|  |
| --- |
| 北京理工大学 |
| 操作系统期末 复习总结 |

|  |
| --- |
| WZQ  2021-9-22 |

目录

[第1章 操作系统概论 2](#_Toc90495656)

[第2章 进程管理 5](#_Toc90495657)

[第3章 进程之间的并发控制和死锁 11](#_Toc90495658)

[第4章 存储器管理 20](#_Toc90495659)

[第5章 文件系统 30](#_Toc90495660)

# 第1章 操作系统概论

1. 计算机系统从功能上分为四个层次：**硬件、操作系统、实用程序和应用程序**，这四者是一种**单项调用**关系。
2. 操作系统是整个计算机系统的管理和控制中心。从设计者角度看，它是**计算机软硬件（数据和程序）资源的管理和控制程序**；从用户角度看，**它是用户与计算机系统之间的一个交口界面**。其设计目标有三个：**1）方便性：使用户方便、简单地使用计算机系统；2）高效性：时间算系统高效且可靠地运转；3）便于操作系统的设计和维护。**
3. 程序，又称为作业，包括：**编写源程序、调用编译源程序、调用连接装配程序、装入可执行程序和运行时的数据**。上述为一个相对独立的步骤叫做**作业步**，其之间高度依赖，**逻辑上顺序执行**。问题：**大量时间用编排程序运行上了**。
4. 简单批处理系统中心思想：使用一个**监控程序软件**，操作员只需将作业顺序放入输入设备即可（单通道批处理）。这样的优点有两个：**1）不存在空闲的机器时间；2）由监控程序处理作业编排问题**。
5. 联机批处理，问题：主机速度降低为慢速的输入或输出设备；脱机批处理，增加**卫星机**专门控制输入/输出。
6. 由于硬件的发展，产生了**硬件通道、中断和缓冲技术**，转向以**主存**为中心。那么**主存中同时存放多个作业**，实质同时处于运行状态的设计方法称为多道程序设计。特点：**宏观上并行，微观串行（只有一个处理机运行）。目的：提高CPU的利用率**。缺点：用户与作业无法交互，作业平均周转时间大。
7. 几个重要概念：

* **资源利用率**：指单位时间实际使用时间或使用容量所占的比率。
* **吞吐量**：一小时内系统所处理的任务个数。
* **周转时间**：从作业进入系统号作业退出系统所经历时间。

1. 分时系统：将CPU的单位时间划分若干时间，每个时间段成为一个时间片，并轮流给各联机用户使用。其特点为：**同时性、独立性、交互性、及时性（通读交集）。**
2. 批处理系统+分时系统=作业处理系统。前者适合较为成熟的、需要消耗大量时间的大型作业、后者适合处理小作业。
3. 实时系统：为外部事件能做出及时相应，它是一个专用系统。特点：**实时性、可靠性、确定性（确实可）。**

11、操作系统功能：

1. **处理机管理**：即进程管理。进程控制、调度、通信。
2. **存储器管理**
3. **设备管理**
4. **文件管理**

12、操作系统特性：**并发性、共享性、虚拟性、异步性**

13、网络操作系统分为：客户/服务器模式、对等模式

14、分布式操作系统没有制定标准协议，其有一个同一的操作系统。机群(cluster)是一种分布式系统。机群节点机私有OS。有**一层软件**运行在节点机之上**。**

15、**虚拟化是实现云计算的基础技术**。Type1、Type2模式，后者（宿主）经过路径更长，性能损失更严重。

16、两种接口：**操作接口**（**命令语言、窗口界面**是用户使用计算机系统的主要接口）、**编程接口**（**系统调用**是用户与操作系统之间的编程接口）。

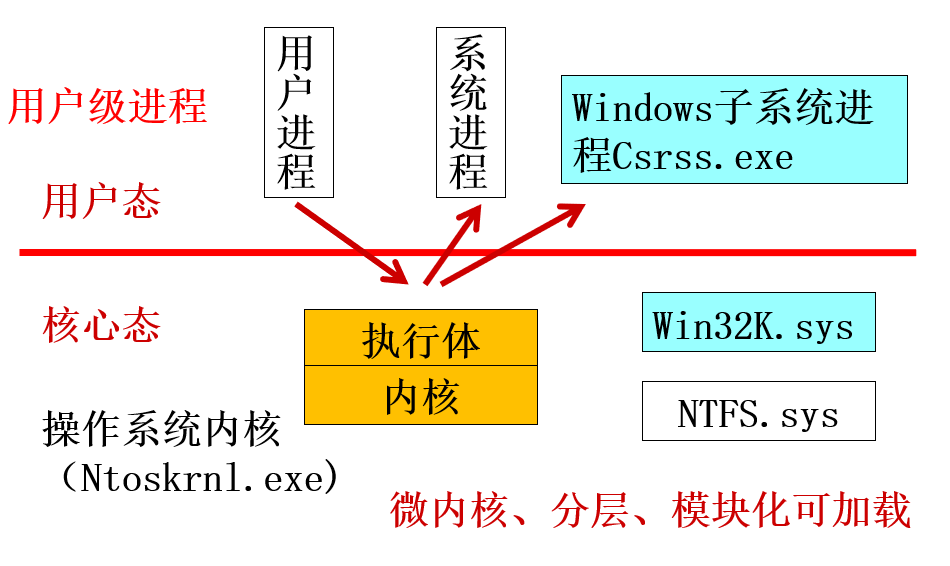
17、在计算机中存在两类性质不同的程序：**操作系统内核程序；用户程序**。操作系统程序运行在**核心态**，允许处理机全部指令，访问所有存储器和寄存器；用户程序运行在**用户态**，只允许执行处理机非特权指令，访问特定寄存器和存储区。

18、**用户级进程无法直接交互，要通过操作新系统内核进行消息传递。**

19、用户态切换到内核态主要发生三种过程：**中断、异常（陷入）和系统调用**。**，**CPU能从**用户程序**的运行转入**操作系统内核程序**的运行**。**中断是执行完当前指令后再去响应中断，是**异步**发生的。异常是**同步**发生的，直接崩掉。中断or陷入后，相应程序要借用用户进程的**核心栈**，在用户进程进行执行。

20、执行体 和 内核**分层**设计，方便进行扩展。微内核：**用户级进程在用户态执行。**

重点：



# 第2章 进程管理

1、操作系统中最核心的概念是**进程**。程序顺序执行特点：**封闭性、可再现性**。

2、程序并发执行特征：

* **失去封闭性和可再现性**：多个程序**共享**系统资源，因而**资源状态**由多个程序改变
* **程序间产生相互制约关系**：**间接制约关系、直接制约关系**
* **程序与CPU活动之间不再一一对应**。程序——静态；活动——动态；

3、进程具有特性：**（病毒冻结）**

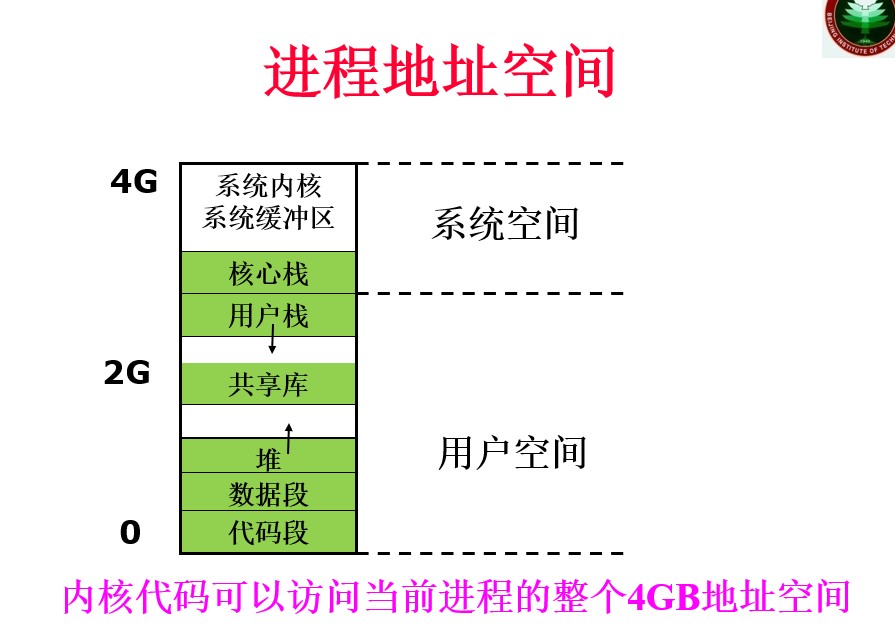
* **动态性：**进程是程序的一次执行过程，是临时的，有生命期的。
* **独立性：**进程是系统进行资源分配和调度的一个独立单位。
* **并发性：**多个进程可在处理机上交替执行。
* **结构性：**系统为每个进程建立一个**进程控制块。**进程由**程序、数据和进程控制块**三部分组成**。**

