深圳大学2015年操作系统期末考试复习提纲

**红色字体部分为本学期考试大题涉及的内容，不包括选择题**

本提纲内容搞懂了及部分概念背诵了既可以拿A

附加题考了固态硬盘，还用信号量同步

制作人：2012170150 吴少滨

第一章

1.操作系统的定义

操作系统是一组控制和管理计算机硬件和软件资源、合理地对各类作业进行调度（有效性），以及方便用户（方便性）的程序的集合

2.操作系统的目标

方便性：配置OS后可使计算机更容易使用（不需要手工输入0,1码）

有效性：有效控制和管理计算机各种软硬件资源，提高资源的利用率

可扩充性：便于扩充新功能

开放性：不同机型可运行相同的程序

3.操作系统的作用：

（1）从用户的角度看：OS是用户与计算机硬件系统之间的接口

（2）从计算机资源的角度看：OS是计算机系统资源的管理者

（3）从功能扩充的角度看：实现计算机资源的抽象，增加了OS的计算机，成为功能更强使用更方便的扩充机器或虚机器

4.单道批处理系统特征：自动性、顺序性、单道性。

5.多道批处理系统特征：多道性、无序性、调度性

6. 分时系统特征：多路性、独立性、及时性、交互性 目的：提高资源的使用方便性

7.操作系统的特征：

并发性：多道用户程序可在同一时间间隔中运行

共享性：系统资源可供内存中多个并发的进程共同使用（包括互斥共享和同时访问）

虚拟性：系统物理资源可虚拟为多个逻辑资源

异步性：内存中多个并发的进程以异步方式运行

8.操作系统的功能

（1）处理机管理：进程控制，进程同步，进程通信，进程调度

（2）存储器管理：内存分配，内存保护，地址映射、内存扩充

注：虚拟存储技术主要采用请求调入和置换功能实现内存扩充

（3）设备管理：缓冲管理，设备分配，设备处理，设备独立性，虚拟设备

（4）文件管理：文件存储空间管理，文件系统

（5）用户接口：命令接口，程序接口，图形接口

第二章

1.为什么需要进程

为了使程序在并发、共享、异步的环境下能正常运行，必须专门设置一个控制数据区，为程序保留运行的现场

2.进程与程序的区别

进程是动态的，程序是静态的(是指令的集合)

一个程序可以包含多个进程

进程可以描述并发活动，程序则不明显

进程执行需要处理机，程序存储需要介质

进程有生命周期，程序是永存的

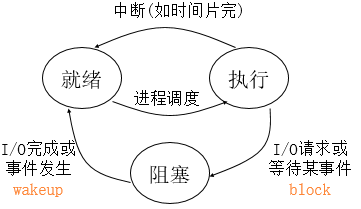
3.进程的定义

进程是程序的一次执行

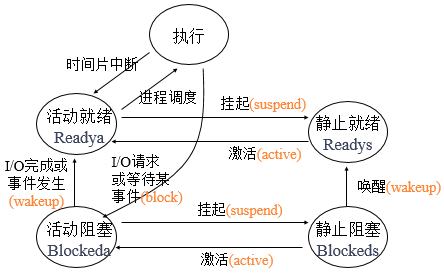
进程是进程实体(包括程序段、数据和PCB)的运行过程，是系统进行资源分配和调度的一个独立单位

4.进程的特征：结构性，动态性，并发性，独立性，异步性

**5.进程基本状态转换**

+

**6.具有挂起状态的进程状态转换**



7.进程控制块PCB的作用

描述进程的变化过程

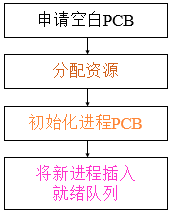
记录进程的外部特征

记录进程与其他进程的联系

是进程存在的唯一标志

系统通过PCB控制和管理进程

8.进程建立



9.临界资源：一个时刻只能由一个进程使用的资源

10.临界资源使用的同步准则

空闲让进：（提高效率）

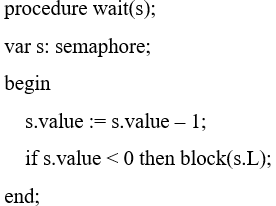
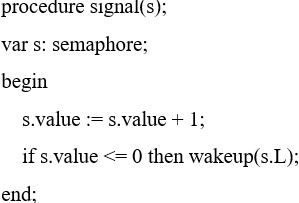
忙则等待：（解决互斥）

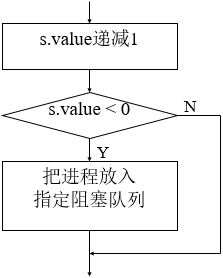
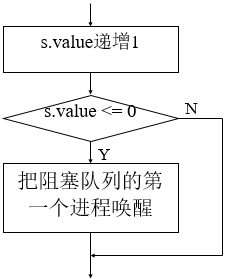
有限等待：等待进入临界区的要求应在一有限时间满足（以免死等）

