# 多道程序设计技术：

1. 现代操作系统采用多道程序设计技术（相较于以前的操作系统来说）；
2. 最基本的两个特征：并发性和共享性；
3. 现代操作系统通常采用 字符流式 的文件逻辑结构；

# 进程：

1. 进程实体= PCB（Process Control Block）进程控制块 + 程序段 + 数据段；
2. 进程之间的直接制约主要源于：进程合作；进程之间的间接制约源于：互斥共享资源；
3. 进程的三个基本状态：就绪态、运行态和阻塞态；
4. 若有m个进程共享一个互斥段且每次最多允许m-1个进程进入互斥段，那么信号量变化的范围是: [1 – m, m]；
5. 进程间的高级通信方式：共享存储器系统、消息传递系统和管道通信系统；

# 进程调度算法：

1. 有抢占式和非抢占式两种方式；
2. 抢占式的主要原则有：时间片原则、短作业优先原则、最高优先权优先原则；
3. 非抢占式的主要原则有：FIFO原则（First In First Out）；
4. 使所有交互型作业均能获得较好的响应时间：时间片轮转调度算法；（时间片大小和作业数量的多少哪个影响响应时间的程度深：当然是时间片的大小。因为时间片大小影响的是O(N）级别，但作业数量的多少影响的是O(1)级别；
5. 能满足短作业优先且不会产生饥饿的现象：高响应比优先算法；

# 进程间通信方式：

1. 共享存储区的通信效率最高；
2. 共享数据结构的通信效率最低；
3. 管道是利用共享的文件来完成进程间的通信；

# 信号量机制：

1. 采用信号量实现临界资源的互斥共享，当S.value>0时，信号量的值表示可用的临界资源数；当S.value<0时，其绝对值表示：等待该资源二阻塞的进程数；
2. 整型信号量违背了同步机制的：让权等待原则；

# 死锁：

1、成为死锁的四个必要条件：

1）互斥条件：一个资源每次只能被一个进程使用；

2）请求与保持条件：一个进程因请求资源而阻塞时，对已获得的资源保持不放；

3）不剥夺条件：进程已获得的资源，在未使用完之前，不能强行剥夺；

4）循环/环路 等待条件：若干进程之间形成一种头尾相接的循环等待资源关系；

如何破坏这四个条件：

1）Spooling技术原则；（Spooling技术利用磁盘作为后援存储器，在联机情况下实现的同时外围操作，实现设备的虚拟分配；系统中采用Spooling技术的目的是：提高独占设备的利用率）；

2）一次性地为进程分配所有资源原则；

3）进程获得了部分资源的情况下，如若得不到其他资源则释放已占有的全部资源；

4）将所有资源编号，并要求进程严格按照资源号递增的次序申请资源；（按序分配资源）

2、

# 请求分页存储管理方式：

1. 在该方式中，地址变换过程中可能会因为：地址越界、缺页和访问权限非法等原因产生中断；

# 文件系统：

1. 在引入索引结点的文件系统中，目录项中只需包含：文件名和磁盘索引结点编号；
2. FAT = FILE ALLOCATION TABLE，文件分配表：既可以用来管理空闲盘块，又可以用它来指出分配给文件的下一个盘块的物理地址；
3. 位示图表：其中的每一个二进制位可以用来表示磁盘中的一个块的使用情况；（位示图法可以用于：磁盘空闲空间的管理）；
4. 文件系统的功能：1、对文件的按名存取；2、负责实现的访问数据时的逻辑结构到物理结构的转换；3、实现对文件的各种操作；
5. 文件的外存分配方式：连续分配、链接分配和索引分配；
6. 对记录式文件，操作系统为用户存取文件信息的最小单位是：数据项；
7. Linux中一个文件的权限是：<https://www.cnblogs.com/devgis/p/16382356.html>
8. Liunx文件系统中的文件类型：目录文件、设备文件和管道文件；

# 磁盘：

1. 按信息交换的单位来分，磁盘属于块设备，磁盘的I/O通常采用DMA（Direct Memory Access）的控制方式；
2. 访问磁盘的时间包括：寻道时间、读写时间和旋转延迟时间；

# I/O设备：

1、设备驱动程序应完成哪些功能？

设备驱动程序是I/O进程与设备控制器之间的通信程序。

设备驱动程序的主要功能如下:

(1)将接收的抽象要求转换为具体要求；

(2)检查用户I/O请求的合法性,了解I/O设备的状态,传递有关参数,设置设备的工作方式；

(3)发出I/O命令,启动对应的I/O设备,完成指定的I/O操作；

(4)及时响应由控制器或通道发来的中断请求,根据中断类型调用相应的中断处理程序进行处理；

(5)对于设置有通道的计算机系统,驱动程序还应能够根据用户的I/O请求,自动的构成通道程序。

2、为了实现设备的独立性，应用程序必须通过 逻辑设备名 来请求使用某类设备；

3、磁盘、智能卡是面向块的设备；打印机是面向字符（流）的设备；

4、流式文件就是无结构的文件；

5、从使用的角度来看，可以把设备分为：物理设备和逻辑设备；

6、从信息交换单位来看，可以把设备分为：字符设备和块设备；

7、用户在使用I/O设备的时候，通常采用：逻辑设备名；

8、打印机是面向字节的设备，是字符设备,传送字节流,且是中速设备,传输速率为数千个~数万个字节/秒。

1）按设备的共享属性分类,可把设备分成独占设备、共享设备和虚拟设备。

2）按信息交换的单位分类,可把设备分成字符设备和块设备。

3）按传输速率分类,可把设备分成低速设备、中速设备、高速设备。

9、

# 请求分页系统：

1. 在请求分页系统中，页表应包括哪些数据项？每项的作用是什么？

Ps：基本分页系统的页表中包含了页号和物理块号（页框号）两个数据项；

页表应包括：

1、页号；

2、物理块号；

3、状态位P：指示该页是否调入内存；

4、访问字段A：用于记录本页在一段时间内被访问的次数，或最近已有多长时间未被访问，提供给置换算法选择换出页面时参考；

5、修改位M：表示该页在调入内存后是否被修改过；

6、外存地址：用于指出该页在外存上的地址，通常是物理块号，供调入该页时使用。

# 临界区：

1. 临界区是指一个访问共用资源的程序片段；

# 内存压缩（紧凑）技术：

1. 把所有作业移动到内存的一端，从而在内存的另一端形成一个大的空闲分区的技术。

# 重定位技术：

1、静态重定位：在程序装入内存的过程中完成。是指在程序开始运行前，程序中的各个地址有关的项均已完成重定位，地址变换通常是在装入时一次完成的，以后不再改变，

2、动态重定位：动态重定位即在程序运行过程中要访问数据时再进行逻辑地址与物理地址的变换（即在逐条指令执行时完成地址映射），即：目标程序可以不经过任何改动就装入物理内存；

3、程序的装入包括：绝对装入方式、静态重定位（可重定位装入方式）和动态重定位（动态运行时装入方式）；

4、

# 磁盘调度算法：

1. 只有N步SCAN可以避免磁臂粘着的现象发生；

# 页式、段式、段页式、动态分区方式：

1. 会产生外部碎片的是：段式、动态分区方式；
2. 会产生内部碎片的是：页式、段页式；

# 操作系统：

1. 操作系统是控制应用程序的执行，并充当应用程序和计算机硬件之间的接口；
2. 操作系统的有效性包含：提高系统资源的利用率和提高系统的吞吐量；
3. 操作系统与硬件有关与应用无关；

# 内存分配策略：

1. 最佳适应算法是按 空闲区大小递增的顺序 形成空闲分区链；
2. 内存分配策略需要考虑：最小物理块数的确定、物理块的分配策略和物理块的分配算法；

# 虚拟存储管理系统：

1. 基础是程序的局部性原理；

# 特权指令：

1. 特权指令指具有特殊权限的指令，这类指令只用于操作系统或其他系统软件。

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（外设）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。

