1. 概念解释

Chapter1

操作系统概念：操作系统是管理计算机硬件和软件资源的应用程序，操作系统能够对系统资源进行调度，进行存储、设备管理，还对计算机的保护和安全进行负责。

操作系统的三种基本类型：

批处理系统：用户将作业交给系统操作员，系统操作员将许多用户的作业组成一批作业(jobs)之后输入到计算机中，在系统中形成一个自动转接的连续的作业流，系统自动、依次执行每个作业。最后由操作员将作业结果交给用户。

实时系统：实时操作系统是指当外界事件或数据产生时，能够接受并以足够快的速度予以处理，其处理的结果又能在规定的时间之内来控制生产过程或对处理系统作出快速响应，并控制所有实时任务协调一致运行的操作系统。

分时系统：操作系统将CPU的时间划分成若干个片段,称为时间片。操作系统以时间片为单位，在用户间快速切换，轮流为每个终端用户服务，每次服务一个时间片。系统的快速切换使用户感到整个系统只为自己所用。

硬中断软中断：

硬中断是由外部事件引起的因此具有随机性和突发性；软中断是执行中断指令产生的

系统调用（System Call）：运行在使用者空间的程序向操作系统内核请求需要更高权限运行的服务。 系统调用提供了用户程序与操作系统之间的接口。大多数系统交互式操作需求在内核态执行

多道程序设计：多道程序设计总使CPU总有一个作业可执行

双重模式概念以及意义：为了区分操作系统代码和用户代码，分为用户态和核心态，当应用程序执行用户代码时则在用户态，当需要操作系统代码时就需要在核心态，用户态可以通过系统调用进入核心态。

Chapter2

Shell:shell就是命令解释程序，用来执行、获取用户的指令

向操作系统传递参数的方法：寄存器；内存中的块；堆栈

虚拟机：指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统

优点：运行在主机上，完全独立，[虚拟机](https://www.baidu.com/s?wd=%E8%99%9A%E6%8B%9F%E6%9C%BA&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)里面的所有操作不会影响主机，即使虚拟崩溃了，一般用来测试或与实验

缺点：它就是建立在主机上，毕竟是[虚拟机](https://www.baidu.com/s?wd=%E8%99%9A%E6%8B%9F%E6%9C%BA&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)，也是文件，很容易被破坏

Chapter3

进程：进程是CPU调度的最小单位，进程是一个活动的实体，不仅包含代码，还有数据PCB块等等。

PCB：用来存在进程的各类信息，进程状态、程序计数器、CPU调度信息等等

论述短期,中期和长期调度之间的区别：

a.短期调度:在内存作业中选择就绪执行的作业,并为他们分配CPU。

b.中期调度:作为一种中等程度的调度程序，尤其被用于分时系统，一个交换方案的实施，将部分运行程序移出内存，之后，从中断处继续执行。

 c.长期调度（作业调度程序）:从进程的缓冲池中，确定哪些作业调入内存以执行.

它们主要的不同之处是它们的执行的频率。短期调度必须经常调用一个新进程，由于在系统中，长期调度处理移动的作业时，并不频繁被调用，可能在进程离开系统时才被唤起。

Chapter4

线程（thread）：是操作系统能够进行运算调度的最小单位。它被包含在进程之中，是进程中的实际运作单位。一条线程指的是进程中一个单一顺序的控制流，一个进程中可以并发多个线程，每条线程并行执行不同的任务。

优点：响应度高；资源共享；经济

Chapter5

抢占调度：cpu决策可能在以下四个情况中发生，①进程从运行->等待②运行->结束③运行->就绪④等待->就绪，当调度只发生①②时，则称调度是非抢占的，否则，就是抢占的

吞吐量：一定时间内CPU处理进程的数量

周转时间：进程从提交到完成的时间

等待时间：进程在就绪队列中等待的时间

相应时间：从提交到第一次相应的时间

FCFS SJF SRJF RR 优先级调度

Chapter6

临界区：每个进程当中有一段代码段成为临界区，在临界区中进程可以改变共同变量，当进程进入临界区内，没有其他进程可被允许进程临界区改变共同变量。

临界区三原则：互斥、前进、有限等待

竞争条件（race condition）：多个进程并发访问和操作同一数据并且执行结果与访问发生的特定顺序有关，称为竞争条件。

经典同步问题：

有限缓冲：

生产者：

Wait(empty)

Wait(mutex)

生产商品放入

Signal(mutex)

Signal(full)

消费者：

Wait(full)

Wait(mutex)

消费

Signal(mutex)

Signal(empty)

读者写着问题：

读者：

Wait(mutex)

Readcount++

If(readcount==1){

Wait(wrt)

}

Signal(mutex)

读

Wait(mutex)

Readcount--

If(readcount==0){

Signal(wrt)}

Signal(mutex)

写者：

Wait(wrt)

写

Signal(wrt)

哲学家进餐：

Wait(chopstick[i])