**4、必背：**

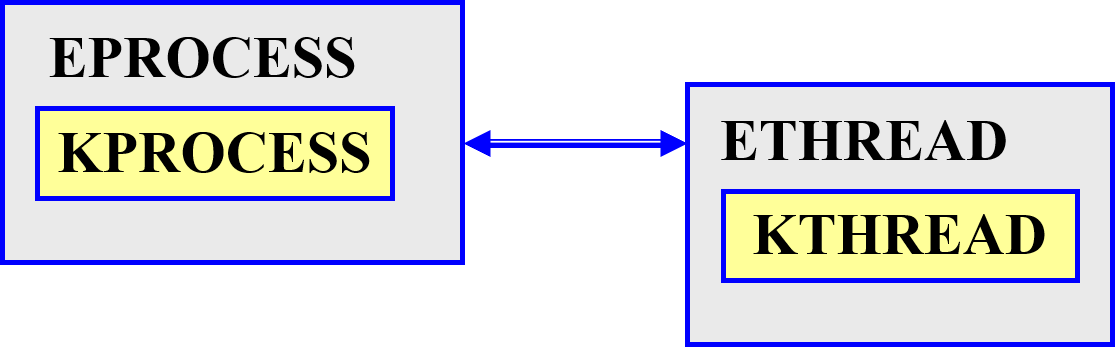
1. **进程是动态的，程序是静态的。程序是有序代码的集合，进程是程序的执行，没有程序就没有进程。通常，进程不可以在计算机之间迁移，而程序可以复制。**
2. **进程是暂时的，程序是永久的。**
3. **进程包括程序、数据和进程控制块。**
4. **通过多次执行，一个程序可对应多个进程；通过调用关系，一个进程可包括多个程序。进程可创建其他进程，而程序不能形成新的程序。**

5、进程控制块 (PCB，Process Control Block)，也叫进程描述符(PD，Process Descriptor)，它是进程存在的**唯一标识**。

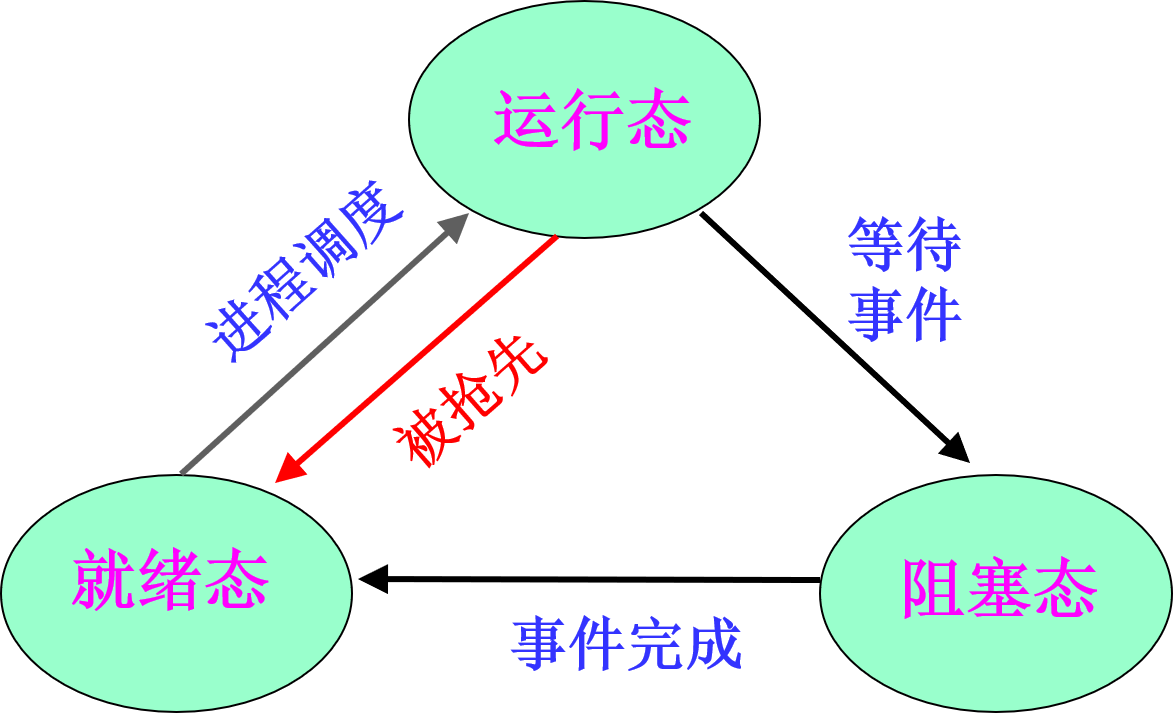
6、程序计数器：记录当前进程中进行第几条指令；**类UNIX由父/子进程关系，windows进程间没有家族关系。**

**7、必背：**

8、Windows是**创建进程到主线程**，有主线程调用程序，并在其中观察是否创建其他线程，如下：



9、进程状态最重要分为：**运行态、阻塞态、就绪态**



被抢先：优先级、分时切片

等待事件：同步发消息、等待输入输出

事件完成：可能提高事件优先级

10、用户态中：主函数调用子函数——是一种过程调用，运行在用户栈中。

11、进程控制是指：**系统使用一些具有特定功能的程序段来创建、撤销进程以及完成进程各个状态间转换的一系列有效管理。这些程序段是系统内核实现的。**

12、创建原语的功能：

找到空闲PCB，分配主存并调入主存中，把调用者提供的参数填入PCB中，插入就绪队列。

13、fork()创建主进程副本，execute()选择执行选用程序。主进程中,fork()返回值大于0；子进程中,fork()返回值等于0.

14、阻塞原语为进程自己阻塞自己，是主动的过程，使自己从运行态变成阻塞态。唤醒原语为从阻塞态到就绪态，一般为等待I/O输入完成，或等待信号发送。

15、处理机调度的高级调度为作业调度，低级调度为进程调度。进程调度的**目标是使一些进程并发执行，以最大化CPU的利用率**。

16、进程调度的功能：

(1)**管理系统中各进程的执行状况**：管理进程控制块，将进程的状态变化及资源需求情况及时地记录到PCB中。

(2)**选择就绪进程占有CPU**

(3)**进行进程上下文的切换**：将正在执行进程的上下文保存在该进程的PCB中，将刚选中进程的运行现场恢复起来，以便执行。

17、进程上下文是进程的**运行环境和物理实体**，它有三部分组成：**用户级、寄存器级和系统级的上下文**构成。

**18、进程的调度方式;**

①非抢先方式(非剥夺方式):某一进程占用CPU,直到运行完或不能运行为止，其间不被剥夺。用在**批处理系统**。主要优点：简单、系统开销小。

②抢先方式(剥夺方式):调度程序基于某种策略（优先级、时间片等）剥夺现运行进程的CPU给其它进程。用在**分时系统、实时系统**。

19、进程调度算法（区分**等待时间**和**周转时间**）：

（1）先来先服务(FCFS)

缺点：容易被大作业垄断。

（2）最短作业优先(SJF)