Eg：启动外设指令、关中断指令和关机指令；

系统调用指令不是特权指令。

1. 访管指令是一类机器指令，执行访管指令可以引起访管中断。访管指令不是特权指令，它可在用户态下运行，也可在管态下运行。
2. 系统调用不是机器指令，每个系统调用命令相当于一个函数；在系统调用命令中，总是包含一条访管指令，当系统调用执行到访管指令时，就引起访管中断，在进入中断处理程序后便由算态转入管态。在管态下，可以执行特权指令完成操作系统提供的功能。当中断处理程序结束后又从管态返回算态。当用户程序想要操作系统提供服务时，就可以在用户程序中使用系统调用命令，它是操作系统与用户的编程的接口。

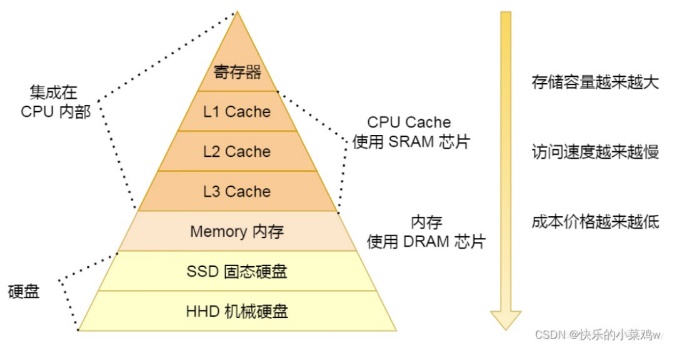
# 缓冲：

1. 缓冲能缓和CPU与I/O设备间速度不匹配的矛盾；
2. 缓冲能减少I/O操作对CPU的中断频率；
3. 缓冲能协调数据处理单位和传输单位不匹配的问题；

# PS：

1. 请求分页存储管理方式和请求调页存储管理方式是一个东西；
2. 并发和并行的区别：
3. 并发：单处理器：A1、B1、A2、B2、A3、C3等交替运行；
4. 并行：双处理器：A、B、C、D、E、F等；

3、在多级存储体系中速度最快的是：CPU寄存器。

4、

5、简述多级反馈队列调度算法，并说明为什么能较好地满足各方面用户的需要？

多级反馈队列调度算法是一种根据先来先服务原则给就绪队列排序，为就绪队列赋予不同的优先级数，不同的时间片，按照优先级抢占CPU的调度算法。算法的实施过程如下：

按照先来先服务原则排序，设置N个就绪队列为Q1，Q2…QN，每个队列中都可以放很多作业；

为这N个就绪队列赋予不同的优先级，第一个队列的优先级最高，第二个队列次之，其余各队列的优先权逐个降低；

设置每个就绪队列的时间片，优先权越高，算法赋予队列的时间片越小。时间片大小的设定按照实际作业（进程）的需要调整；

进程在进入待调度的队列等待时，首先进入优先级最高的Q1等待。

首先调度优先级高的队列中的进程。若高优先级中队列中已没有调度的进程，则调度次优先级队列中的进程。例如：Q1,Q2,Q3三个队列，只有在Q1中没有进程等待时才去调度Q2，同理，只有Q1,Q2都为空时才会去调度Q3。

对于同一个队列中的各个进程，按照时间片轮转法调度。比如Q1队列的时间片为N，那么Q1中的作业在经历了时间片为N的时间后，若还没有完成，则进入Q2队列等待，若Q2的时间片用完后作业还不能完成，一直进入下一级队列，直至完成。

在低优先级的队列中的进程在运行时，又有新到达的作业，那么在运行完这个时间片后，CPU马上分配给新到达的作业即抢占式调度CPU。

为什么能较好地满足各方面用户的需要？

对终端型作业用户而言,由于他们所提交的作业大多属于交互型,作业通常比较短小,系统只要能使这些作业在优先级最高的第1个队列所规定的时间片内完成,就可使终端型作业用户感到满意。对于短批处理作业用户而言,他们的作业开始时像终端型作业一样,如果仅在第1个队列中执行一个时间片即可完成,就可获得与终端型作业一样的响应时间;对于稍长的作业,通常也只需要在第2个队列和第3个队列中各执行一个时间片即可完成,其周转时间仍然较短;对于长批处理作业用户而言,他们的长作业将依次在第1,2,...,n个队列中运行,而随着队列的逐级改变则运行时间片成倍增加,这样也使得长批处理作业获得较多的运行时间。因此多级反馈队列调度算法能较好地满足各类用户的需求。

1. Liunx进程状态：<https://blog.csdn.net/qq_51659538/article/details/126786597>
2. Linux索引结点：inode就是索引节点，它用来存放档案及目录的基本信息，包含时间、档名、使用者及群组等。

每一个索引节点都是一个表项，包含有关文件的信息(元数据)：

1、文件类型，权限，UID，GID

2、链接数（指向这个文件名路径名称个数）

3、该文件的大小和不同的时间戳

4、指向磁盘上文件的数据块指针

5、有关文件的其他数据

1. 联想存储器就是快表：Translation Lookaside Buffer，TLB
2. 某虚拟存储器的用户空间共有32个页面,每页1KB,主存16KB。假定某时刻系统为用户的第0、1、2、3页分配的物理块号为5、10、4、7,而该用户作业的长度为6页,试将十六进制的虚拟地址0A5C、103C、1A5C转换成物理地址。

由题目所给出条件可知,该系统的逻辑地址有15位,其中高5位为页号,低10位为页内地址;物理地址有14位,其中高4位为块号,低10位为块内地址。另外,由于题目中给出的逻辑地址是16进制数,故可先将其转换成二进制数以直接获得页号和页内地址,再完成地址的转换。

(1)逻辑地址(0A5C)16的页号为(00010)2,即2,故页号合法;从页表中找到对应的内存块号为4,即(0100)2;与页内地址(10 0101 1100)2拼接形成物理地址(01 0010 0101 1100)2,即(125C)16。

(2)逻辑地址(103C)16的页号为4,页号合法,但该页未装入内存,故产生缺页中断。

(3)逻辑地址(1A5C)16的页号为6,为非法页号,故产生越界中断。

Ps：32个页面 每页1KB 那就是32KB 这是对于用户空间来说的（读题）

主存就是物理地址吧（这个应该好理解，就是实际能存多少）

1. linux中断分为：上半部（顶半部）与下半部（底半部）

上半部：上半部就是中断处理函数，那些处理过程比较快，不会占用很长时间的处理就可以放在上半部完成。

下半部：如果中断处理过程比较耗时，那么就将这些比较耗时的代码提出来，交给下半部去执行，这样中断处理函数就会快进快出。

目的：

Linux 内核将中断分为上半部和下半部的主要目的就是实现中断处理函数的快进快出，那些对时间敏感、执行速度快的操作可以放到中断处理函数中，也就是上半部。剩下的所有工作都可以放到下半部去执行，比如在上半部将数据拷贝到内存中，关于数据的具体处理就可以放到下半部去执行。如何区分上/下半部？根据实际情况：