Wait(chopstick[(i+1)%5])

吃饭

Signal(chopstick[i])

Signal(chopstick[(i+1)%5])

管程（monitor）：包括一组变量的声明和函数的实现，用来实现进程同步互斥

Chapter7

死锁：一组进程中的每个进程都在等待某个事件，而这个事件只能由这一组进程的另一个进程产生。

死锁产生的必要条件：互斥、占有并等待、非抢占、循环等待

安全状态：如果按照某个顺序能够为每个进程分配资源并且不产生死锁，那么，这个状态就是安全的，这个序列叫做安全序列。

银行家算法：allocation max need avaliable

Chapter8以后看书吧

RAID：其基本思想就是把多个相对便宜的硬盘组合起来，成为一个硬盘阵列组，使性能达到甚至超过一个价格昂贵、容量巨大的硬盘。根据选择的版本不同，RAID 比单颗硬盘有以下一个或多个方面的好处：增强数据集成度，增强容错功能，增加处理量或容量。另外，磁盘阵列对于计算机来说，看起来就像一个单独的硬盘或逻辑存储单元。简单来说，RAID 把多个硬盘组合成为一个逻辑扇区，因此，操作系统只会把它当作一个硬盘。RAID 常被用在服务器计算机上，并且常使用完全相同的硬盘作为组合。

设备驱动程序（Device Driver）：是一个允许高级（High level）计算机软件（computer software） 与硬件（hardware）交互的程序，这种程序创建了一个硬件与硬件，或硬件与软件沟通的接口，经由主板上的总线（bus）或其它沟通子系统（subsystem）与硬件形成连接的机制，这样的机制使得硬件设备（device）上的数据交换成为可能。使得操作系统能够控制硬件设备。

工作集：为了保证内存能够给各个进程分配合适的页数量，通过检查最近△个页的引用，这最近△个引用的页形成的页集合成为工作集，然后OS分配给进程大于工作集合的帧数

FCB：为了便于对文件进行控制和管理，在文件系统内部，给每个文件惟一地设置一个文件控制块，这种数据结构通常由下列信息项组成：

1.文件标志和控制信息

2.文件逻辑结构信息

3.文件物理结构信息

4.文件使用信息

5.文件管理信息

假脱机（SPOOLING）：用来保存设备输出的缓冲区，这些设备不能接受交叉的数据流，应用程序将输出存到假脱机上，假脱机对假脱机文件进行排队，并一次复制一个假脱机文件到设备上。

VFS：为操作系统提供不同文件系统的接口

颠簸:如果对进程没有分配足够的页，那么当需要使用该页时，并且在全局模式下，系统会进行页置换，但是被置换的页在未来仍然需要马上用，并形成恶性循环，进程在页置换停留的时间要长于进程处理的时间。

1. 简答题

·简述磁盘分配的三种方式（continuous linked indexed）和各自优缺点

Continuous：将文件存储在一段连续的磁盘块中

优点：①用于连续分配文件所需要的寻道数少，因此所需要的寻道时间也小②连续分配的访问比较简单，可以支持顺序访问和直接访问

缺点：①存在外部碎片（可以通过合并处理，将磁盘上的内容->复制到备份磁盘磁带->复制回原磁盘，停机时候执行，比较麻烦）②每次创建文件时，都需要确定文件大小（可以采用扩展的方法，给文件增加额外的磁盘块）

Linked：文件在磁盘中通过一系列链接着的块组成

优点：①不存在外部碎片②创建文件时不需要确定文件大小

缺点：①只能进行顺序访问②存储指针需要空间③存在可靠性问题，指针丢失或者损坏

Ps.针对链接分配不能进行直接访问可以使用FAT表

Indexed：每一个文件维护一个指向所有文件所使用的磁盘快的索引表

优点：①没有外部碎片②可以进行直接访问顺序访问

缺点：①指针的空间开销

·简述死锁避免、死锁预防并比较区别

死锁预防针对占有并等待、非抢占、循环等待三个条件，而死锁避免只针对循环等待一个条件。

死锁预防是系统预先确定一些资源分配策略以及调度手段，进程按规定申请资源，系统按预先规定的策略进行分配，从而防止死锁的发生。

死锁避免是当进程提出资源申请时系统测试资源分配，仅当能确保系统安全时才把资源分配给进程，使系统一直处于安全状态之中,从而避免死锁。

·访存操作可能会导致IO的进行，某进程读写文件时可能并没有IO设备进行，为什么

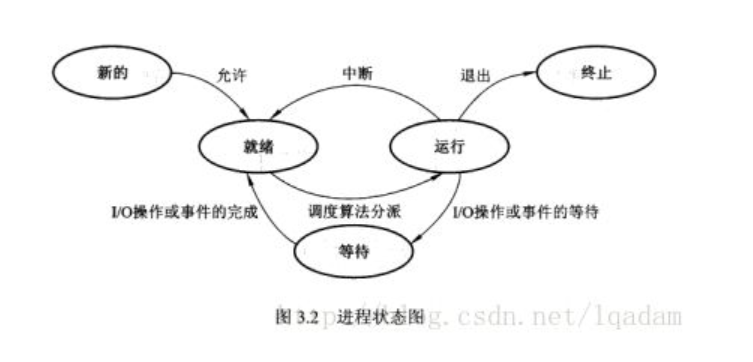
读写文件首先是CPU读写内存，然后内存中的数据与设备进行IO交互。所以读写文件时在CPU和内存交互时并没有IO设备参与（此时数据可能在缓冲区、高速缓存或者假脱机中）