对短作业有利，作业的平均周转时间最佳。容易出现**饿死**现象。

（3）响应比高者优先(HRN)**会计算！**

Rp =(作业等待时间+作业估计运行时间)/作业估计运行时间

**（4）优先级调度法**

将CPU分配给就绪队列中**优先级最高**的进程

**静态优先级**：在进程创建时确定的，运行时保持不变。通常赋予**系统进程**较高优先级；赋予**申请资源量少**的进程较高优先级。

**动态优先级**：原优先级可随进程的推进而改变。根据进程**占用CPU或等待CPU时间的长短**动态调整

**（5）轮转法(Round Robin)**

用在**分时系统**，轮流调度所有就绪进程。

利用一个**定时时钟**，使之定时地发出中断。时钟中断处理程序在设置新的时钟常量后，立即转入进程调度程序。

**（6）多级反馈队列轮转法**

优先级不同的进程排在不同的就绪队列。

**刚创建的进程**和**因请求I/O而未用完时间片的**进程排在高优先级队列。

**高优先级进程的时间片较短**，低则较长。

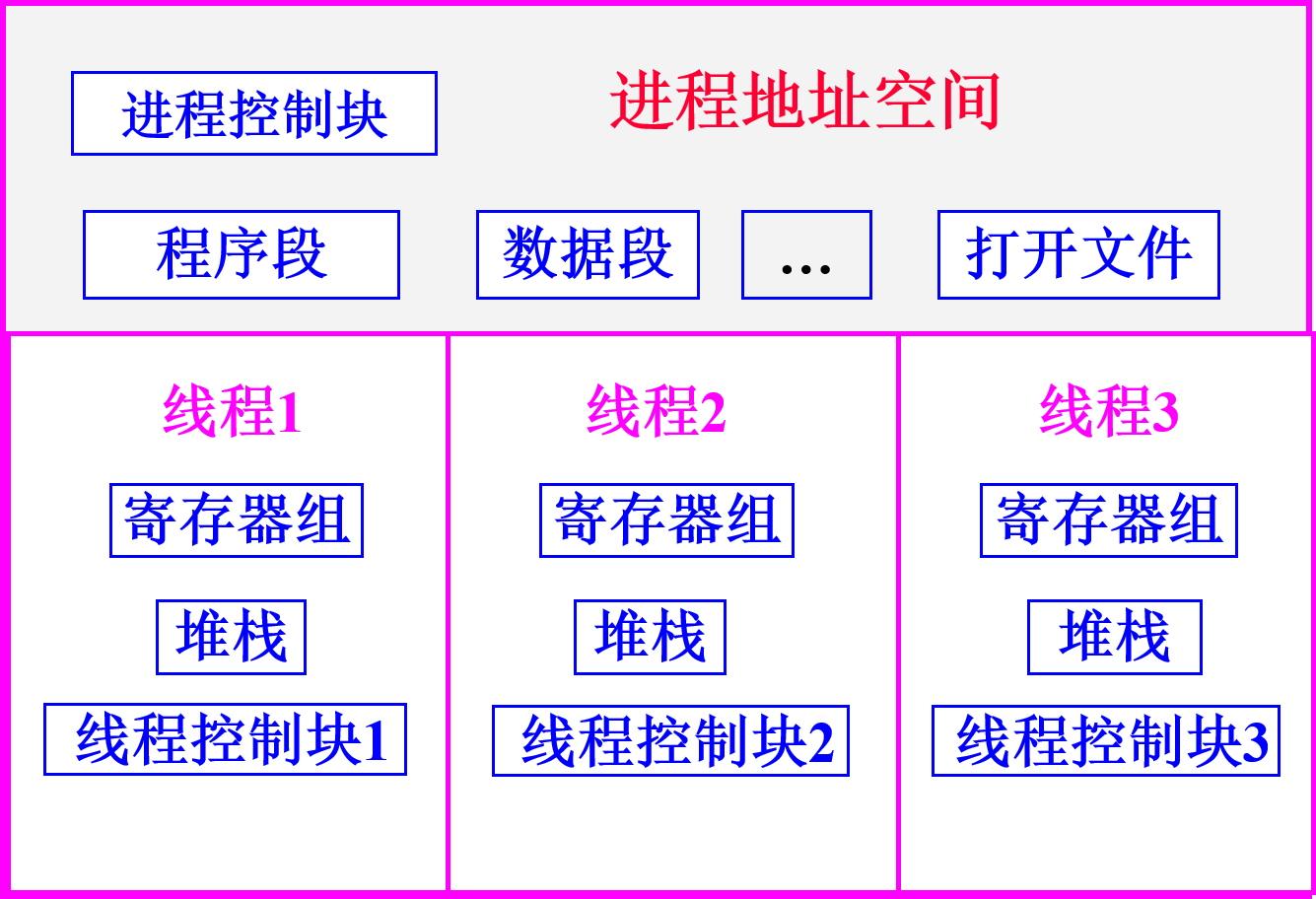
运行2~3个时间片还未完成的进程**降级**。

多个就绪队列可采用**前后台**运行。前台队列采用RR调度；后台采用FCFS。

20、实时系统调度算法（理解就成）

**时钟驱动法+加权轮转法**

21、以进程为单位分配资源，以线程为单位调度执行（CPU）。线程间通过**数据段**传送数据，不通过操作系统。以线程控制块进行线程的描述。



**22、线程与进程的比较（必考）**

**（1）拥有的资源**

* **进程拥有一个独立的地址空间，用来存放若干代码段和数据段。拥有打开的文件，以及至少一个线程。**
* **一个进程内的多线程共享该进程的所有资源，线程自己拥有很少资源。**

**（2）调度**

* **进程调度需进行进程上下文的切换，开销大。**
* **同一进程内的线程切换，仅把线程拥有的一小部分资源变换了即可，效率高。同一进程内的线程切换比进程切换快得多。不同进程的线程切换则会引起进程上下文切换。**

**（3）并发性**

* **引入线程后，使得系统的并发执行程度更高。进程之间、进程内的多线程之间可并发执行。**

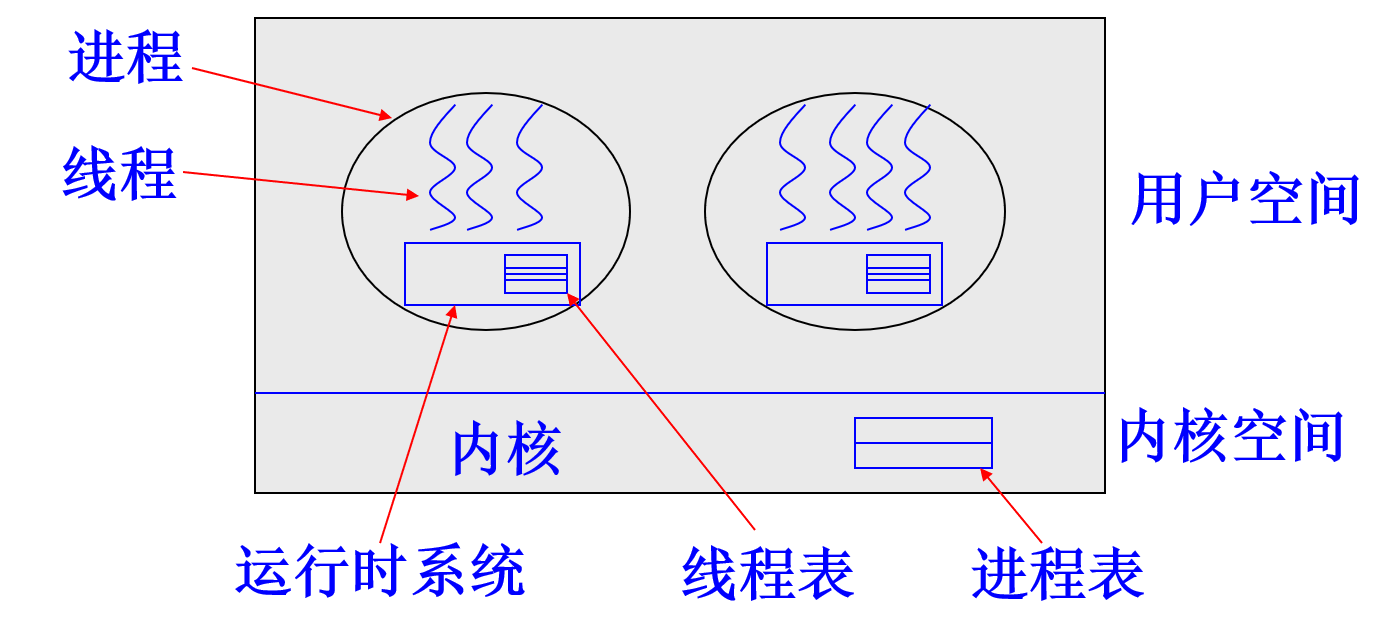
**（4）安全性**

* **同一进程的多线程共享进程的所有资源，一个线程可以改变另一个线程的数据导致错误发生，共享数据方便，但若出错则会影响这个进程内数据安全。多进程实现相比于多线程实现，安全性更好。**