参考点：

①、如果要处理的内容不希望被其他中断打断，那么可以放到上半部

②、如果要处理的任务对时间敏感，可以放到上半部。

③、如果要处理的任务与硬件有关，可以放到上半部

④、除了上述三点以外的其他任务，优先考虑放到下半部。

上半部处理很简单，直接编写中断服务函数即可。

Linux内核提供了多种下半部机制

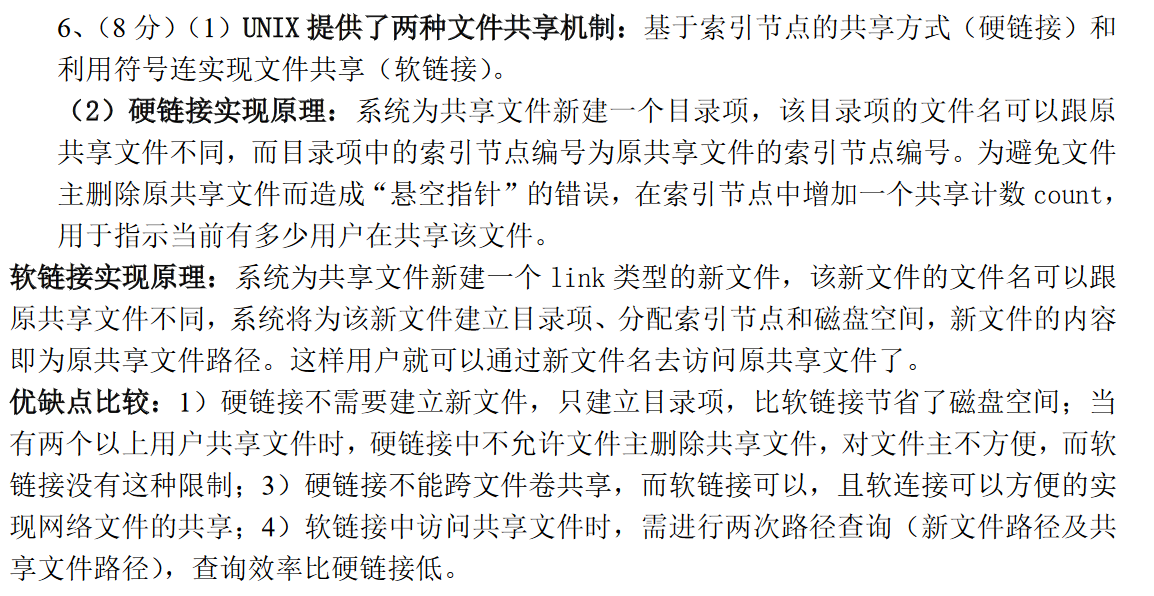
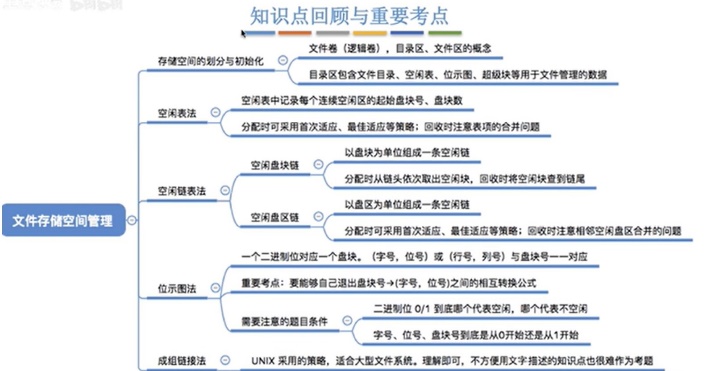
软中断

Tasklet

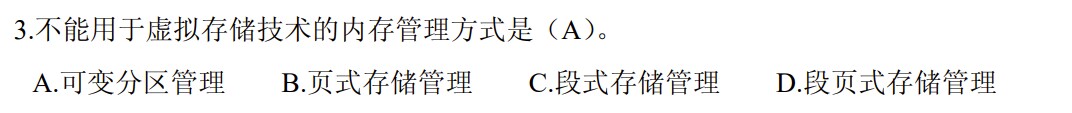
工作队列

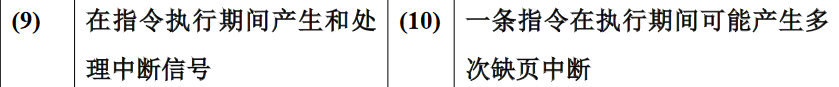
1. 在分时系统中，时间片一定，用户数越多，响应时间越长。分时系统的响应时间T可以表达为：T≈Q×N，其中Q是时间片，Q是用户数。当时间片一定，用户数越多(即N越大)，T就越大。
2. 进程是系统进行资源分配的基本单位；线程是操作系统能进行运算调度的最小单位；
3. 字节多路通道:这是一种按字节交叉方式工作的通道。它通常都含有许多非分配型子通道,其数量可从几十到数百个,每个子通道连接一台I/O设备,并控制该设备的I/O操作。这些子通道按时间片轮转方式共享主通道。只要字节多路通道扫描每个子通道的速率足够快,而连接到子通道上的设备的速率不是太高时,便不致丢失信息。
4. 数组选择通道:字节多路通道不适于连接高速设备,这推动了按数组方式进行数据传送的数组选择通道的形成。
5. 数组多路通道:数组选择通道虽有很高的传输速率,但它却每次只允许一个设备数据。数组多路通道是将数组选择通道传输速率高和字节多路通道能使各子通道(设备)分时并行操作的优点相结合而形成的一种新通道。它含有多个非分配型子通道,因而这种通道既具有很多高的数据传输速率,又能获得令人满意的通道利用率。
6. 信箱通信是一种“间接”通信方式。信箱通信是借助于收发双方进程之外的共享数据结构作为通信中转，发送方和接受方不必直接建立联系，没有处理时间上的限制；发送方可以在任何时间发送信息，接收方也可以在任何时间收信。
7. 文件的类型和权限：

<https://blog.csdn.net/weixin_42324404/article/details/85019916>

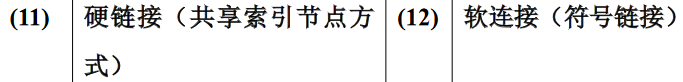
1. 
2. 存的都是没用过的空间
3. 分层式OS结构是吧OS的功能模块划分为若干层，每层之间的模块只能单向调用。（分层就是要求高内聚低耦合，这也是模块化软件设计的要求）

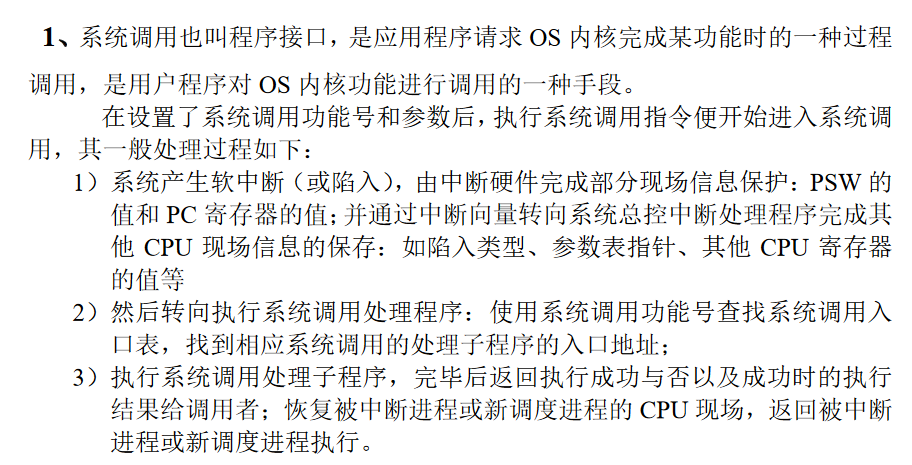
（所谓的层次结构，就是把操作系统所有的功能模块按照功能调用次序分别排成若干层，各层之间的模块只有单向调用关系(例如，只允许上层或外层模块调用下层或内层模块)。分层的优点是： (1)把功能实现的无序性改成有序性，可显著提高设计的准确性。 (2)把模块问的复杂依赖关系改为单向依赖关系，即高层软件依赖于低层软件。）

1. 引入进程的概念后，必须为程序创建进程后程序才能运行，因此进程和程序间是一一对应的。 错误！！！ 错在：一个程序可以对应多个进程
2. 
3. 主存就是内存；
4. 通道是dma的发展，可以进一步减少cpu干预；
5. 分区存储管理方案不能实现虚拟存储的原因是 作业地址空间不能超过存储空间。
6. 缺页中断和其他中断的区别：