让权等待：放弃占用CPU（以免忙等）

11.信号量：一个整型变量+wait（S）：等待操作（P操作）+signal（S）：发信号操作（V操作）

（1）P原子操作（wait）： （2）V原子操作（signal）

**12.生产者消费者问题（初始化）P58**

**13.哲学家进餐问题P61**

14.管程的定义

（1）一个管程定义了一个数据结构和能为并发进程所执行（在该数据结构上）的一组操作，这组操作能同步进程和改变管程中的数据

（2）管程实际上是一种能实现进程同步的特殊的子程序（函数、过程）的集合

15.管程的优点（作用）

（1）使用临界资源的进程进行调用时非常简单

（2）进程结构清晰

（3）易于查错

16.进程通信的类型

（1）共享存储器系统（无格式）：进程间以共享存储器方式进行数据通信

（2）消息传递系统（有格式）：

进程间的数据交换以消息（message）为单位

操作系统直接提供一组命令（原语）实现通信

（3）管道通信系统（相当于文件）：

17.线程是调度和执行的基本单位，进程是分配的基本单位

18.线程与进程的关系

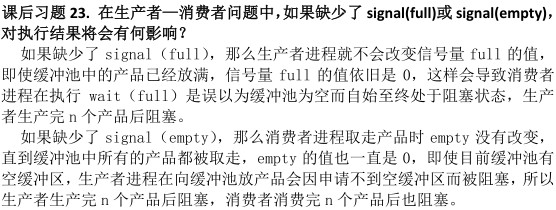
（1）线程是进程中的运行实体

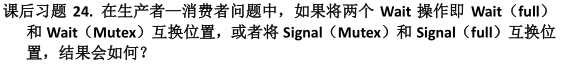
（2）一个进程可包含多个线程

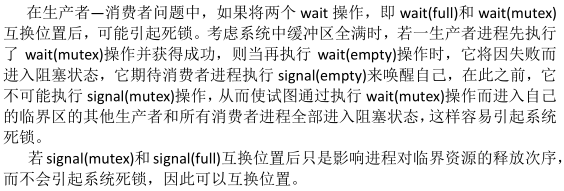
（3）一个进程中至少包含一个线程，称主线程

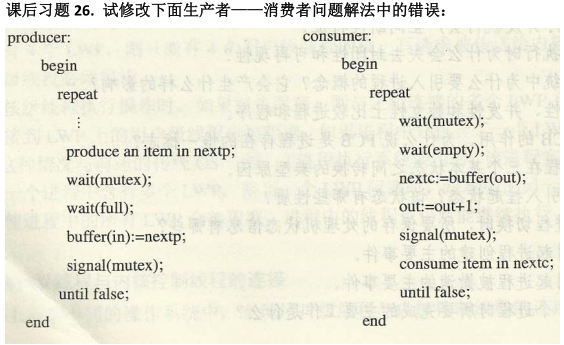
（4）进程相当于线程的载体

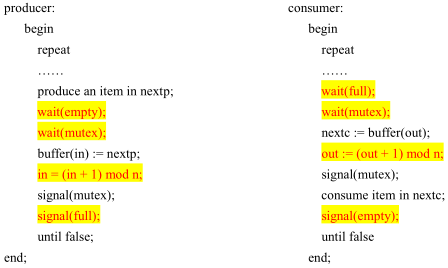
19.线程的属性：轻型进程，独立调度和分派的基本单位，可并发执行，共享进程资源

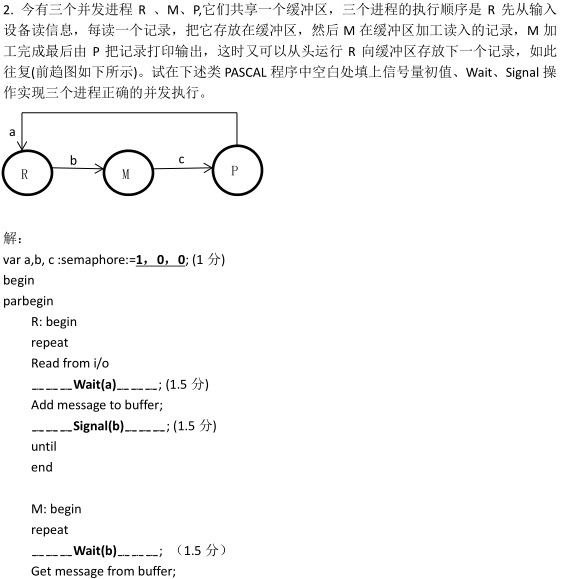


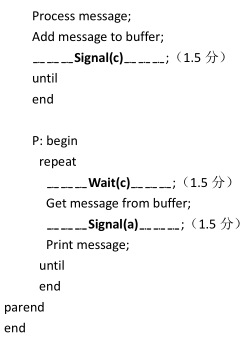




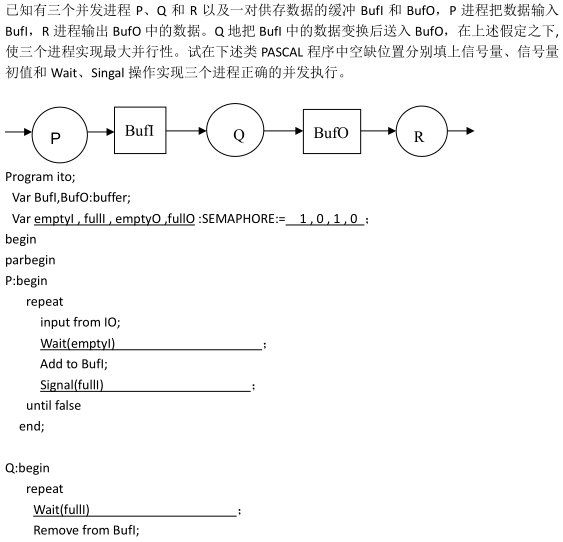


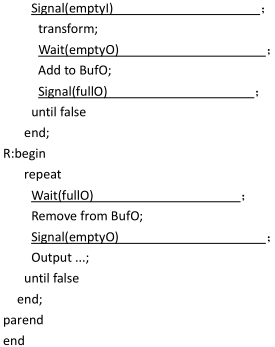






**考试题型如这道题：（所谓的代码题）**





第三章

1.作业调度的类型、作用及区别

（1）高级调度：即作业调度。

根据调度算法和计算机当前状态，挑选一个或多个后备作业投入运行

为选中的作业分配基本的内存和设备资源

为选中的作业建立进程，将进程实体装入内存

（2）中级调度：不用则调至内存外等待，用则调入内存

中级调度决定哪些进程可参与竞争CPU

中级调度将进程从活动态（活动就绪、活动阻塞）变为静止的挂起态（静止就绪、静止阻塞）；或相反

中级调度实际上是实现“挂起”和“激活”操作

中级调度也称为进程交换调度，通常仅用于分时系统

（3）低级调度

低级调度即进程(线程)调度

低级调度决定哪个进程可获得CPU

低级调度从活动就绪队列中挑选一个进程，将它变为运行态，同时启动CPU执行该进程

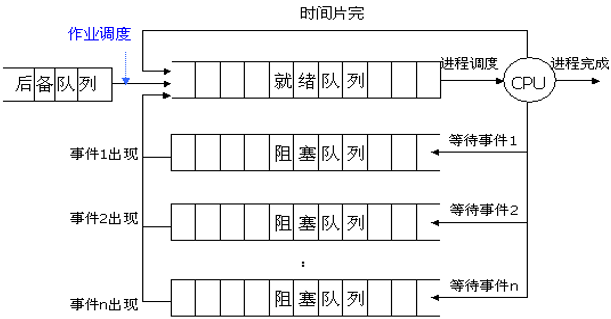
低级调度也称微观调度

2.调度队列模型

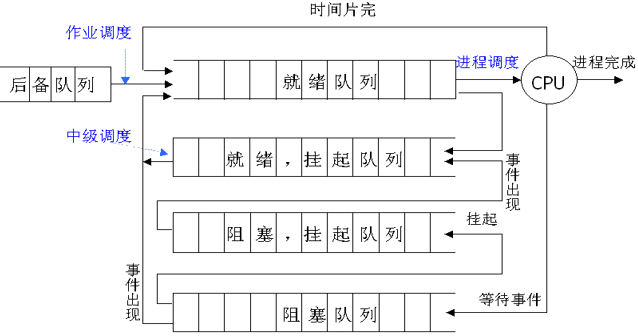
（1）只有低级调度的调度队列模型



（2）具有高级调度和低级调度的调度队列模型



（3）具有三级调度的调度队列模型



3.周转时间：Ti = 完成时刻 – 进入时刻

4.带权周转时间： 越小越好≥1

5.调度原则

（1）面向用户的原则

周转时间短

响应时间快

截止时间的保证

优先权准则

（2）面向系统的原则

系统吞吐量高

处理机利用率好

各类资源的平衡使用

**6.调度算法**

**（1）先来先服务（FCFS）**

****

**优点：简单，有利于CPU繁忙型作业（进程），有利于长时间作业（进程）**

**缺点：对短时间作业（进程）不利，对I/O繁忙型作业（进程）不利，对紧迫作业（进程）不利**

**（2）短作业优先（SF）**

****

**优点：有利于短时作业**

**缺点：**

**对长时间作业（进程）不利**

**未考虑作业（进程）的紧迫程度**

**抢占方式中，最短指总需要时间最短还是剩余时间最短（而且是估计值）**

**在抢占方式下，即使一个长作业（进程）正在运行，但也可能会被长时间地延迟**

**（3）高响应比优先（HRN）**

**响应比RP**

****

****

**优点：有利于短时作业，也有利于先来者**

**缺点：每次调度前，必须计算Rp，增加系统开销，未考虑作业（进程）的紧迫程度**

**（4）最高优先权（HPF）**

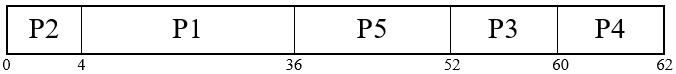
**静态优先权：优先权不变**

**动态优先权：优先权在运行过程发生改变**

****

**平均周转时间：T=39.6**

**平均带权周转时间：W=8.575**

****

**优点：可以根据要求，照顾到对系统、用户综合来说最优先的作业（进程）的执行**

**缺点：优先权的计算可能比较复杂，增加系统开销**

**（5）时间片轮转（RR）**

**q=1**

****

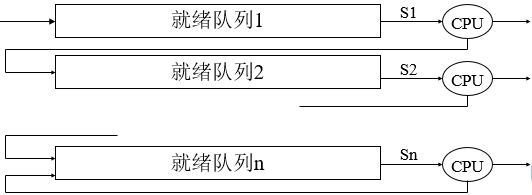
