现 CPU 与设备、设备与设备之间的**并行操作**，**CPU 的利用率较程序 IO 控制方式大大提高**。缺点是**IO 控制器的数据缓冲寄存器通常较小**，且数据缓冲寄存器装满数据后将会发出中断，因此一次数据传送过程中中断次数较多，耗去了大量 CPU 时间；如果系统中配置的外设数目较多，且都以中断方式进行控制，则将耗去大量 CPU 时间或因 CPU 来不及处理而造成数据丢失。

(3) DMA 方式。与中断方式相比，DMA 方式的优点是在**一批数据传送完成后中断 CPU**，从而大大减少了 CPU 进行中断处理的次数，并且 DMA 方式下的数据传送是在 DMA 控制器控制下完成的，在**数据传输过程中无需 CPU 的干预**，缺点是 DMA 方式仍有一定的局限，如对外设的管理和某些操作仍由 CPU 控制，且多个 DMA 控制器的使用也不经济。

(4) 通道控制方式。通道是一个专管 IO 工作的处理机。优点：在通道控制方式下，CPU 只需发出 IO 指令，通道就能完成相应的 IO 操作，并在 IO 操作结束时向 CPU 发出中断信号。由此可见，CPU 仅在 IO 操作开始和结束时花极短的时间处理与 IO 操作有关的事宜，其余时间都与通道并行工作，此外一个通道还能控制多台外设。缺点是通道价格较高，从经济角度出发不宜过多使用。

5、某时刻进程的资源使用情况如下表所示：此时的安全序列是____。D

进 程	已分配资源	尚需资源	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	

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

解析：在避免死锁的方法中，所施加的限制条件较弱，有可能获得令人满意的系统性能。在该方法中把系统的状态分为安全状态和不安全状态，只要能使系统始终都处于安全状态便可以避免发生死锁。银行家算法的基本思想是分配资源前，判断系统是否是安全的，若是，才分配。它是最具有代表性的避免死锁的算法。

设进程 P 提出请求 REQUEST[i]，则银行家算法按如下规则进行判断。

- (1) 如果 $REQUEST[P][i] \leq NEED[P][i]$ ，则转 (2)；否则，出错。
- (2) 如果 $REQUEST[P][i] \leq AVAILABLE[P][i]$ ，则转 (3)；否则，出错。
- (3) 系统试探分配资源，修改相关数据：

$AVAILABLE[i] = REQUEST[P][i];$

$Allocation[P][i] += REQUEST[P][i];$

$NEED[P][i] = REQUEST[P][i];$

(4) 系统执行安全性检查，如安全，则分配成立；否则试探性分配作废，系统恢复原状，进程等待。

安全线检查算法：

- (1) 设置 2 个工作向量 $work = AVAILABLE; FINISH$
- (2) 从进程集合中找到一个满足下述条件的进程， $FINISH == false; NEED \leq Work$; 如找到，执行 (3)；否则，执行 (4)。
- (3) 设进程获得资源，可顺利执行，直至完成，从而释放资源。

$Work += ALLOCATION; Finish = true; GOTO(2)$ 。

- (4) 如所有的进程 $Finish = true$, 则表示安全；否则系统不安全。

死锁的预防是指破坏死锁产生的 4 个必要条件之一，死锁的避免使用银行家算法，死锁的解决有 2 种方法：资源剥夺法和撤消进程法。

6、在缺页处理过程中，操作系统执行的操作可能是____。D

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：本题涉及虚拟内存中的请求分页存储管理方式，具体考查的是 OS 在缺页处理过程中的操作。缺页中断调入新页面，肯定要修改页表项和分配页框，所以 I、III 可能发生，同时内存没有页面，需要从外存读入，会发生磁盘 IO。

7、当系统发生抖动(thrashing)时，可采取的有效措施是____。A?

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：本题是对虚拟内存管理中抖动现象的考查。在具有对换功能的 OS 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换出的进程。抖动现象是指刚刚被换出的页很快又要被访问，为此，又要换出其他页，而之后该页又很快被访问，如此频繁的置换页面，以至于大部分时间都花在页面置换上。撤销部分进程可以减少所要用的页面数，防止抖动。交换区大小和进程优先级都与抖动无关。

8、在虚拟内存管理中，地址变换机构将逻辑地址变换为物理地址，形成该逻辑地址的阶段是____。B（好像教材里没有直接提到）

A. 编辑 B. 编译 C. 链接 D. 装载

解析：本题是对虚拟内存的基本概念的考查，编译过程指编译程序将用户源代码编译成目标模块。源地址编译成目标程序时，会形成逻辑地址。

9、某文件占 10 个磁盘块，现要把该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 $100\mu s$ ，将缓冲区的数据传送到用户区的时间是 $50\mu s$ ，CPU 对一块数据进行分析的时间是 $50\mu s$ 。在单缓冲区及双缓冲区结构下，读入并分析完该文件的时间分别是____。B

A. $1500\mu s$ ， $1000\mu s$ B. $1550\mu s$ ， $1100\mu s$ C. $1550\mu s$ ， $1550\mu s$ D. $2000\mu s$ ， $2000\mu s$

解析：本题考的是高速缓冲区和缓冲区。单缓冲区下，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，结果为 1550。双缓冲区下，不存在等待磁盘块从缓冲区读入用户区的问题，也就是 $100 \times 10 + 100 = 1100$ 。