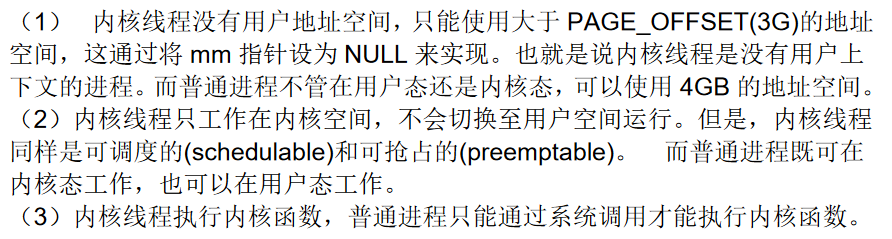


1. Linux系统提供的两种文件共享方式：

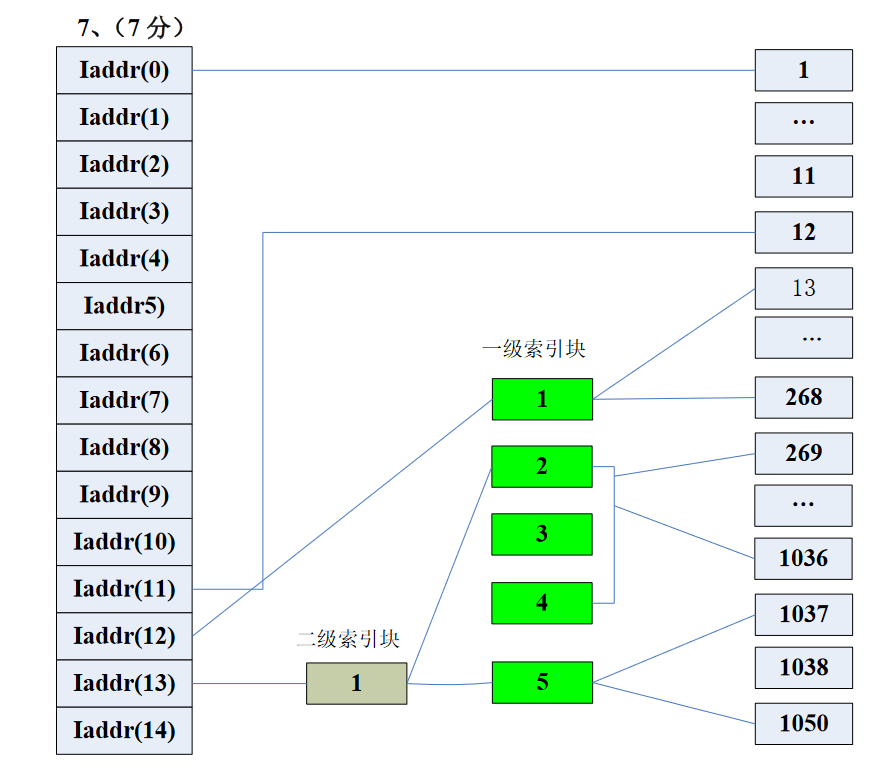


25、

26、Linux内核线程和普通线程的区别：



27、Linux的Ext2文件系统默认磁盘块大小为1KB，每个盘块号占4B。设文件file1有1050个逻辑块，试画出该文件在Ext2文件系统下的索引结构。



28、Linux文件名的长度不能超过2656个字符；

29、利用共享文件进行通信的方式 = 管道通信；

30、

<https://www.nowcoder.com/questionTerminal/b962f032c5814611a5ffba9989db0d62?orderByHotValue=1&page=1&onlyReference=false>

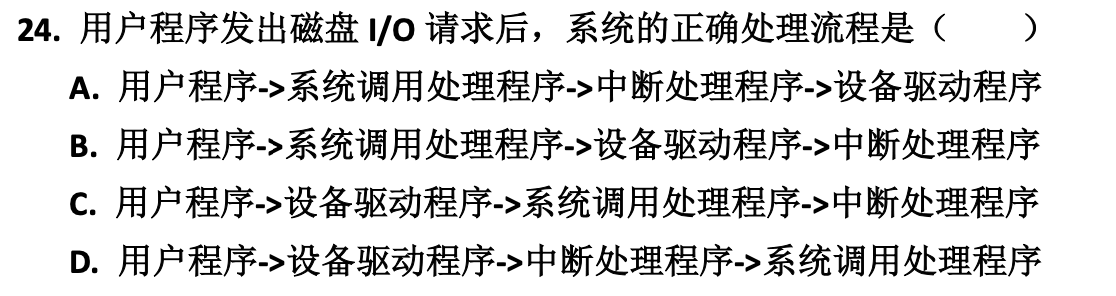
31、设备控制器是CPU与外围设备之间的接口，一个设备控制器可以同时连接多个设备。（是错误的）——可以控制多个设备。

32、现代操作系统都具有并发、共享、虚拟和异步的特性，并发是其它三个特征的前提。

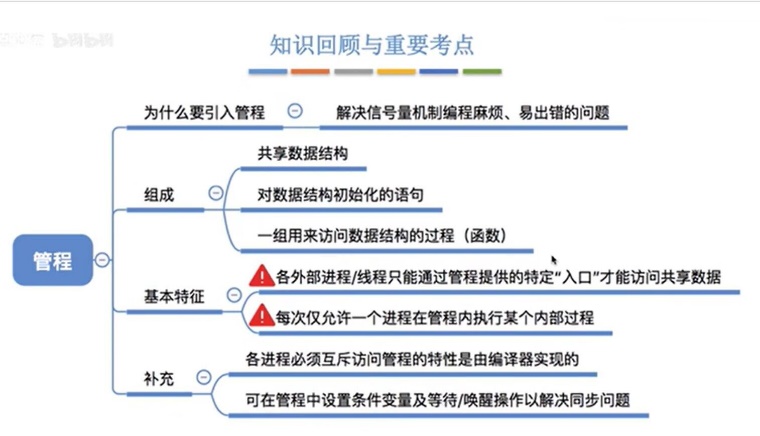
33、在单用户系统环境中，I/O队列的长度通常为1，因此，先来先服务FCFS算法是最经济实惠的磁盘调度算法。（没有饥饿问题）