23、系统对线程的支持：

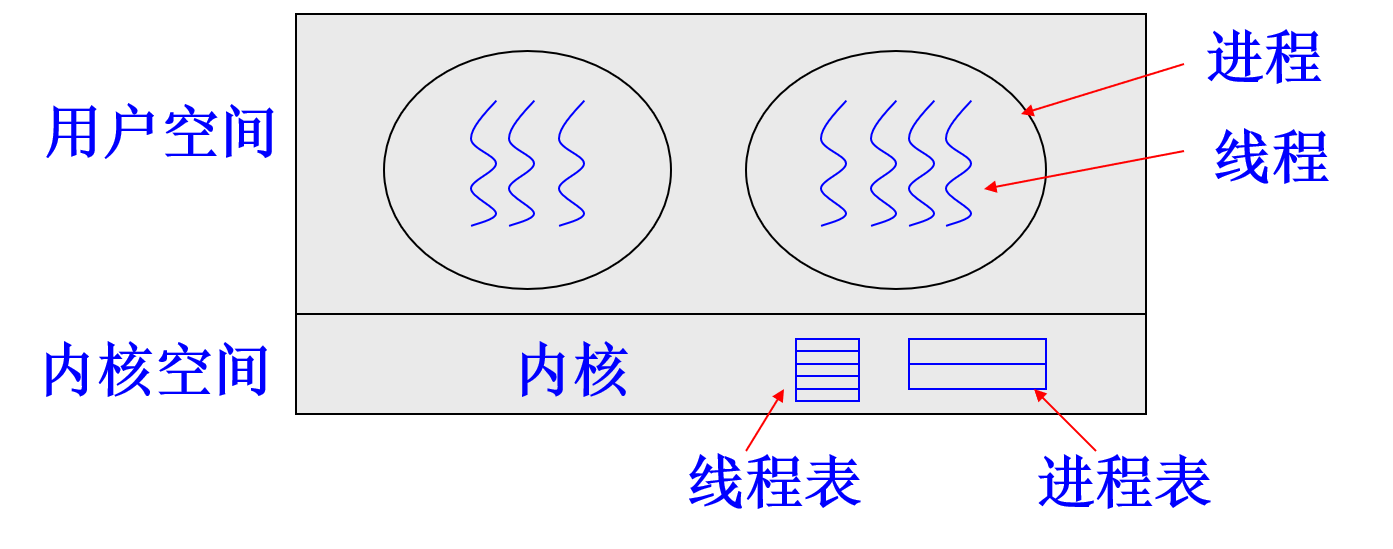
**（1）用户级线程**

**有关线程的所有管理工作都由用户程序通过调用在用户态运行的线程库完成。系统内核并不知道线程的存在（透明）。系统内核以进程为单位进行调度。一个线程阻塞，其依附的进程也阻塞。**



**（2）核心级进程**

**有关线程的管理工作都由内核完成。应用程序通过系统调用来创建或撤销线程。一个线程的阻塞，不影响其他线程的执行。**

****

**（3）两级组合**

**多个用户级的线程对应等量或少量的核心级的线程。当一个线程阻塞时，核心可以调度同一进程或不同进程的另一个线程执行，使得线程在宏观和微观上都能很好地并行。**

# 第3章 进程之间的并发控制和死锁

1、并发进程的特点：**互斥关系、同步关系前序关系**。其中，互斥关系是间接制约关系，因为共享资源而使两个独立进程产生关系。同步关系是直接制约关系，是一组协作进程为完成同一任务而引发的。

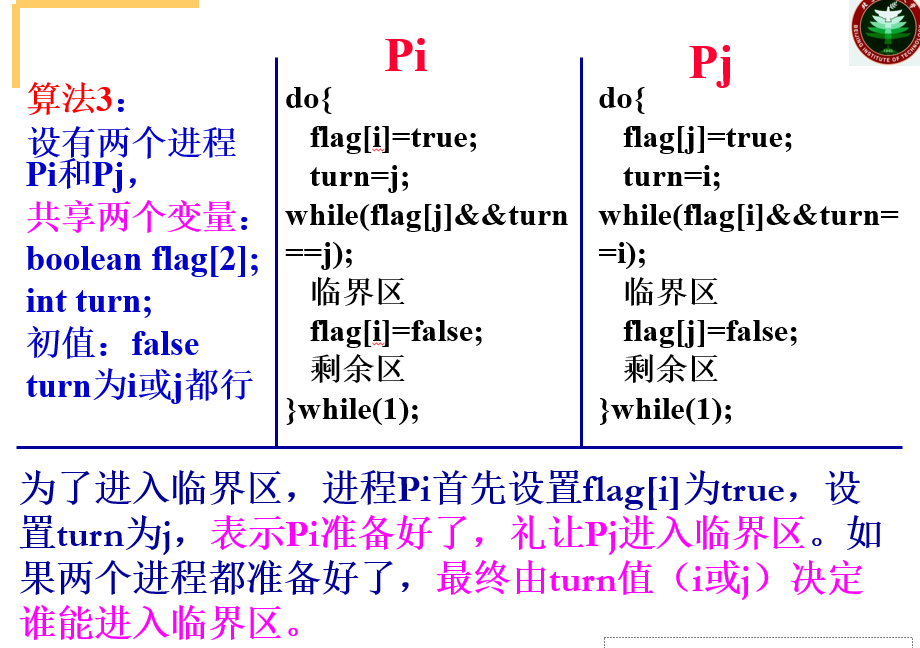
2、进程之间的低级通信：**通过信号量实现进程之间的互斥和同步关系。**进程间通信（IPC）。

3、进程之间的互斥：（**必考考点！！！**）

* **共享资源：①慢速的硬设备，如打印机；②软件资源，如共享变量、共享文件和各种队列等。**
* **临界资源：一次仅允许一个进程使用的系统中共享资源。**
* **临界区(critical section)：并发进程访问临界资源的那段必须互斥执行的程序。**

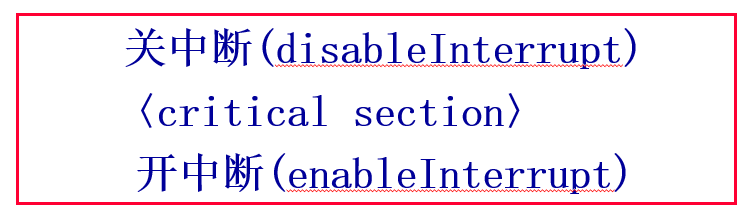
4、并发进程临界区四个准则：**互斥使用；让权等待；有空让进；有限等待。**

5、软件实现互斥：



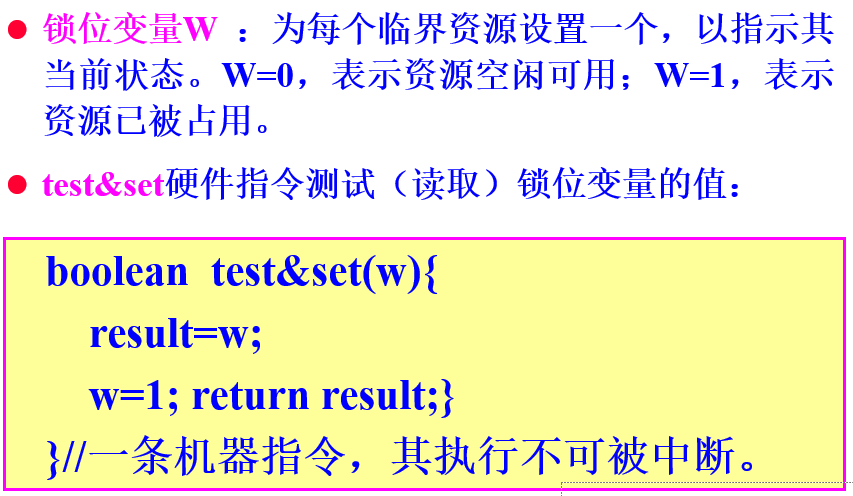
但是，共享变量需要进程间进行通信，这就需要系统调用完成，显然不现实。那么这种技术就只能在**多线程**中进行使用。