**优点：有利于交互性、事务性进程、有利于I/O繁忙型的进程**

**缺点：调度开销较大，未考虑实时响应要求**

**（6）多级队列调度算法**

**设置多个就绪队列,并从高到低赋予不同的优先级**

**每个队列采用RR算法，时间片长度从高优先级到低优先级依次增加（一般加倍）(S1<S2<…<Sn)**

****

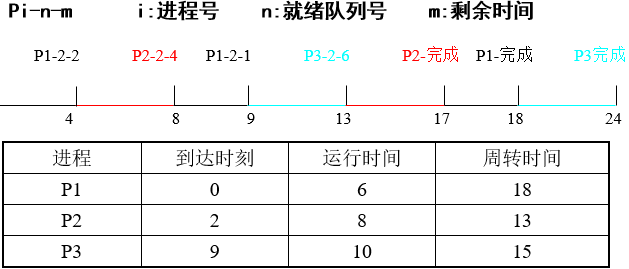
**特性：同一计算机系统存在多个OS**

**优点：可以同时兼顾到分时及批量处理任务**

**缺点：未考虑紧迫性作业或进程，调度算法比较复杂，调度开销较大**

**例：有一系统，采用三级反馈队列调度算法，时间片大小分别为：4，8，16，现有三个进程，到达时刻分别为0，2，9，执行时间分别为6，8，10，求每个进程的周转时间。**

****

****

**7.实时调度**

**（1）非抢占式调度算法：a.非抢占式时间片轮转调度算法 b.非抢占式优先调度算法**

**（2）抢占式调度算法：**

**a.基于时间终端的抢占式优先权调度算法**

**b.立即抢占的优先权调度算法**

**8.常用实时调度算法**

**（1）最早截止时间优先（EDF）算法 P101**

**根据任务的开始截止时间确定任务的优先级**

**开始截止时间越早，优先级越高**

**（2）最低松弛度优先（LLF）算法**

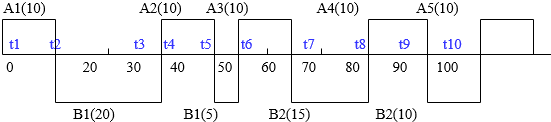
**松弛度（LF）=完成时间-处理时间-当前时间**

**例：任务A要求每20ms执行一次，执行时间10ms**

**任务B要求每50ms执行一次，执行时间25ms**

**t1( 0):LF(A1)=20-10-0 =10ms t2(10): LF(A2)=40-10-10=20ms**

**LF(B1)=50-25-0 =25ms**

****

9.死锁：指多个线程在运行过程中因争夺资源而造成的一种僵局，当进程处于这种僵持状态时，若无外力作用，他们都将无法再向前前进。

10.死锁的原因：

（1）竞争资源：当两个或以上进程需要两个或以上资源

（2）进程间推进顺序非法：请求和释放资源的顺序不当

11.产生死锁的必要条件

（1）互斥条件 （2）请求和保持条件 （3）不剥夺条件 （4）环路等待条件

12.预防死锁的方法

（1）摒弃“请求和保持”条件 （2）摒弃“不剥夺条件” （3）摒弃“环路等待”条件

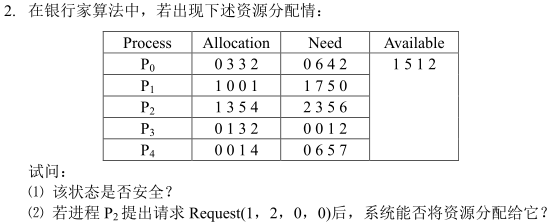
13.银行家算法 P109 PPT115

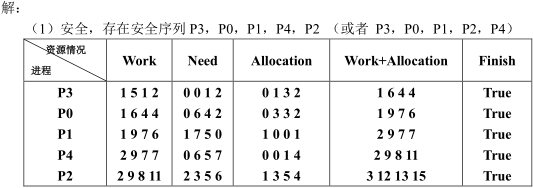
14.资源分配图 PPT126

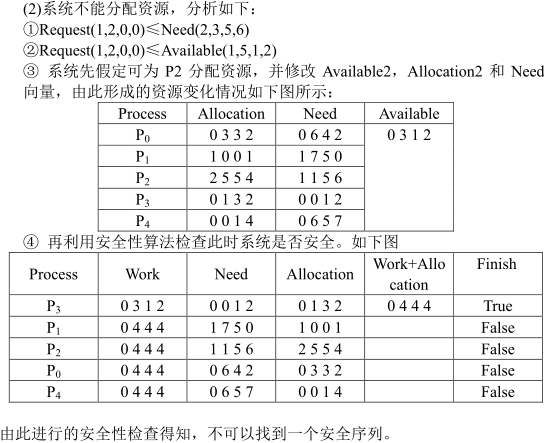
第三章作业











第四章

1.程序的装入

（1）绝对装入方式： 将模块装入到内存中事先指定的位置，逻辑地址与物理地址相同

（2）可重定位装入方式：从0开始，逻辑地址与物理地址不同

（3）动态运行时装入方式：内存的物理地址可发生改变

2.程序的链接：静态链接、装入时动态链接、运行时动态链接

3.存储器管理的目的

为用户使用存储器提供方便

⑴.用户只需在自己的逻辑空间内编程，不需要了解自己将放在内存中的物理位置，也不需要了解其它用户程序在内存中的物理位置

⑵.为用户提供充分大的存储空间（虚存管理）

充分发挥内存的利用率：让尽量多的用户程序调入内存运行

4.存储器管理的内容：

（1）内存分配：静态分配和动态分配

（2）地址映射：绝对映射、静态映射、动态映射

（3）内存保护：a.保护内存不被非法访问 b.不非法访问其它用户（包括系统）内存

（4）内存扩充：在逻辑上扩充内存的空间

5连续分配方式

（1）单一连续分配

一个用户程序独占连续的内存用户区

只能用于单用户、单任务的OS中

系统分两个内存区：系统区和用户区

（2）固定分区分配：划分多个区域可供多用户、多任务使用

①划分分区方法

a.分区大小相等：简单，大程序装不下，小程序浪费、

b.分区大小不等：将内存区分成多个较小的分区、适量的中等分区和少量的大分区

适应性强，特别大的程序可能仍装不下

②内存分配：

a.首次适应算法(FF)：按序选择第一个满足要求的内存区

b.最佳适应算法(BF)：仅当与程序大小相当的分区空闲时才予分配