34、引入多道程序技术的前提之一是系统具有中断功能。

35、B：



36、



37、分页：操作系统负责；分段：编译器负责；分页速度快于分段；

38、Unix系统中采用了：信号量集、管程机制和软中断信号机制；

39、屏蔽中断和关中断的概念不一样，屏蔽中断是指不允许发生中断，而关中断是为了避免中断寄存器的值被其他的中断发生更改。

40、中断可分为可屏蔽中断和不可屏蔽中断。

可屏蔽中断 ：可被CPU通过指令限制某些设备发出中断请求的中断

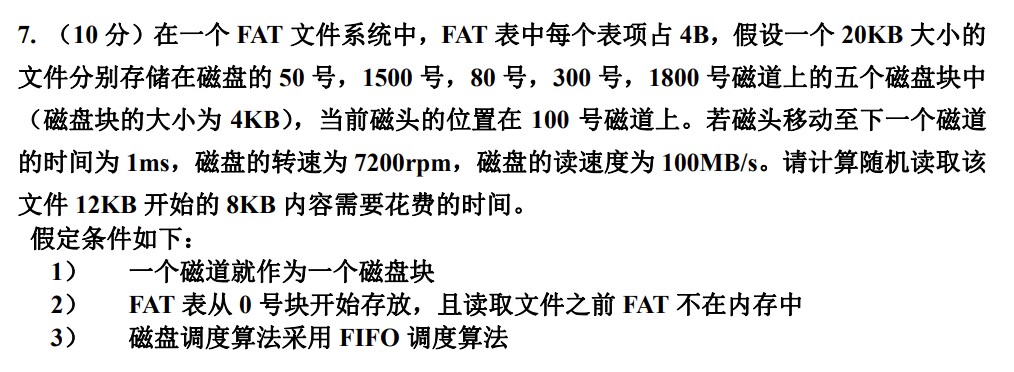
不可屏蔽中断：不允许屏蔽的中断如电源掉电

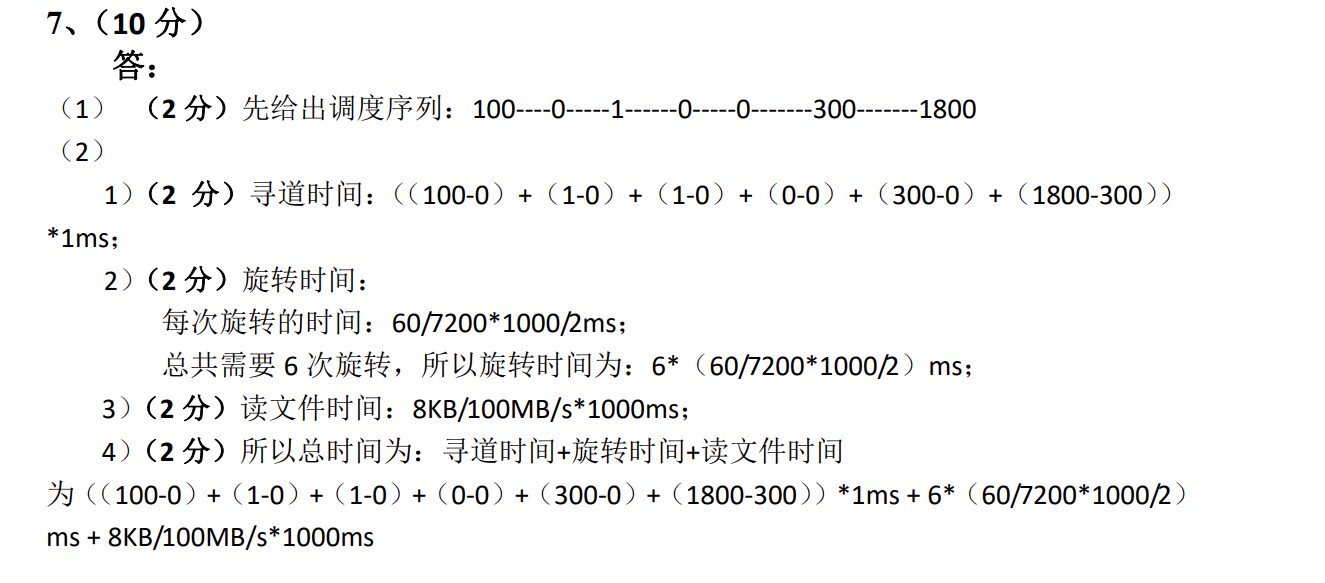
41、UNIX文件系统：

<https://www.nowcoder.com/questionTerminal/b99d824616fb485fa390a93cd38b329b?mutiTagIds=642&onlyReference=false&orderByHotValue=1>

42、磁盘容错技术往往也称为系统容错技术SFT（System Fault Tolerance）。分为三个级别：SFT-1是低级磁盘容错技术，主要用于防止磁盘表面发生缺陷所引起的数据丢失；SFT-II是中级磁盘容错技术，主要用于防止磁盘驱动器和磁盘控制器故障所引起的系统不能正常工作；SFT-III是高级系统容错技术。

**43、假设一个 16KB 大小的文件存储在 FAT 文件系统中，FAT 表存放在磁盘开始位置，每个盘块号占 4B，每个磁盘块的大小为 4KB，文件占用的磁盘块号分别是100、200、300、400，每个磁道有 10 个磁盘块，当前磁头在 50 号磁道上。若磁头移动至下一个磁道的时间为 1ms，磁盘的转速为 7200rpm，磁盘的读速度为 4KB/ms。请计算读取该文件需要花费的时间。**





一个磁盘4KB 一个表项4B 所有一个磁盘块可以存1K个表项；

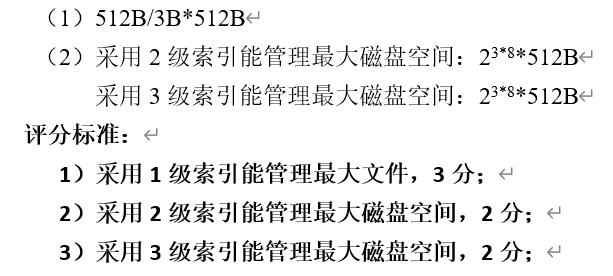
50 80 300 的FAT表项都在0号磁盘 其他的在1号；

数据在300 1800 去读数据；

44、某文件系统采用索引文件结构，假定文件索引表的每个表项占3个字节，存放一个磁盘块的块号（磁盘块的大小为512B）。试问

（1） 该文件系统能管理的最大文件是多少字节？

（2） 若采用2级或3级索引该文件系统能管理的最大磁盘空间又是多少字节？



45、主存储器与外围设备之间的数据传送控制方式有：程序直接控制、中断驱动方式、DMA 、通道控制。

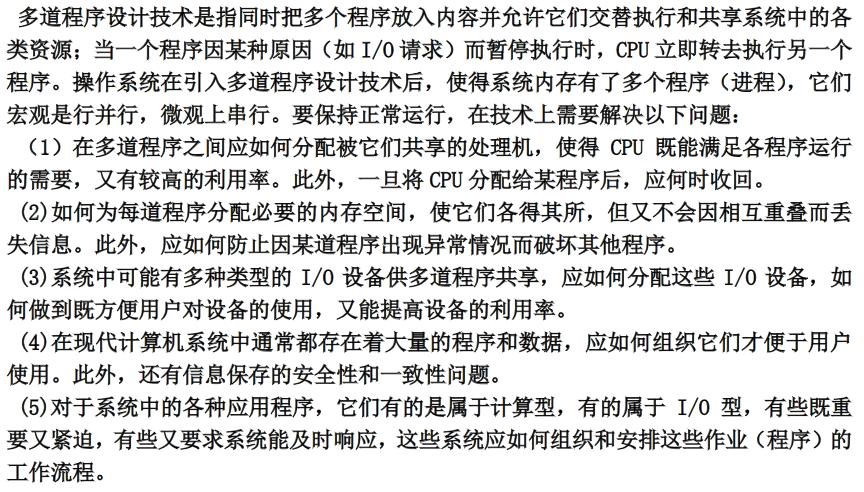
46、在Linux系统中，以文件方式访问设备；

47、有 10 个进程共享同一个互斥段，若每次最多允许 3 个进程同时进入互斥段，则用于实现互斥的信号量的变化范围是：[-7, 3]

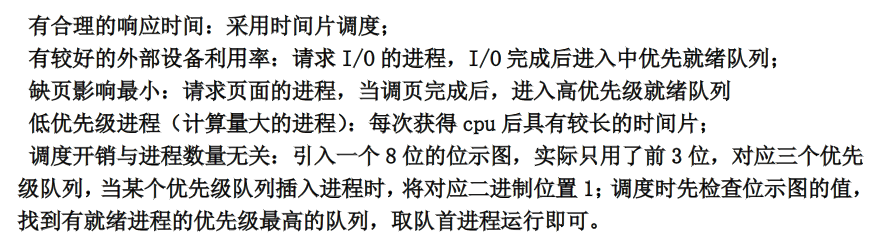
48、一个计算机系统的虚拟存储器，其最大容量是由计算机的地址结构决定的。

49、文件的物理结构分为：顺序（连续）结构、链接结构、索引结构

50、linux 文件系统中文件类型：普通文件、目录文件、块设备文件、字符设备文件、管道文件、符号链接文件、socket 文件。

51、

52、



53、可抢夺的资源分配策略可预防死锁,但它只适用于：主存空间和处理器

54、常见的文件系统有：fat32 ntfs exfat ext2 、ext3 、ext4；

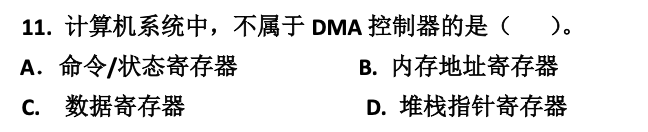
55、动态优先级其值范围100（最高优先级）到139（最低优先级）的数表示普通进程的静态优先级。值越大静态优先级越低。动态优先级的调度程序在选择新进程来运行时候使用的数，与静态优先级关系如下公式：