6、关中断实现互斥：



原理简单，但是限制了处理机**交叉**处理的能力。若是**多处理机系统**，仅禁止一个CPU而不阻碍其他CPU是没有任何用的。同时把这个权力交给用户是不明智的（**有可能关上了不打开**）。

7、使用测试和设置硬件指令：



等待时需要用while循环一直消耗计算机cpu资源，这叫做“**忙等待**”。

8、信号量与P、V操作（**必考！！！**）

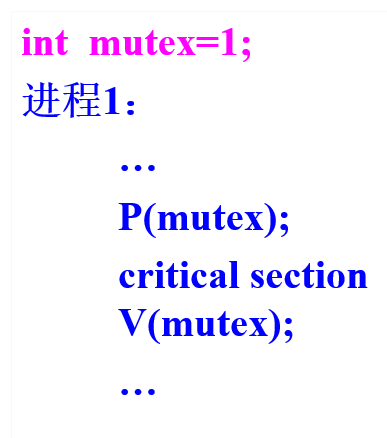
信号量用一个数据结构描述：



对于信号量只允许执行原语操作，其在系统中定义，用户自己不定义。

* 对于同步来说：“P在等待，V在发送”；
* 对于互斥来说：“P在申请，V在释放”；

9、利用信号量实现进程间互斥：



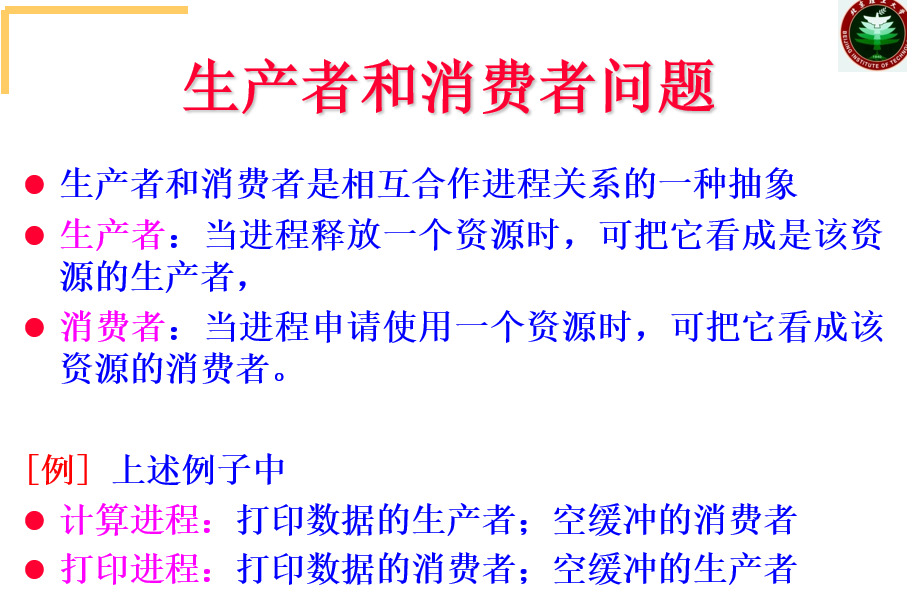
* 注意定位和赋初值的位置（写在所有进程前面！！）
* 定义时默认int类型，好赋初值。
* 取值范围：**+1～-(n-1).**

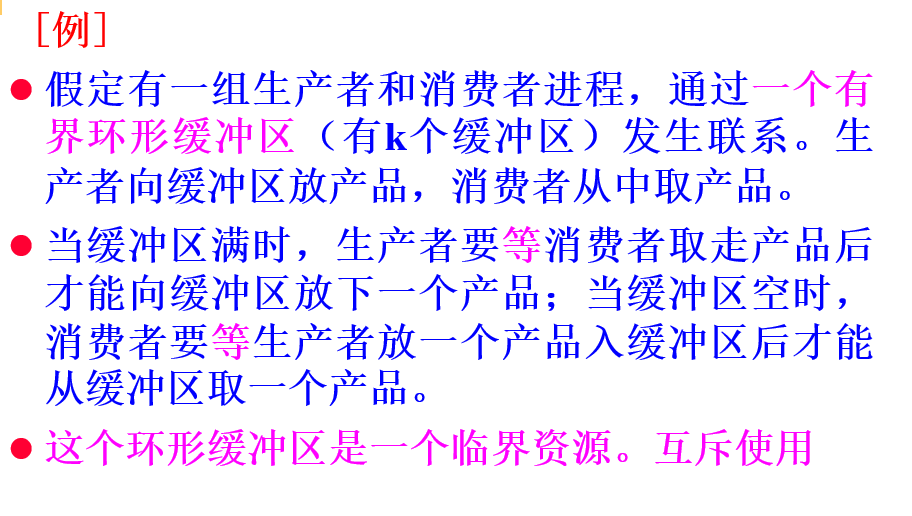
10、利用信号量实现进程间同步

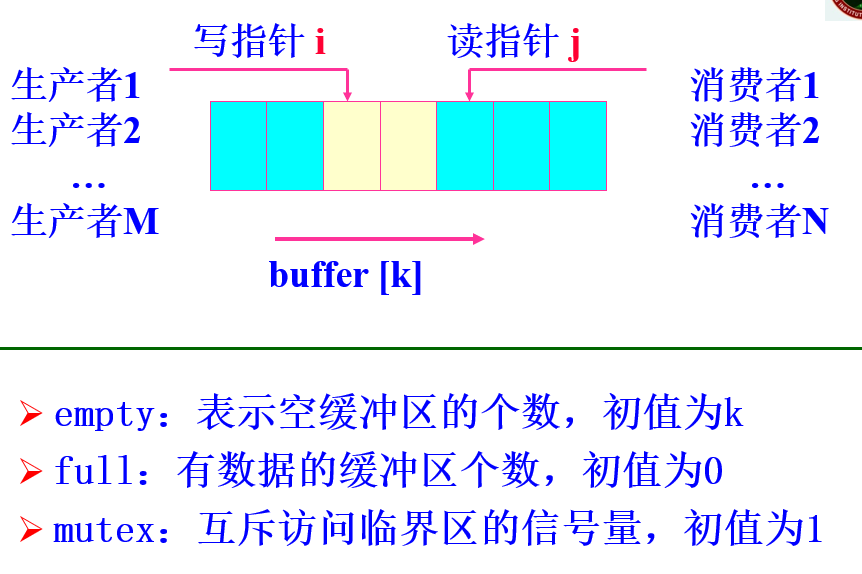
对于计算器+打印机问题，注意不能用一个同步信号量，否则有可能还没来得及写数据的时候，时间片到了，造成读操作出现bug。