****

**（3）动态分区分配**

**优点：可以按照用户进程实际大小，动态地分配内存空间，提高内存的使用效率**

**缺点：不管采用何种算法，都必将产生小的、不可利用的空闲分区（碎片）**

**①分配所用的数据结构**

**a.空闲分区表 b.空闲分区（双向）链表**

**②回收操作**

**无相邻空闲分区自己建立一个新表项**

**回收区与相邻的空闲分区合并以前一个空闲分区地址的首址为新空闲分区的首址**

**③分配算法**

**a.首次适应算法（FF）：**

**优点：保留高地址部分的大空闲区**

**缺点：低地址存在很多小的、无法利用的空闲分区，且查找时间较长**

**b.首次适应循环算法（CF）：从上次找到空闲分区的下一个分区开始，按序选择第一个满足要求的内存区**

**优点：空闲分区在内存中均匀分布,查找时间少**

**缺点：缺乏大的空闲分区**

**c.最佳适应算法（BF）：分区从小到大排列提高效率**

**优点：提高了内存使用效率，保留大的空闲区**

**缺点：存在许多很小的、无法利用的空闲分区**

**d.最坏适应算法（WF）：在整个空闲分区中查找最大空闲分区分割给作业（分区从大到下）**

**优点：不产生很小的碎片，查找效率高**

**缺点：缺乏大的空闲分区**

**e.快速适应算法（QF）**

**将空闲分区按大小分类，同一类设立一个空闲链表，根据进程长度寻找容纳它的最小空闲区链表**

**优点：查找效率高，不分割空闲去，保留大分区**

**缺点：算法复杂，开销大**

（4）动态重定位分区分配

①紧凑（拼接）：空闲分区的搬迁及合并

②动态重定位：要有动态定位机制支持

6.对换

对换是指把内存中暂不能运行的进程，或暂时不用的数据，换出到外存上，以腾出内存空间

整体对换：以进程为单位(挂起操作)

部分对换：以页或段为单位(虚拟存储器)

7.对换空间的管理

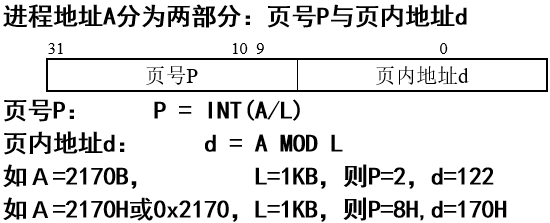
将外存分为文件区和对换区

对换区是连续的外存存储区

8.离散分配方式主要包括：分页存储管理、分段存储管理、段页式存储管理

9.内存物理块与进程页面：大小相等，页面可任意存放在任何物理快中，最后一页可不放满。

10.进程地址与页面地址



11.地址转换

（1）基本地址变换机构

**有一基本分页存储管理系统，页面大小为1024B，地址长度为16位，请问该系统共有多少内存块？写出逻辑地址8230H的页号和页内地址；如果该页对应的内存块号为18H，求该逻辑地址对应的物理地址。**

**①系统地址长度为16位；**

* **现每页大小为1024B = 210B，即页内地址占10位；**
* **因此系统中可表示页码的位数为：16-10=6位，即系统中内存块数为：26 = 64（块）。**

**②逻辑地址8230H展开成二进制为：1000,0010,0011,0000B**

* **页内地址占后10位，其余高位为页号；**
* **因此，页内地址为：10,0011,0000 = 230H；**
* **页号为：10,0000 = 20H。**

**③现该页对应的内存块号为18H，表示成二进制为：0001,1000B；**

* **块内地址与页内地址相同，即：10,0011,0000；**
* **将内存块号与块内地址合并后，得：0110,00 10,0011,0000B = 6230H。**

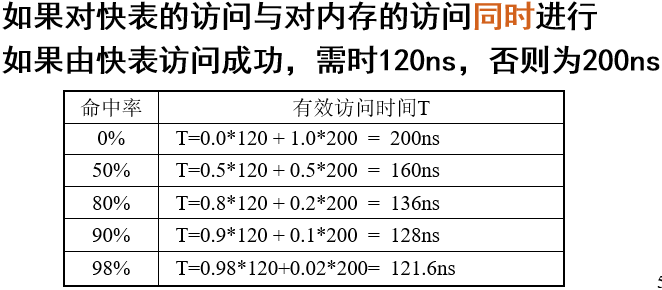
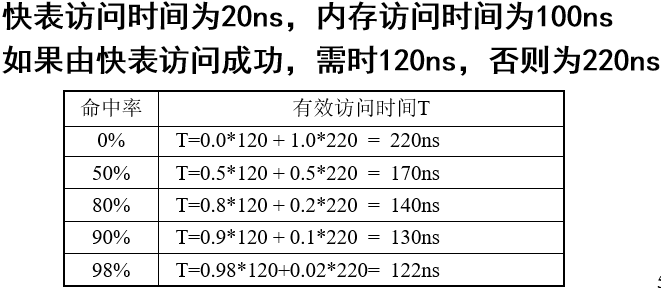
**缺点：访问一次逻辑地址，必须两次访问内存。**

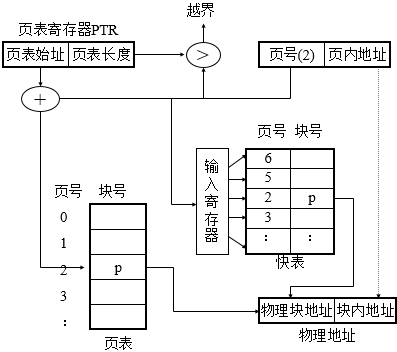
**第一次访问内存(即访问页表)为取得逻辑地址对应的内存物理地址**

**第二次访问内存为取得逻辑地址中的数据**

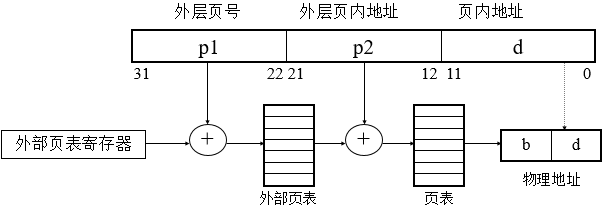
（2）具有快表的地址变换机构：内存也差，也到快表查

如果内存与快表不是同时进行：成功则只用访问一次内存120s，失败则要访问两次内存220s。





12.两级页表

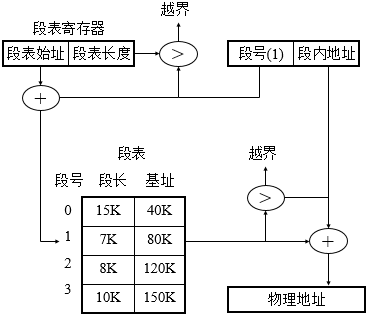


**有一基本分页存储管理系统，内存块大小为4096Bytes，每个块号占用4个Bytes，如果页表也以离散方式放在内存(块)中，求采用一级页表和两级页表，系统能支持文件的逻辑空间最多为多少？假如逻辑地址为123456H，求页内地址和外层页号。**

* **每个内存块占4Bytes，即每个页表项占4B**
* **每个内存块大小为4096Bytes，**
* **每个内存可放下最多4096/4=1024个页表项**
* **采用一级页表能支持文件的逻辑空间最多为：1024x4096=4MBytes**
* **采用二级页表能支持文件的逻辑空间最多为：1024x1024x4096=4GBytes**
* **4096占用12bits,即页内地址占12bits**
* **1024占10bits，即外层页内地址占10bits**
* **123456H=0001,0010,0011,0100,0101,0110b**
* **页内地址=后12bits=0100,0101,0110b=456H**
* **外层页号=(12+10)bits以上地址=00b=0H**

13.分段存管理的目的：**方便编程**、**分段共享**、**分段保护**、**动态链接、动态增长**

14.分段地址结构：作业的地址是二维结构，

16. 