·CPU和操作系统在分页中各自承担了那些工作，简要说明

CPU:生成逻辑地址；

OS：将内存和进行进行分块；选择进程中适当的页调入内存；维护页表和帧表；根据逻辑地址找到物理地址

·进程状态图



·简述阻塞、饥饿、死锁的区别

阻塞：某一事件由于等待某一资源或者其他事件的发生而处于停滞状态

饥饿：某一事件由于等待某种资源而长期处于停滞状态，若该事件一直得不到资源，那么这个事件就会被饿死

死锁：存在一系列进程，分别占有某种资源，这些进程均在等待某一事件的产生，而该事件则只能由这一系列事件中的另一个进程产生，于是，就形成了死锁

·从硬件层面、文件管理、设备管理、存储管理、进程同步等方面简述保护的概念

陷入内核

硬件层面：操作系统提供监督模式和用户模式

内存管理：①为了防止用户在内存中访问到操作系统进程，需要通过基地址寄存器和界限地址寄存器来判断是否安全②在页表中通过设置有效位无效位来防止非法地址访问

文件管理：①提供文件锁，防止其他进行进程访问②通过定义文件的读写等访问类型，来限制用户的操作③可以规定不同级别的用户对文件进行不同级别的操作④对文件进行加密设置⑤对每个文件提供ACL

设备管理：通过一些IO中断

进程管理：保证进程独立性，进程同步时避免产生竞争条件

·阻塞IO和非阻塞IO的区别

阻塞 I/O，当应用程序发出一个阻塞系统调用时，应用程序挂起，应用程序从运  
行队列转入等待队列。等系统调用完成之后再回到就绪队列，在合适的时候继续运行。绝大多数操作系统为应用程序提供的都是阻塞系统调用，因为它代码更加简单，更容易理解。  
非阻塞 I/O，一个非阻塞调用在程序执行过长时间并不中止应用程序，它会很快  
返回，其返回值表示已经传输了多少字节。在多个程序协作完成 I/O 的工作时，可能会使用非阻塞 I/O

·分页式文件系统，页表的主要数据结构，使用分页有哪些硬件支持

页表，PTBR TLB 磁盘

1. 计算题

·某磁盘逻辑地址32位，页大小16K，页表项大小4B

①采用多层页表结构，该采用几层页表？页偏移多少比特？画出地址分配。

②对逻辑地址54321（10进制），简述求实际地址的过程（忽略缺页中断）

①页大小：16KB=2^14 页偏移位数为14

|  |  |
| --- | --- |
| 页号 | 页偏移 |
| 32-14=18 | 14 |

因此根据逻辑地址可以知道，逻辑地址想要的页号在0-2^18之间

假设页表在内存中占有一页的大小，那么

页表中可以存储16KB/4B=2^14/2^2=2^12个页号 小于2^18

所以一级页表结构是不科学的

采用两级页表

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 6 | 12 | 14 |

②54321/2^28=0……x

Etc

1. 信号量问题

·从前有座山，山上有座庙，山下有口井。庙里小和尚需要挑水。有人舞担，有人拿桶，有人诵挑水秘诀。挑水时，三个和尚必须一人持担，一人拿桶、一人诵挑水秘诀(同时进行)后方能挑水。每个和尚都是先喜欢诵诀，其次持担、其次持桶。请写出信号量和相关伪代码。

设持担、拿桶、诵经的人数分别为SD，ST，SJ，初始化为0

P(MUTEX)

If(SJ==0){

P(J)

SJ++

}

V(MUTEX)

P(MUTEX)

Else if(SD==0){

P(D)

SD++

}

V(MUTEX)

P(T)

·一个场地最多容纳22人，其中有打篮球的（不妨设为活动A），打羽毛球的（不妨设为活动B）。

如果人数达到上限则等待

如果活动A的人数比活动B的人数多5人，则参与A的同学需等待；如果活动B的人数比活动A的人数多5人，则参与活动B的同学需等待。

活动A和活动B的同学可随时离开

请用“参与A”“参与B”“离开A”“离开B”和信号量来说明这个过程。

A：

P(student)

While(SA-SB>5){

}

P(MUTEX)

SA++

V(MUTEX)

进行活动

P(MUTEX)

SA--

V(MUTEX)

V(student)