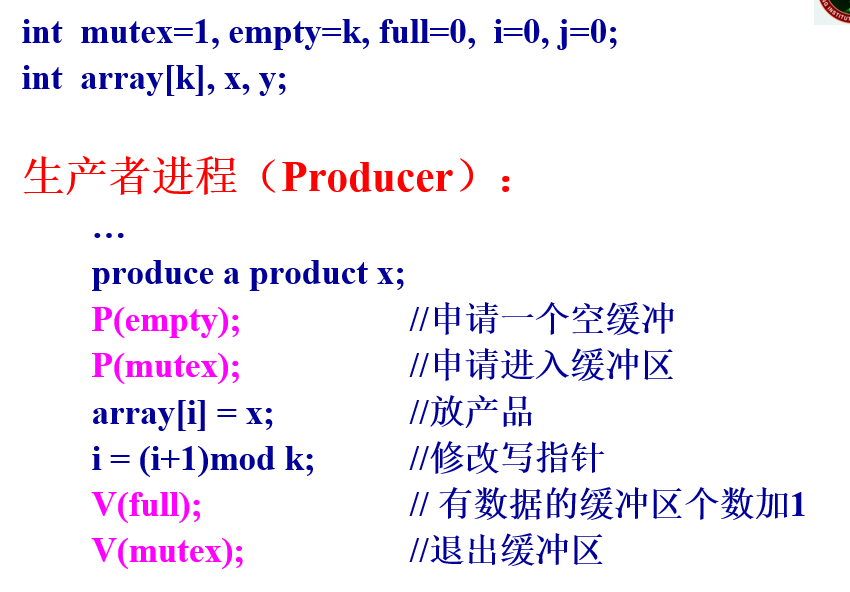
【**以下重点模型，牢记，考试时套用**】

**11、生产者和消费者问题**

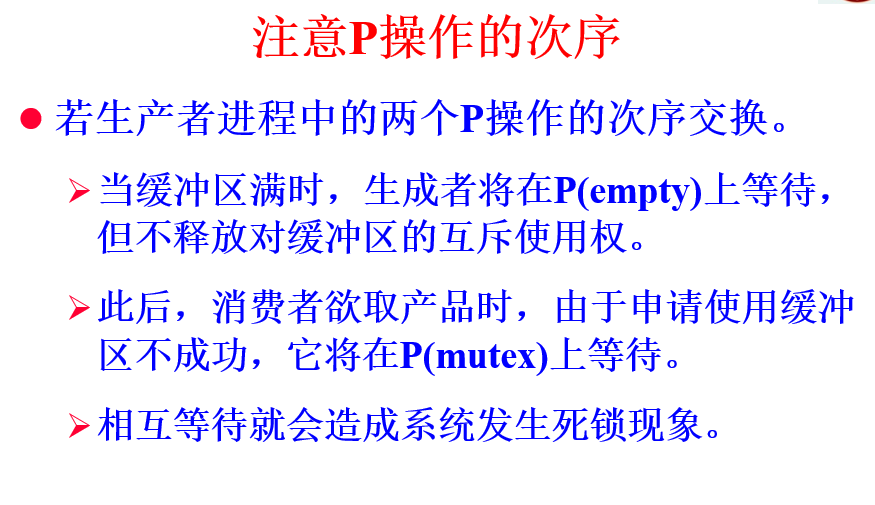


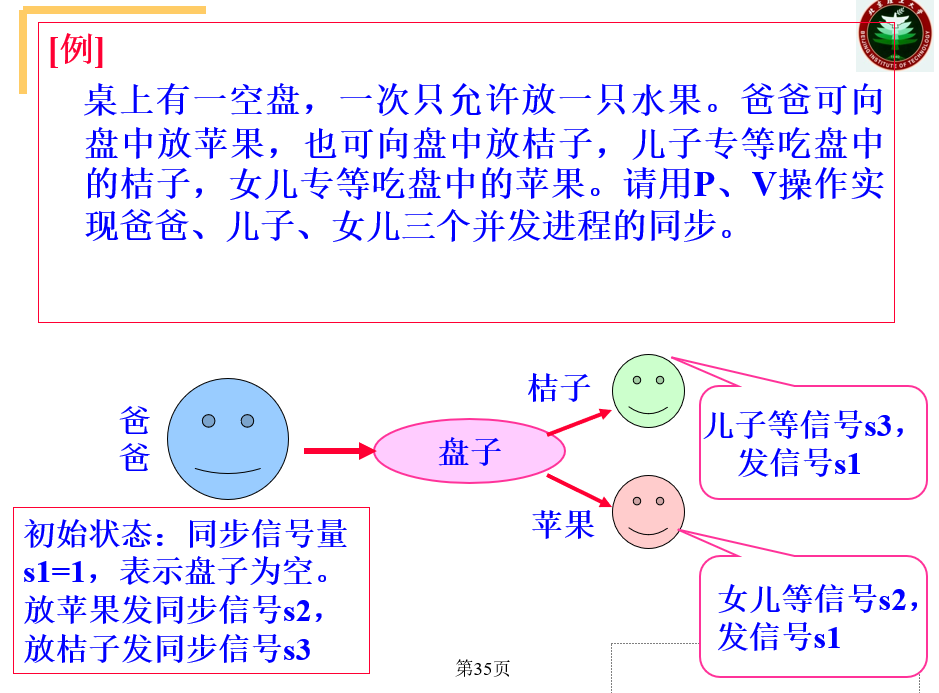


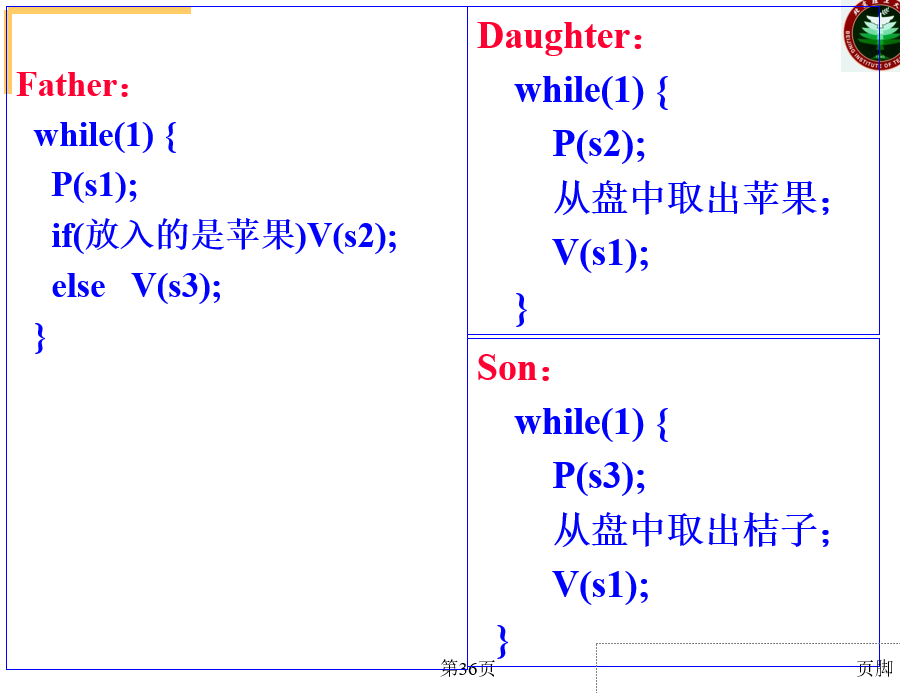






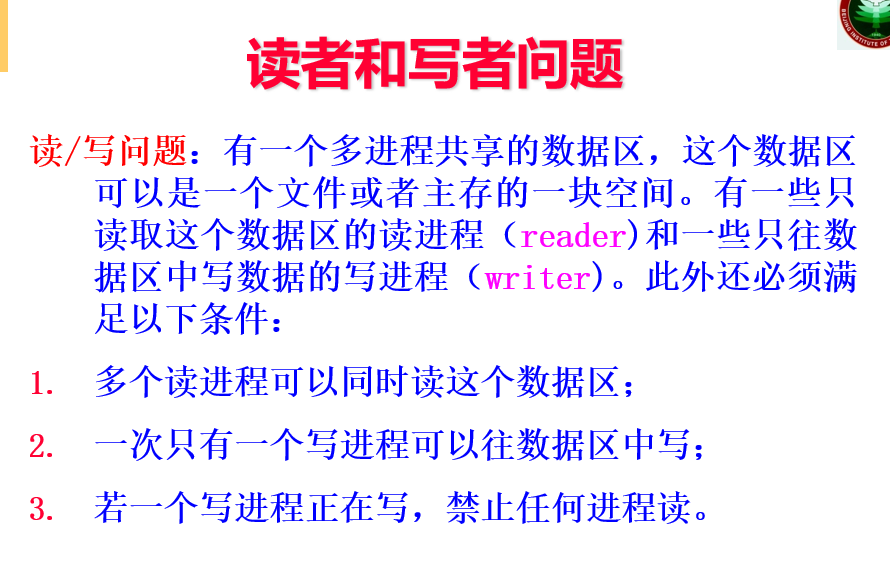


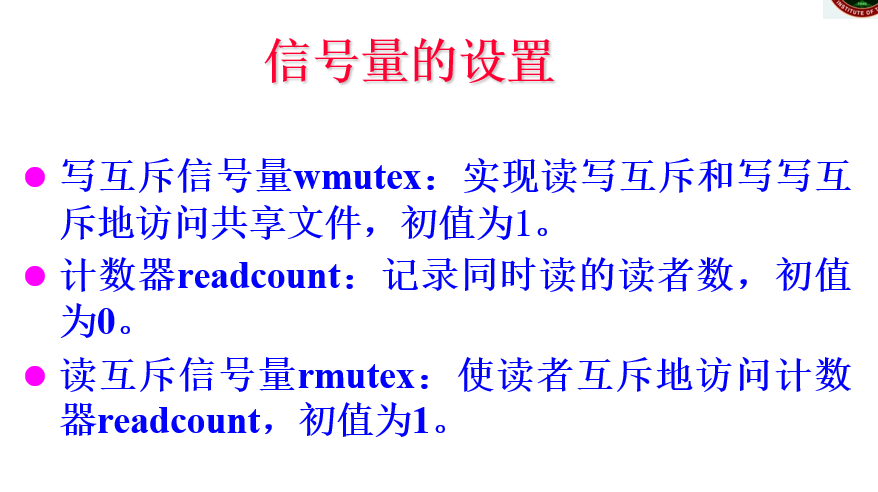


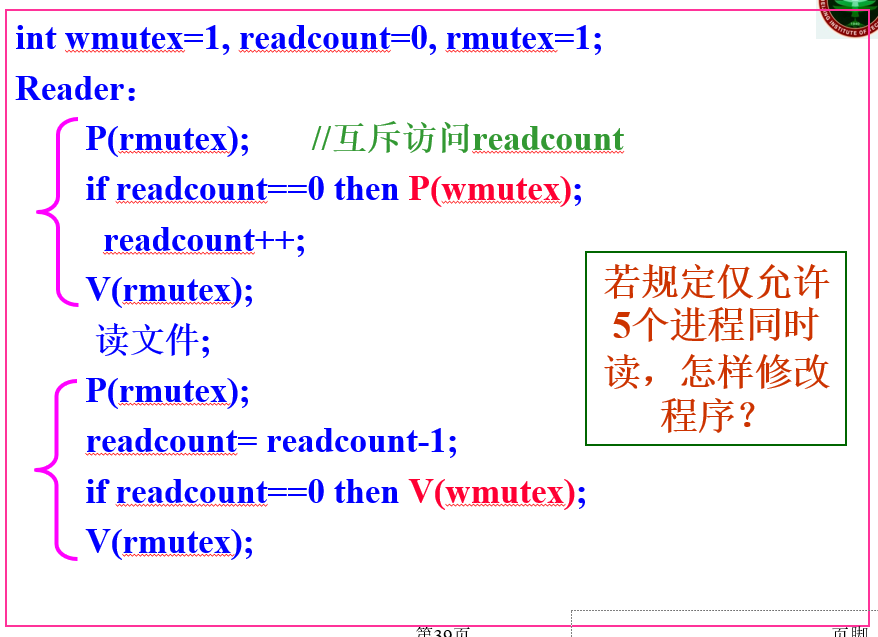