17.信息共享：**段是信息的逻辑单位，在各自的段表，标上相同的段基址和段长度即可共享。**

**18.分页和分段的主要区别**

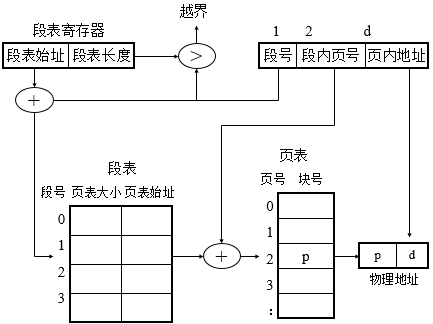
**页是信息的物理单位(满足系统管理的需要) 段是信息的逻辑单位(满足用户的需要)**

**页的大小固定（由系统决定） 段的大小不固定（由用户决定）**

**分页的作业（进程）地址是一维的　 分段的作业（进程）地址是二维的**

19.段页式管理的优点：既节省内存空间，提高内存分配的效率，又兼顾用户编程的需要



20.段页式地址变换机构

当进程要访问某个逻辑地址时，先取得段号

如果段号大于段表长度，越界错误

将段表始址与段号相加，得段表中位置

从段表中相应位置，得到该段页表的首址

将段内页号与页表首址相加，得页表中位置

从页表中相应位置，得到物理块号p

将块号p与页内地址d组合成物理地址

**练习**

**（1）有一基本分页存储管理系统，内存块大小为512Bytes，每个块号占有4个Bytes，如果页表也以离散方式存放在内存中，求采用三级页表，系统能支持文件的逻辑空间最多为多少。假如逻辑地址为15C3B6AH，求页内地址和外层页号。**

**每个内存块可以放512/4=128个内存块号。采用三级页表，系统能支持文件的逻辑空间最多为128\*128\*128\*512B = 16\*8\*128\*64\*2\*512B = 16\*1K\*64\*1KB=1K\*1K\*1KB=1GB。**

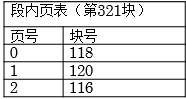
**假如逻辑地址为15C3B6AH，则转化为二进制是：1,0101,1100,0011,1011,0110,1010**

**页内地址占（512=29）9位，为1,0110,1010=16AH**

**每层页号需用（128=27）7个比特存储。首层页号为0011,101，二层页号为101,1100，外层页号为10，即2H。**

**（2）段页式存储器管理系统中，某作业的段表、段内页表格式如下，页的大小为1K，现有逻辑地址[2]<2248>(十进制)，求其对应的物理地址。**

****

****

**逻辑地址[2]<2248>，则页号为：2248/1024=2，找到321块2号页表块号为116，页内地址2248 mod 1024=200 (整除取余数)，对应的物理地址为：116\*1024+200=118984 (1D0C8).**

21.虚拟存储器

概念：

虚拟存储器是指具有请求调入功能和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统。其逻辑容量由内存容量和外存容量之和所决定，其运行速度接近于内存速度，而每位的成本却又接近于外存。

特征：多次性、对换性、虚拟性

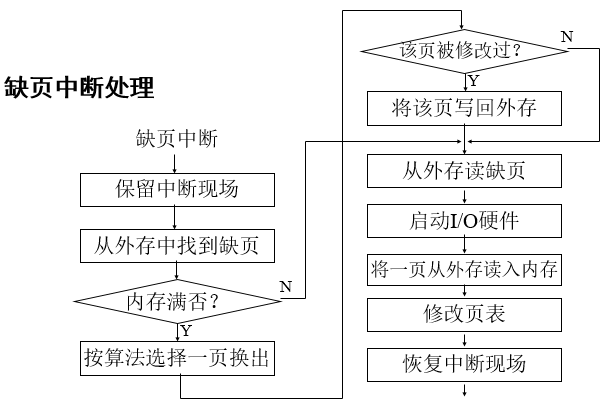
22.虚拟存储与实存的区别

23.局部性原理：程序执行具有局部性特征，即在一较短时间内，程序的执行仅限于某个部分；访问的存储空间也局限于某个区域。

（1）空间局部性 （2）时间局部性

24.虚拟存储器管理策略：调入策略、分配策略、淘汰策略

25.请求分页存储管理：

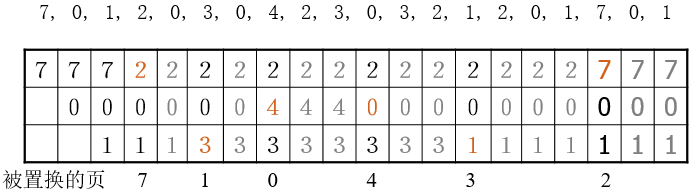


**26.页面置换算法**

**（1）最佳页面置换算法：选择被置换的页面，将是永不使用的页面，或最长时间不使用的页面**

**优点：可保证最低的缺页率**

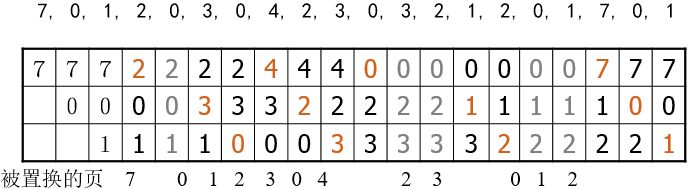
**缺点：不可能真正实现，只可作为其它算法的评价参考**

****

**（2）先进先出（FIFO）页面置换算法**

**选择最先进入内存的页面，即选择在内存中驻留时间最长的页面予以淘汰**

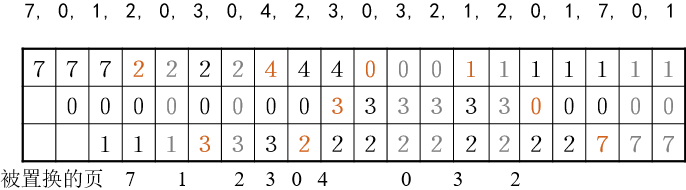
**优点：算法简单 缺点：性能不佳**

****

**（3）最近最久未使用（LRU）页面置换算法：选择最近最久未被使用的页面淘汰**

**优点：性能较好**

**缺点：需要较多的硬件支持**

****

**（4）Clock置换算法**

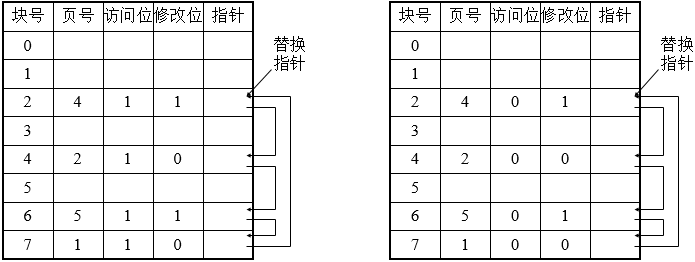
**（5）改进型Clock置换算法**

**①查找指针前进一步，判断当前页是否为第1类页（A=0，M=0），是则选择该页淘汰**

**②否则，继续沿链向下查找**

**③如果沿链查找一周，没有第1类页面，则查找第2类页（A=0，M=1），找到后淘汰，并将访问过的页面访问位置为0（A=0）**

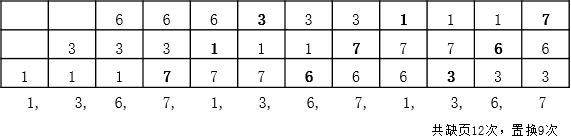
**④如果沿链查找一周后，未找到第2类页面（则所有页面的访问位置已被置为0），重新从第⑴步开始**