动态优先级 = max（100 ， min（静态优先级 – bonus + 5 ，139））

bonus是范围0-10的值，值小于5表示降低动态优先级以示惩罚，值大于5表示增加动态优先级以示奖赏。bonus的值依赖于过去进程平均睡眠时间。（注：平均睡眠时间是进程在睡眠状态所消耗的平均纳秒数）

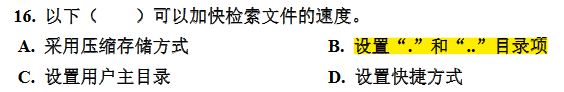
https://blog.csdn.net/weixin\_42092278/article/details/81431230

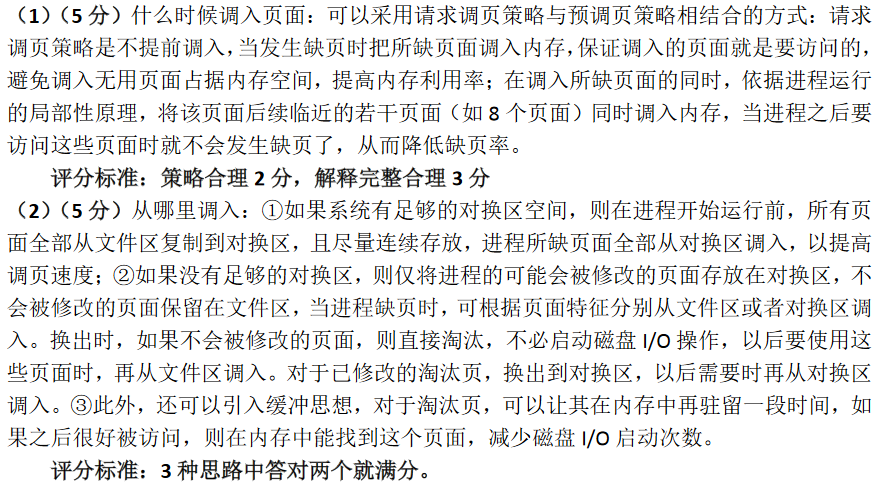
56、D:



57、CAW（通道地址字）是用来存放通道程序首地址的主存固定单元。

58、缓冲技术中的缓冲池在主存中。

59、

60、

61、操作系统经历了从多道批处理到分时系统发展的过程，在这个发展过程中涉

及的硬件支持变化和发展有哪些，请详细分析。

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

62、假设块大小为512B,块号为4B,Ext2系统可以管理的最大文件的大小是：6KB + 1/16MB + 129/128GB。（12+128+128\*128+128\*128\*128）\*512/（1024\*1024\*1024）=1.007879GB

63、线程是进程中的一个实体,是CPU调度和分派的基本单位。

线程具有许多传统进程的特征,故又称为轻型进程。传统的进程称为重型进程,相当于只有一个线程的任务。在引入线程的操作系统中,通常一个进程拥有若干个线程,至少也有一个线程。下面从调度、并发性、拥有资源和系统开销几个方面对线程和进程进行比较。

(1)调度。在传统的操作系统中,进程既是资源分配和拥有的基本单位,又是独立调度和执行的基本单位。而在引入线程后,则把线程作为调度和执行的基本单位,把进程作为资源分配和拥有的基本单位,把传统进程的两个属性分开,使线程轻装运行,从而显著提高系统的并发程度。同一进程中两个线程的切换不会引起进程切换,但由一个进程中的线程切换到另一个进程中的线程时,将会引起进程切换。

(2)并发性。在引入线程的操作系统中,不仅进程之间可以并发执行,而且在一个进程中的多个线程之间也可以并发执行,因而使系统具有更好的并发性,从而能更有效地使用系统资源和提高系统吞吐量。

(3)拥有资源。不论是传统的操作系统,还是引入线程的操作系统,进程都是拥有资源的独立单位。线程基本上不拥有资源(只有一点运行时必不可少的资源),但它可以访问其所属进程的全部资源。亦即一个进程的代码段、数据段以及系统资源(如已打开的文件、I/O设备等),可供该进程的所有线程共享。

(4)系统开销。系统在创建或撤消进程时,都要为之分配或回收资源,如内存空间、I/O设备等,因此系统所付出的开销将显著地大于创建或撤消线程的开销。同样,在进行进程切换时,需要保存当前进程的执行环境,设置和恢复被调度进程的执行环境,而线程切换只需保存和设置少量寄存器的内容,不涉及存储管理方面的操作,因而进程切换的开销也远大于线程切换的开销。