高速缓存是可以保存数据拷贝的高速存储器。访问高速缓存要比访问原始数据更为高效，速度更快。

高速缓存和缓冲区都是介于一个高速设备和一个低速设备间的，但是它们之间有着很大的区别：

（1）两者存放的数据不同。高速缓存上放的是低速设备上的某些数据的拷贝，也就是说高速缓存上有的数据低速设备上必然有；而缓冲区则是放置低速设备传递给高速设备的数据，这些数据从低速设备传递到缓冲区中，而在低速设备中却不一定有备份。

（2）两者的目的不同。高速缓存是为了存放低速设备上经常要被访问到的数据的拷贝，这样一来，高速设备就不需要每次都访问低速设备，但如果要访问的数据不在高速缓存中，那么高速设备还是需要访问低速设备；而缓冲区是为了缓和高速设备和低速设备间速度不匹配的矛盾而存在的。

10、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。C（就是书上的，4、5、6 三种结果）

//加 1 操作 //减 1 操作

load R1, x //取 x 到寄存器 R1 中 Load R2, x

inc R1 dec R2

store x, R1 //将 R1 的内容存入 x store x, R2

两个操作完成后，x 的值____。

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

解析：本题考查的是用软件方式实现临界区之间的互斥，将 P1 中 3 条语句编号为 1,2,3，P2 中 3 条语句编号为 4,5,6。则依次执行 1,2,3,4,5,6 得结果 1，依次执行 1,2,4,5,6,3 得结果 2，执行 4,5,1,2,3,6 得结果 0。结果 -1 不可能得出。

实现临界区之间的互斥有多种方式，考生需要掌握的是用软件方法实现临界区之间的互斥

和用信号量机制实现临界区之间的互斥。其中信号量机制实现临界区之间的互斥是 OS 中的重难点。

11、某银行提供 1 个服务窗口和 10 个供顾客等待的座位。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。顾客及营业员的活动描述如下：

```
cobegin
{
    process 顾客 i
    {
        从取号机获取一个号码；
        等待叫号；
        获得服务；
    }
    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, //有 10 个座位的资源信号量

Mutex=1, //取号机互斥信号量

haveCustomer=0; //顾客与营业员同步，无顾客时营业员休息

```
cobegin
{
    process 顾客 i
    {
        P(seats); //等空位
        P(mutex); //申请使用取号机
        从取号机获取一个号码；
        V(mutex); //取号完毕
        V(haveCustomer); //通知营业员有新顾客到来
        等待叫号；
        V(seats); //离开座位
        获得服务；
    }
    process 营业员
```



```

{
    while (TRUE)
    {
        P(haveCustomer); //没有顾客则休息
        叫号；
        为顾客服务；
    }
}
}coend

```

一个无红绿灯的十字路口 E1 可以从四个方向来车，每个方向的车在通过这个十字路口前都会先在自己方向的停车线上停一下，然后按从每个方向的车到达停车线的先后严格顺序通过十字路 E1，用 PV 操作实现这一机制，有以下要求：

- (1) 说明你的设计思想。
- (2) 信号量及其他变量的定义和初值。
- (3) 写出程序。
- (4) 证明不会出现死锁。

解析：

(1) 将十字路口的中心看成是临界区，每次只能接收一辆车进入这个区域。到达停车线的车用 P 操作申请进入临界区，如果临界区有车则等待一个 V 操作来唤醒。离开临界区的车用 V 操作来释放对临界区的拥有权，同时通知最早进行等待的车进入临界区。

(2) 说明： $m=1, a=1, b=1, c=1, d=1$ ， m 表示临界区信号量， $a \sim d$ 表示各个方向是否有车在等待。 $Pro(i) (i=a \sim d)$ 表示某个方向的操作。

(3) 程序：

```

While(true){
Pro(a);
Pro(b);
Pro(c);
Pro(d);
}
Pro(i){
P(i);
P(m);
Pass;
V(m);
V(i);
}

```

(4) 因为 $a \sim d$ 之间并没有联系，因此不会出现交叉等待的情况。即不满足死锁出现的其中一个必要条件——循环等待，故不会出现死锁。

12、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，需在 FCB 中设计哪些相关字段？

(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，连续的数据块组织方式完全可以满足一次性写入磁盘。同时连续文件组织方式减少了其他不必要的空间开销，而连续的组织方式顺序查找读取速度是最快的。

(2) FCB 集中存储好。目录是存在磁盘上的，所以检索目录的时候需要访问磁盘，速度很慢；集中存储是将文件控制块的一部分数据分解出去，存在另一个数据结构中，而在目录中仅留下文件的基本信息和指向该数据结构的指针，这样一来就有效地缩短减少了目录的体积，减少了目录在磁盘中的块数，于是检索目录时读取磁盘的次数也减少，于是就加快了检索目录的次数。

存在一个基于 FAT 的文件系统，表项为 16 位，一个用户想要安装一块包含 1310752 个扇区（每个扇区的大小为 512 字节）的磁盘。

(1) 可能存在什么样的问题？

(2) 在保证文件系统的兼容性的前提下，如何解决该问题？你的解决方案对于磁盘空间的利用率有什么影响？

解析：

(1) 2^{16} 小于 1310752，使得有些扇区无法被访问到。

(2) 采用多级索引的方法可以解决该问题。这种方案的不足之处会使得磁盘空间的利用率下降。