****

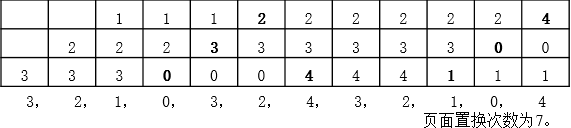
1. **页面缓冲（PBA）置换算法**

**练习**

**（1）在一个请求页式存储管理系统中，访问串为1, 3, 6, 7, 1, 3, 6, 7, 1, 3, 6, 7时，试用最近最久未使用（LRU）置换算法，计算当分配给该进程的内存块数为3时，访问过程中发生的页面置换过程及页面置换次数。（假定开始时，物理块中为空）**

****

**（2）在一个请求页式存储管理系统中，进程P共有5页，访问串为3，2，1，0，3，2，4，3，2，1，0，4时，试用CLOCK置换算法，计算当分配给该进程的内存块数为3时，访问过程中发生的页面置换过程及页面置换次数。（假定开始时，物理块中为空）**

****

27.当需要访问的段不在内存时,便产生一缺段中断

与缺页中断的相同点：

⑴.缺段中断同样可能发生在指令执行过程中

⑵.一条指令执行期间，可能发生多次缺段中断

与缺页中断的不同点：

⑴.一条指令只可能存在于一个段中

⑵.一个（组）数据只可能存在于一个段中

**习题**

**1. 分页和分段存储管理相同点和不同点是什么？**

**a.分页和分段都采用离散分配的方式， 且都要通过地址映射机构来实现地址变换，这是它们的共同点；**

**b.对于它们的不同点有三，第一，从功能上看，页是信息的物理单位，分页是为实现离散分配方式，以消减内存的外零头，提高内存的利用率，即满足系统管理的需要，而不是用户的需要；段是信息的逻辑单位，它含有一组其意义相对完整的信息， 目的是为了能更好地满足用户的需要； 第二页的大小固定且由系统确定， 而段的长度却不固定， 决定于用户所编写的程序； 第三分页的作业地址空间是一维的， 而分段的作业地址空间是二维的。**

**2. 某系统采用页式存储器管理,页长为 1K(1024)字,有一作业四个页面 0， 1， 2，3 按顺序装入主存的 2，4，5，7 块中去,当前正在运行该作业。求（1）逻辑地址 2500 所对应的物理地址为多少； （2）物理地址 7218 对应的逻辑地址是多少？(上述地址都是十进制表示)**

**解：**

**（1）页号=逻辑地址/页大小= 2500/1024=2**

**页内地址=逻辑地址 mod 页大小=2500 mod 1024=452**

**查页表 页 号 2 对应块号 5**

**物理地址=块号 \* 页大小 +页内地址= 5\*1024+452 =5572 （十六进制:15C4）**

**（2）物理块号=物理地址/页大小 = 7218/1024 = 7**

**页内地址 =物理地址 mod 页大小= 7218 mod 1024 =50**

**查页表块号 7 对应的页号为 3**

**逻辑地址=页号\*页大小+页内地址=3\*1024+50 = 3122 （十六进制：C32）**

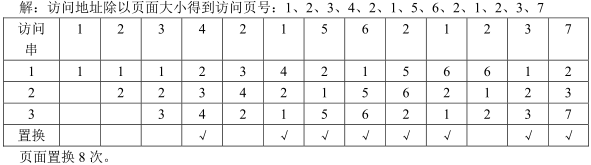
**3. 在一个请求页式存储管理系统中，页面大小为 100B，一个程序的访问地址序**