64、分时系统的响应时间（及时性）主要是根据：用户所能接受的等待时间 所确定的

65、Linux缺省的文件系统是：Ext系列

66、引入缓冲技术的主要目的是：

（1）解决信息的到达率与离去率不一致的矛盾；

（2）缓存起中转站的作 用；

（3）使得一次输入的信息能多次使用。

设备缓冲区的原则是：

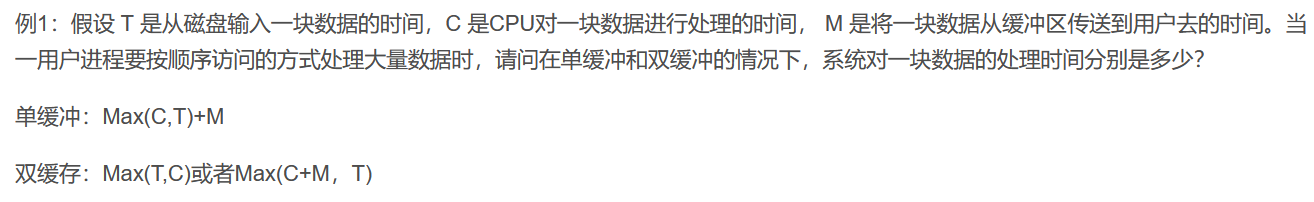
如果数据到达率与离去率相差很大，则可采用单缓冲方式；

如果信息的输入和输出 率相同（或相差不大）时，则可用双缓冲区；

对于阵发性的输入、输出，可以设立多个缓冲区。

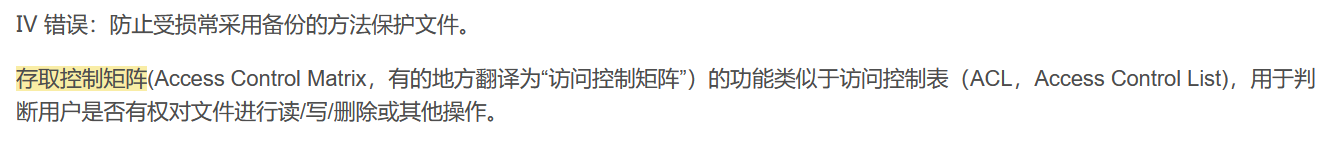
67、

68、JMP —— E

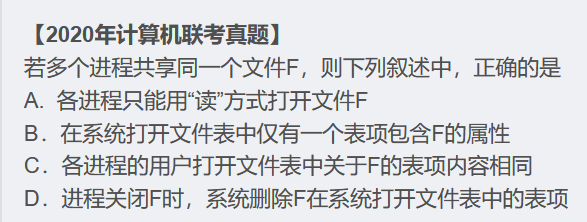
69、

70、对一个文件的访问，常由用户访问权限和文件属性共同限制。

71、



72、



73、删除文件的顺序：释放filel占用的磁盘空间、释放file1的文件控制块、删除目录dir中与filel 对应的目录项

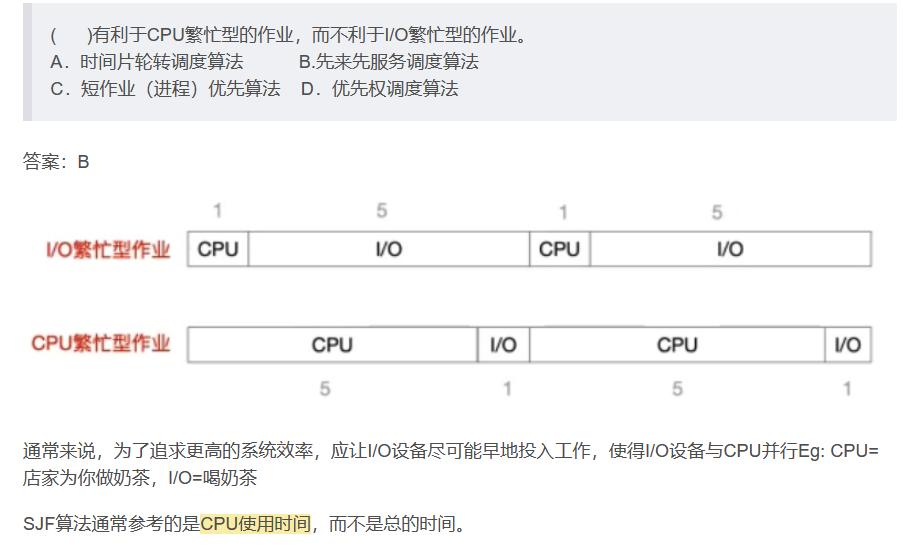
74、



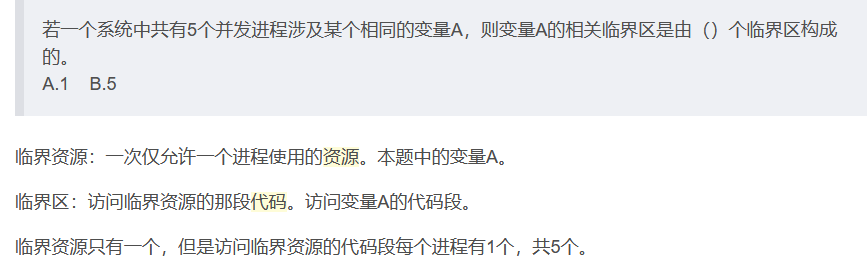
75、



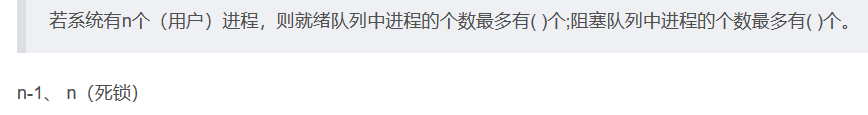
76、



77、

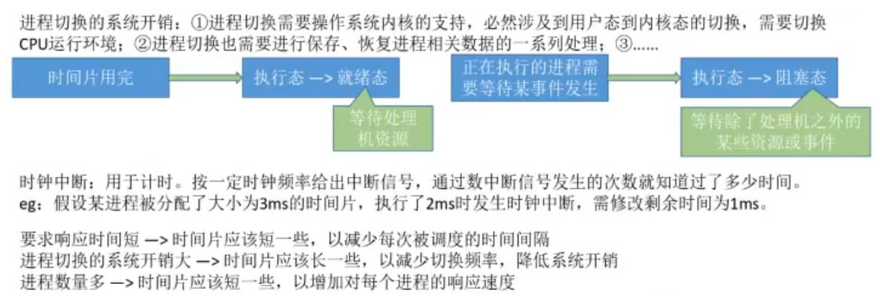


78、

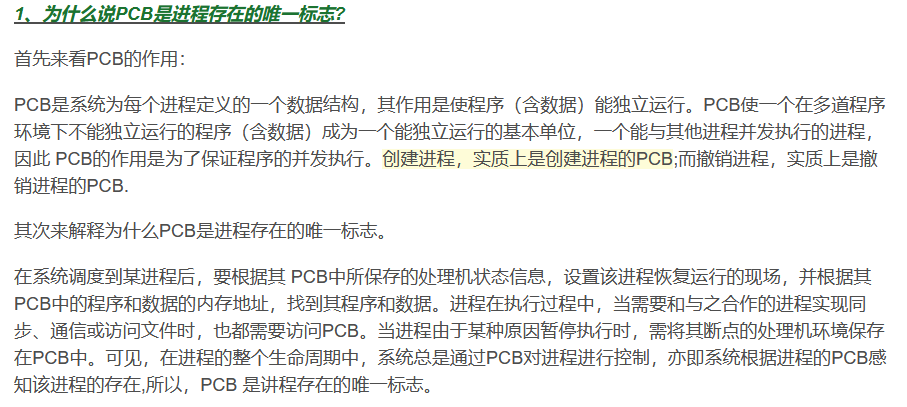


可以写两个 n - 1