**列为：115、218、320、402、246、102、521、632、260、167、280、311、**

**720，若系统采用最近最久未使用（LRU）置换算法，计算当分配给该进程的**

**内存块数为 3 时，访问过程中发生的页面置换过程及页面置换次数。 （假定开**

**始时，物理块中为空）**

****

第五章

1.设备管理的必要性（为什么管理？）

设备种类繁多，特性（速率、传送单位、容许操作）各异

设备驱动、控制、分配复杂，用户难以胜任

设备速率与CPU速率相差很大，如何提高效率

2.设备管理的目标：方便性、并行性、均衡性、无关性

3.设备管理的功能：

监视所有设备的状态

制定设备分配策略，实现设备合理分配和回收

设备的驱动和处理

数据缓冲

4.I/O设备分类

（1）按传输速率分类：低速设备、中速设备、高速设备

（2）按信息交换的单位分类

块设备：信息的存取以数据块为单位，如磁盘

字符设备：用于无结构的数据I/O，如键盘、打印机

（3）按设备的共享属性分类：独占设备、共享设备、虚拟设备

（4）按设备的外部特征分类：存储设备、输入设备、输出设备

5.设备与控制器之间的接口：数据信号线、控制信号线、状态信号线

6.设备控制器的功能

接收和识别命令

标识和报告设备的状态

数据交换

地址识别

数据缓冲

差错控制

7.I/O通道

（1）单通路I/O通道：设备与计算机之间只有一条通路

（2）多通路I/O通道：设备与计算机之间有多条通路

8.I/O控制方式

（1）程序I/O控制方式

由CPU采用循环查询的方式直接控制I/O

（2）中断方式

CPU启动I/O后，继续其它工作，等I/O设备数据准备好后，再将数据读入/写出

（3）DMA方式

CPU启动I/O后，由DMA设备控制器自动完成一块（连续）数据的读写工作，CPU可以继续其它工作，等I/O设备完成数据块读/写工作后，再作后续处理

（4）I/O通道方式

CPU启动I/O后，由I/O通道设备控制器自动完成一组数据块（离散）数据的读写工作，CPU可以继续其它工作，等I/O设备完成一组数据块读/写工作后，再作后续处理

9.引入缓冲的原因

* 缓和CPU与I/O设备间速度不匹配的矛盾
* 减少对CPU的中断频率，放宽对中断响应时间的限制
* 提高CPU与I/O设备之间的并行性

10.缓冲管理的方法：单缓冲、双缓冲、循环缓冲、缓冲池

11. I/O软件的设计目标和原则

* 与具体设备无关
* 统一命名
* 错误处理
* 缓冲技术
* 设备分配与释放
* I/O控制方式

12.中断处理程序

* 唤醒被阻塞的驱动程序
* 保护被中断进程CPU环境
* 转入相应的设备处理程序
* 中断处理
* 恢复被中断进程现场

13.设备分配：独享设备、共享设备、虚拟设备

分配方式：静态分配、动态分配

分配算法：先来先服务、优先级高者优先

安全分配方式：*进程发出I/O申请后，即进入阻塞状态，直到I/O操作完成*

优点：不会引起死锁 缺点：进程执行缓慢

不安全分配方式：*进程发出I/O申请后，继续运行；如果还需要I/O设备，再提出申请*

优点：进程执行较快 缺点：可能造成死锁（先进行安全性分析）

14.SPOOLING技术：脱机操作

特点：

* 提高I/O速度
* 将独占设备改造为共享设备（如打印机）
* 实现了虚拟设备功能

15.磁盘访问时间

（1）寻道时间Ts=m\*n+s （将磁头位移至指定磁道经历的时间）

s：磁盘启动时间

m: 递进一个磁道时间（约0.2ms，高速<0.1ms） n: 当前磁道与指定磁道距离

（2）旋转延迟时间Tr = 1/(2r) （将磁头位移至指定扇区经历的时间）

对于硬盘，典型转速为90/s，Tr=5.55ms

对于软盘，典型转速为10/s，Tr=50ms

（3）传输时间Tt = b/(rN) （将数据读出/写入磁盘经历的时间）

r: 磁盘旋转速度

N: 单磁道上的字节数

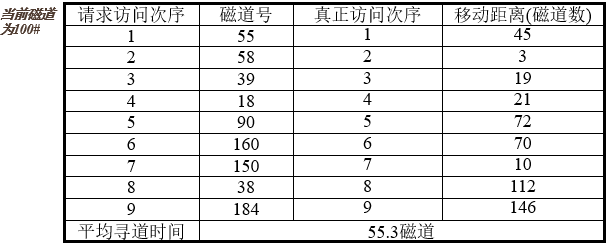
b: 读出/写入的字节数

磁盘访问时间Ta= Ts + Tr + Tt

**16.磁盘调度算法**

**（1）先来先服务（FCFS）**

**依据进程请求访问磁盘的先后次序进行调度**

****

**优点：简单，每个请求都能依次得到处理**

**缺点：平均寻道距离较大**

**（2）最短寻道时间优先（SSTF）**

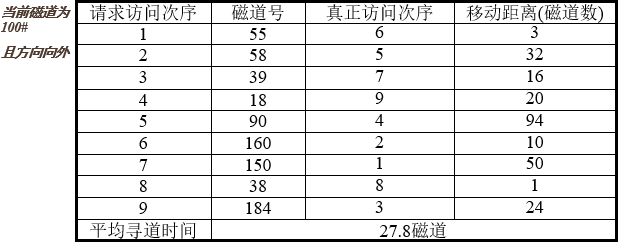
**依据访问磁道与当前磁道最近原则**

****

**优点：平均寻道时间较短**

**缺点：会导致某些进程发生“饥饿”现象，磁头有可能长期停留在同一磁道上(磁臂粘着)**

**（3）扫描调度算法（SCAN）**

****

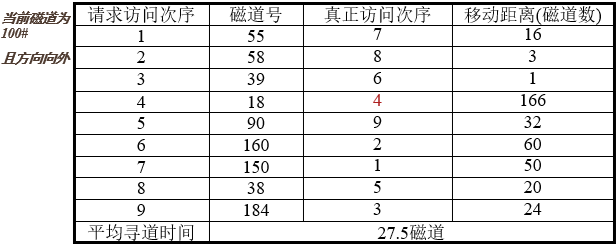
**优点：不会出现进程“饥饿”现象；平均访问寻道时间较短**

**缺点：与磁头近但在磁头运动反方向的磁道等待时间长**

**磁头有可能长期停留在同一磁道上(磁臂粘着)**

**（4）循环扫描调度算法（CSCAN）**

**到达最外磁道后，返回最小磁道开始SCAN算法**

****

**优点：**

**不会出现进程“饥饿”现象**

**平均访问寻道时间较短**

**最长等待时间较SCAN短（一半）**

**缺点：**

**磁头有可能长期停留在同一磁道上(磁臂粘着)**

**（5）FSCAN算法**

* **将磁盘访问请求进程分成两个队列:当前申请队列和扫描期间申请队列**
* **用SCAN或CSCAN算法处理所有当前申请队列中进程对磁盘的访问请求**
* **将扫描期间申请队列中的进程,移入当前申请队列**

**（6）N-Step-SCAN算法**

1. **将磁盘访问请求进程分成若干个长度为N的子队列**
2. **依次按FCFS算法处理这些子队列**
3. **每个子队列中的进程请求,则按SCAN或CSCAN算法进行调度处理**

* **当N=1,则算法退化为FCFS**
* **当N=∞,则算法退化为SCAN或CSCAN**

练习

假设一活动头磁盘有200个磁道，编号从0-199。当前磁头正在127道服务，并刚刚完成了92道的请求。现有如下访盘请求序列（磁道号）：82，110，132，177，94，56，106，87，150。试给出采用扫描调度（SCAN）算法，磁头移动的顺序和移动总量（总磁道数）。

SCAN，方向向外

|  |  |  |  |
| --- | --- | --- | --- |
| 请求访问次序 | 磁道号 | 真正访问次序 | 移动距离（磁道数） |
| 1 | 82 | 8 | 5 |
| 2 | 110 | 4 | 7 |
| 3 | 132 | 1 | 5 |
| 4 | 117 | 3 | 33 |
| 5 | 94 | 6 | 12 |
| 6 | 56 | 9 | 26 |
| 7 | 106 | 5 | 4 |
| 8 | 87 | 7 | 7 |
| 9 | 150 | 2 | 18 |
| 平均寻道 | 13磁道 | | |

CSCAN,方向向外

|  |  |  |  |
| --- | --- | --- | --- |
| 请求访问次序 | 磁道号 | 真正访问次序 | 移动距离（磁道数） |
| 1 | 82 | 4 | 26 |
| 2 | 110 | 8 | 4 |
| 3 | 132 | 1 | 5 |
| 4 | 117 | 9 | 7 |
| 5 | 94 | 6 | 7 |
| 6 | 56 | 3 | 94 |
| 7 | 106 | 7 | 12 |
| 8 | 87 | 5 | 5 |
| 9 | 150 | 2 | 18 |
| 平均寻道 | 19.7磁道 | | |

**最短寻道时间优先（SSTF）**

|  |  |  |  |
| --- | --- | --- | --- |
| 请求访问次序 | 磁道号 | 真正访问次序 | 移动距离（磁道数） |
| 1 | 82 | 7 | 5 |
| 2 | 110 | 3 | 7 |
| 3 | 132 | 1 | 5 |
| 4 | 117 | 2 | 15 |
| 5 | 94 | 5 | 12 |
| 6 | 56 | 8 | 26 |
| 7 | 106 | 4 | 4 |
| 8 | 87 | 6 | 7 |
| 9 | 150 | 9 | 94 |
| 平均寻道 | 19.4磁道 | | |

**第5章课后习题答案**

1. 假设一活动头磁盘有200个磁道，编号从0-199。当前磁头正在127道服务，并刚刚完成了92道的请求。现有如下访盘请求序列（磁道号）：82，110，132，177，94，56，106，87，150。试给出采用扫描调度（SCAN）算法，磁头移动的顺序和移动总量（总磁道数）。

答：磁头由内往外运动，移动的顺序：

   127 -> 132 -> 150 -> 177 -> 110 -> 106 -> 94 -> 87 -> 82 -> 56

   移动距离：

          5      18     27     67     4      12    7     5     26

   移动总量（总磁道数）：171

|  |  |  |  |
| --- | --- | --- | --- |
| 请求访问次序 | 磁道号 | 真正访问次序 | 移动距离（磁道数） |
| 1 | 82 | 8 | 5 |
| 2 | 110 | 4 | 67 |
| 3 | 132 | 1 | 5 |
| 4 | 177 | 3 | 27 |
| 5 | 94 | 6 | 12 |
| 6 | 56 | 9 | 26 |
| 7 | 106 | 5 | 4 |
| 8 | 87 | 7 | 7 |
| 9 | 150 | 2 | 18 |
| 平均寻道 | 19磁道 | | |
| 总磁道 | 171磁道 | | |

2. 简述什么是虚拟储存器。

答：虚拟存储器是指具有请求调入功能和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统。其逻辑容量由内存容量和外存容量之和所决定，其运行速度接近于内存速度，而每位的成本却又接近于外存。（参见课本4.6小节，只要答对基本要点即可）

第6章

1.文件系统的功能：OS中实施文件管理的机构称为文件管理系统

实施文件存储空间的分配与回收（即磁盘管理）

实现文件名到文件空间的映射

实现用户要求的各种文件操作(如存、取等)

提供文件共享能力以及保护与保密措施

2.外存分配方法

（1）连续外存分配：指为文件分配一组相邻接的盘块

优点：

顺序访问容易

一次大批量数据的访问速度较快

缺点：

要求有连续的外存空间(同样存在“碎片”问题)

必须事先知道文件的长度

（2）隐式链接：将链接指针直接放在盘块数据区中

优点：①不存在“碎片” ②不需要连续外存空间 ③不需要事先知道文件长度

缺点：

①只适合于顺序访问，不适合随机访问(低效)

②可靠性差(中间任何指针出错，整个链断开)

③数据区大小不等于2的次幂（指针占用数据区空间），不利于与内存页对应

（3）显式链接：将用于链接文件各盘块的指针，显式地存放在一张链接表(文件分配表FAT)中

优点：

①FAT占空间小，可以调入内存

②查找速度快，适合随机存取

同样不存在“碎片”，不需要连续外存空间

不需要事先知道文件长度

缺点：

①FAT需要占用外存空间

②FAT需要占用较大内存空间：对于大的硬盘，每次只调取部分FAT进入内存

③不能高效直接存取

（4）索引分配

每个文件先分配一个索引块

当已分配的索引块装不下所有盘块号时，再分配新的索引块

优点：方便直接存取

缺点：需要更多外存空间

混合索引：

直接地址+一级间址+二级间址+三级间址=10\*4KB+1K\*4KB+1K\*1K\*4KB+1K\*1K\*1K\*4KB

3.目录的功能：a实现按名存取 b.提高对目录的检索速度 c.文件共享 d.允许文件重名

4.目录管理的方法

（1）单级目录：优点：简单 缺点：查找速度慢，不允许重名，不便于实现文件共享

（2）两级目录：主文件目录（MFD）→用户文件目录（UFD）

优点：

①提高了检索目录的速度

②在不同的用户目录中，可以同名

③不同用户可共享文件

缺点：

①不同用户之间不能互相访问

②用户不能继续建立自己新的子目录

（3）树形目录结构：现在用的目录结构

具有三级或三级以上的目录结构称为树型结构的目录

树型目录结构中，允许在任何一级子目录中，再创建新的子目录

第5节 文件存储空间的管理

6.空闲表法

空闲盘区的分配算法

（1）首次适应算法

（2）循环首次适应算法

（3）最佳适应算法

（4）最坏适应算法

空闲盘区的回收算法

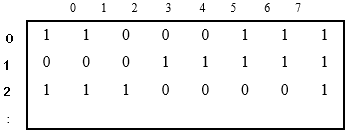
对邻接盘区进行拼接

7.空闲链表法

（1）空闲盘块链表法

（2）空闲盘区链表法

**8.位示图法：用二进制表示盘块使用情况**

****

**变换盘块号（n）**

**n=i\*8+j（找到第i行第j列）**

**修改位示图**

**优点：**

**①位示图占用空间较小，可以调入内存**

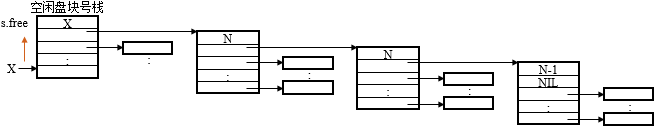
**②分配和回收操作简单、有效**

**9.成组链接法**

**（1）盘块的分配**

**①若s.free=1，则将栈底盘块号所对应的盘块（内容）调入空闲盘块号栈，并将该盘块分配出去**

**②否则，直接将栈顶盘块号所对应的盘块分配出去，并将s.free递减1**



（2）**盘块的回收**

**①若s.free=N，则将空闲盘块栈内容写入新释放的盘块中，并使s.free=1，且将该盘块号作为栈底**

**②否则，s.free递增1，将该盘块号作为栈顶**

**成组链接法举例**

**某一操作系统采用成组链接法管理磁盘空闲空间。为简单起见，假定3块一组，并在某一时刻，空闲盘块号栈的内容从S.free开始依次为2，6，3，第6块中的管理信息为3，1，2，4。问：**

**分配3块，哪3块被分配出去？空闲盘块号栈的内容（从S.free开始）是什么？**

**回收第7，8块，空闲盘块号栈的内容（从S.free开始）是什么？如此时有空闲盘块的内容改变，是哪块？变为什么？**

**解：**

**成组链接法举例[分配3块]**

**先分配第3块，空闲盘块号栈的内容从S.free为：1，6**

**再分配第6块时，先将第6块的内容3，1，2，4调入空闲盘块号栈中，然后将第6块分配出去；**

**再分配第4块，空闲盘块号栈的内容从S.free为：2，1，2。**

**因此，分配3块，依次是第3、6、4块被分配出去，最后空闲盘块号栈的内容从S.free为：2，1，2。**

**成组链接法举例[回收第7、8块]**

**先回收第7块，空闲盘块号栈的内容从S.free为：3，1，2，7**

**再回收第8块时，先将空闲盘块号栈的内容写入第8块中，然后将第8块放在栈底。**

**因此，回收第7、8块，空闲盘块号栈的内容从S.free为：1，8；第8空闲盘块的内容被改变，其内容为3，1，2，7。**

10.文件共享

（1）绕弯路文件共享方式

（2）基于索引节点的共享方式

（3）利用符号链实现文件共享 例如http://www.szu.edu.cn/szu2007.asp

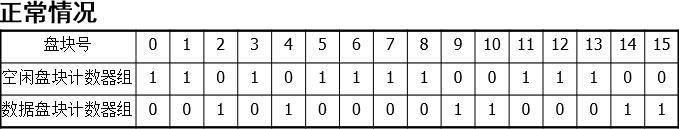
11.磁盘容错技术目标：

磁盘某部分出现缺陷或故障时，

（1）磁盘仍能正常工作

（2）不致造成数据的错误和丢失

12.磁盘块号的一致性









4、某一操作系统采用成组链接法管理磁盘空闲空间。为简单起见，假定5块一组，并在某一时刻，空闲盘块号栈的内容从S.free开始依次为3，5，3，2。问：  
⑴、回收第7，8，9块，空闲盘块号栈的内容（从S.free开始）是什么？如此时有空闲盘块的内容改变，是哪块？变为什么？  
⑵、分配2块，哪两块被分配出去？空闲盘块号栈的内容（从S.free开始）是什么？  
  
解：空闲盘块号栈的内容从S.free开始依次为3，5，3，2。  
⑴、回收第7块，空闲盘块号栈的内容从S.free开始依次为4，5，3，2，7  
回收第8块，空闲盘块号栈的内容从S.free开始依次为5，5，3，2，7，8  
回收第9块时，由于空间盘块号栈已满（规定5块为一组），因此，先将空闲盘块号栈的内容5，5，3，2，7，8写入第9块，再将第9块作为栈底。因此：  
回收第7，8，9块，空闲盘块号栈的内容（从S.free开始）为：1，9；第9空闲盘块的内容发生了改变，变为5，5，3，2，7，8

⑵、分配2块，分别是第9块和第8块被依次分配出去，空闲盘块号栈的内容（从S.free开始）为：4，5，3，2，7