79、



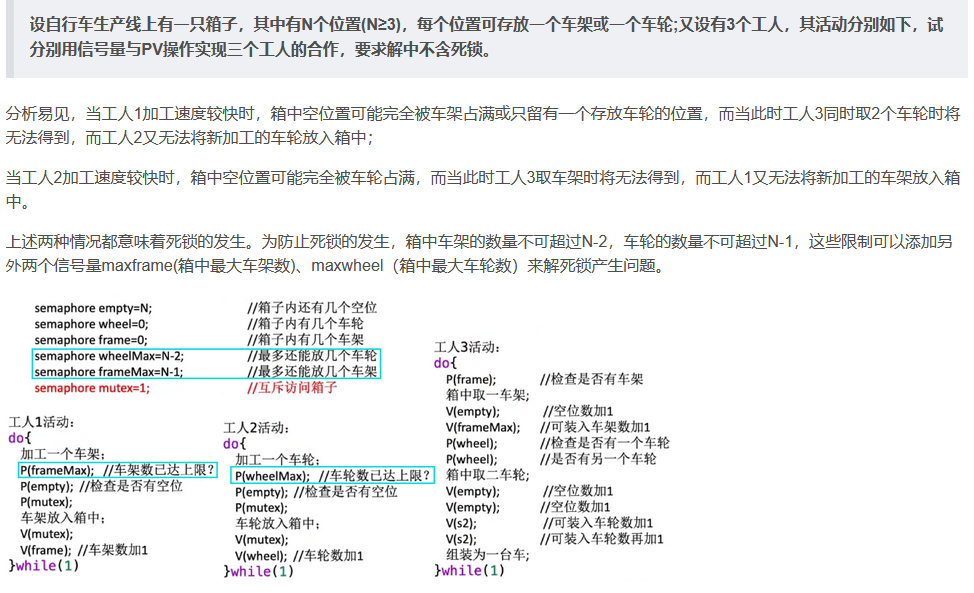
80、



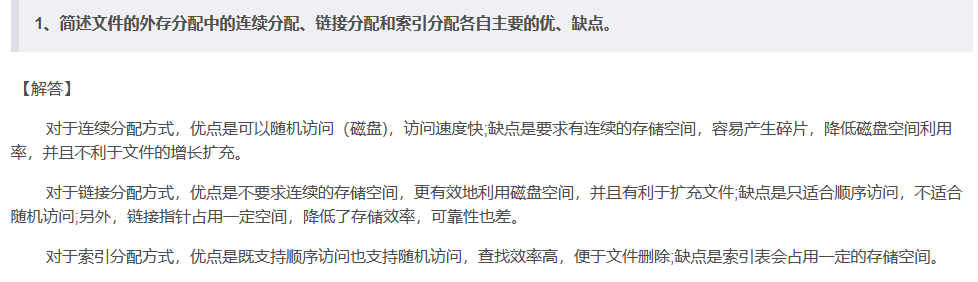
81、

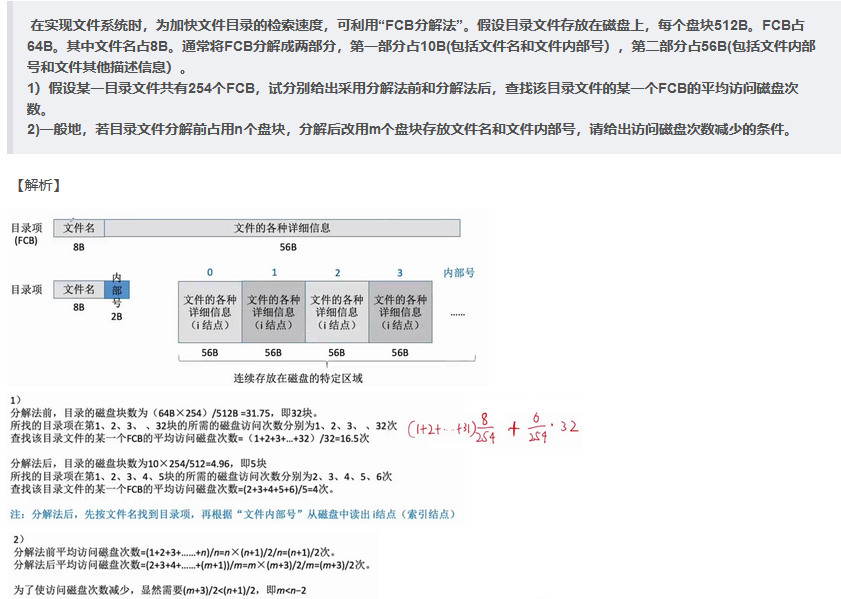


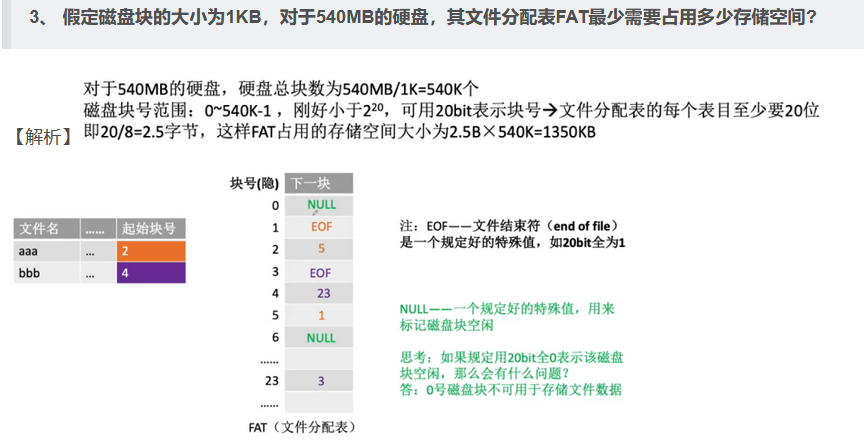
82、

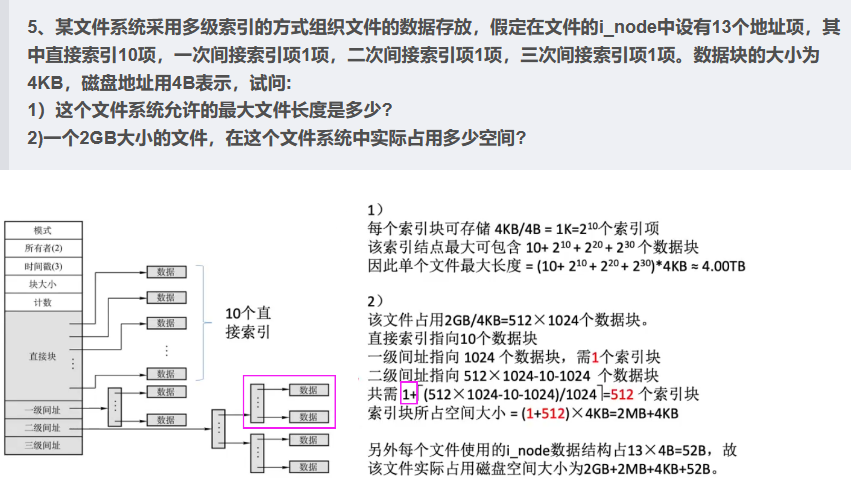
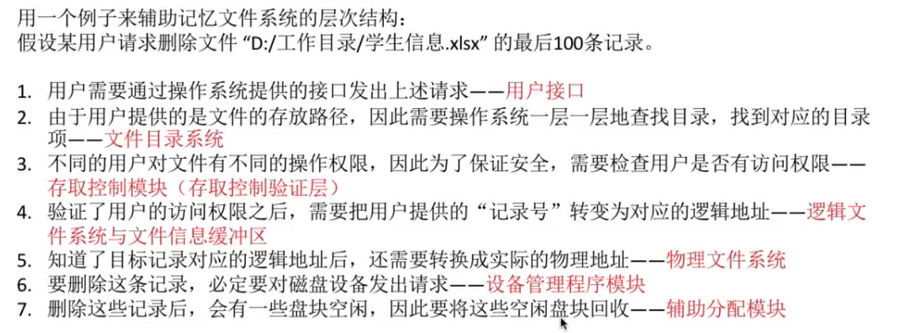


83、

84、

85、

86、

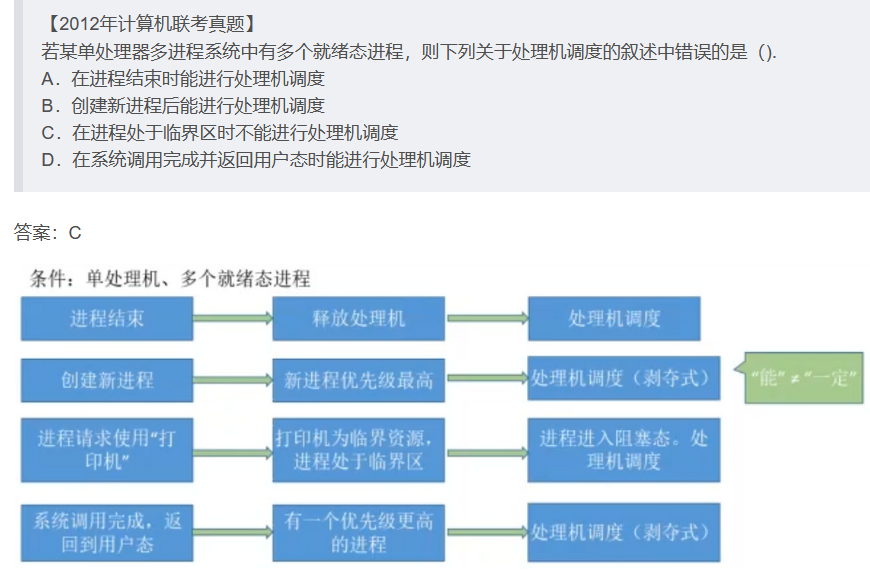
87、88、

89、1B = 1字节，64位os，一个字=8个字节，32位os，一个字=4个字节

90、

外部中断处理过程首先要保护现场，使得中断处理完之后能够恢复程序的执行状态继续执行。保护现场有两个含义：（1）由中断隐指令保存程序的断点（程序计数器）（2）由中断服务程序保存通用寄存器和状态寄存器的内容，中断服务程序是操作系统的一部分。

91、



92、**系统态（又叫管态、内核态、核心态、特权态）**  
操作系统的管理程序执行时机器所处的状态，又称处理机的特权级。  
在此状态下处理机可使用全部指令(包括一组特权指令)。  
使用全部系统资源(包括整个存储区域)。

**用户态（又叫目态、普通态）**  
用户程序执行时机器所处的状态称为目态（用户态）。  
在此状态下禁止使用特权指令。  
不能直接取用资源与改变机器状态。  
并且只允许用户程序访问自己的存储区域。

93、