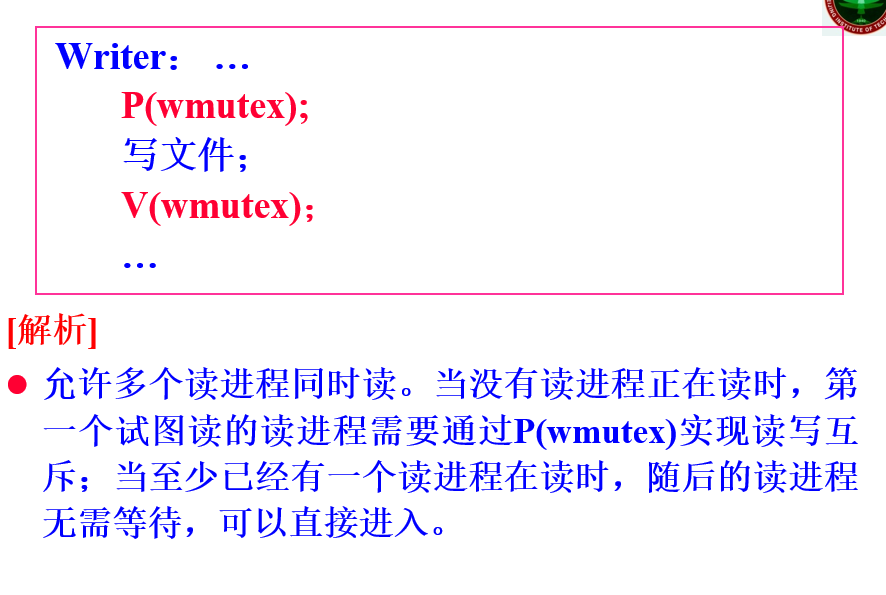
**注意这里要有while循环！！**

**12、读者和写者模型**

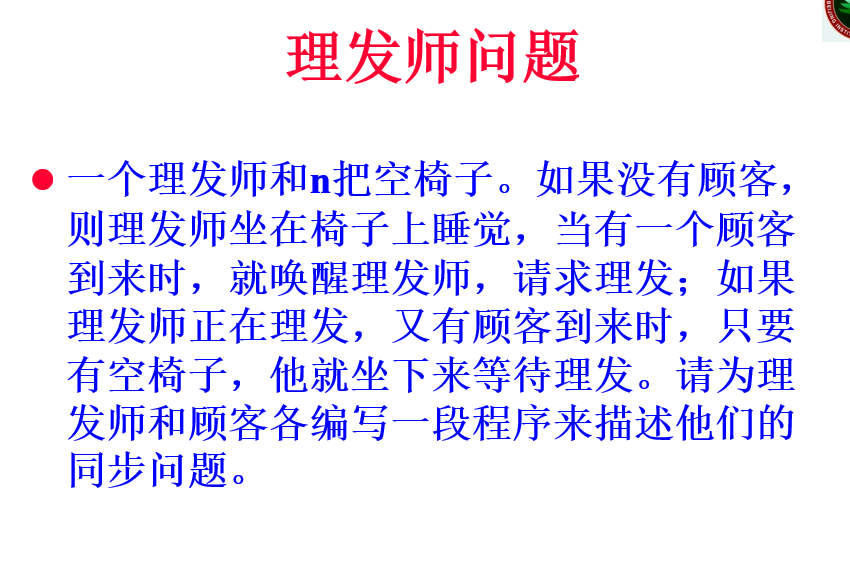


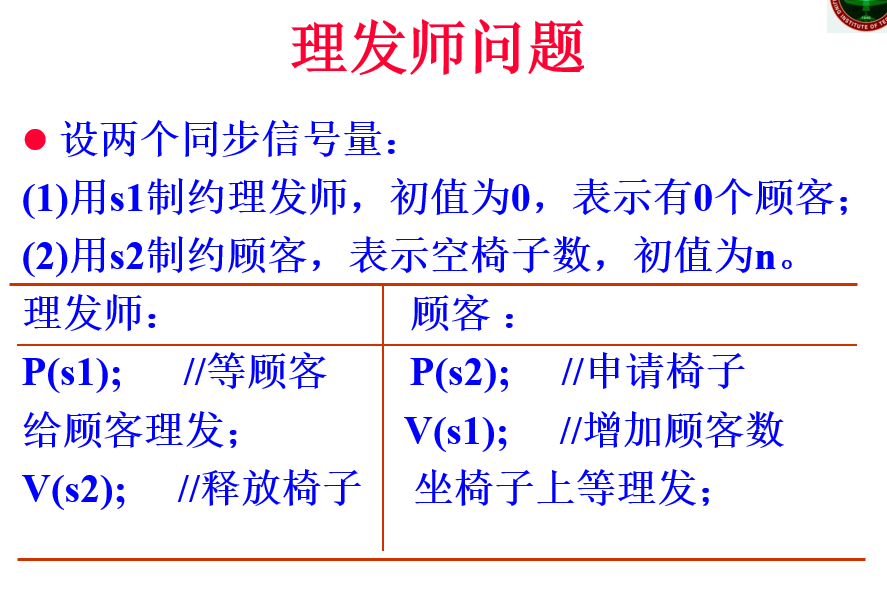




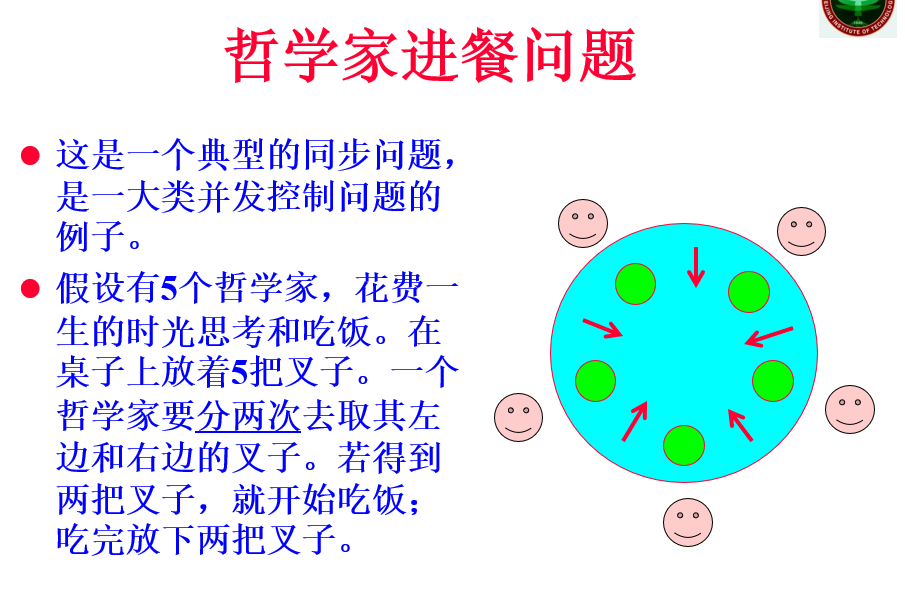


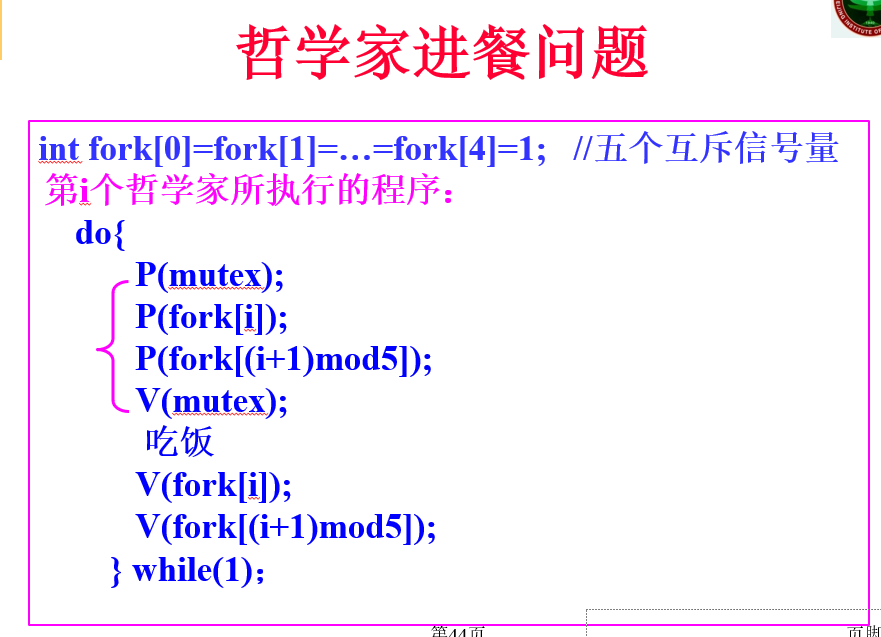
**13、理发师问题**





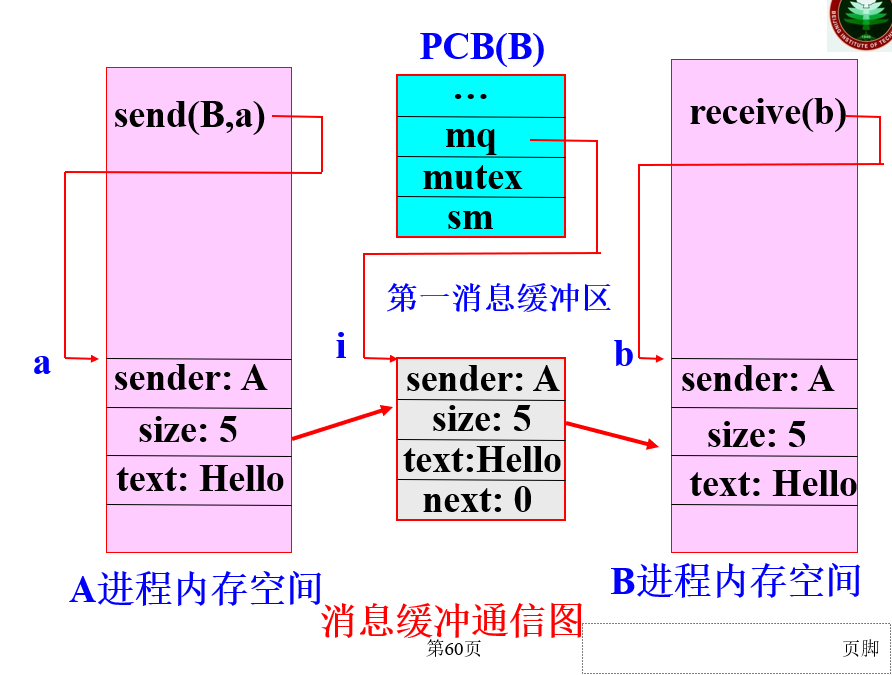
**14、哲学家就餐问题**

****

****

15、高级通信指进程采用**操作系统**提供的多种通信方式来实现通信。如**消息缓冲、信箱、管道、共享主存区**等 （**背！考！**）

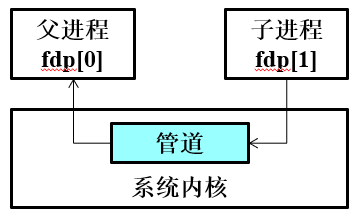
16、消息缓冲（属于直接通信）：



* 发送者先在**自己的地址空间**形成一个**消息发送区**，将消息写入其中，然后调用发送原语。
* **发送原语：**从**系统缓冲区**申请一个**消息缓冲区**，将消息从发送区传入其中， 然后挂到接收进程的**消息队列上。**
* **接收原语：**将消息接收到自己的**接收区。**

17、信箱通信（属于间接通信）：不指定接收进程的名字，由接收者自行处理。

18、管程通道：是指用于连接一个读进程和一个写进程的共享文件，又称pipe文件。OS强制实施互斥。通过操作系统的核心缓冲区来实现单项通信，采用**先进先出**的环形缓冲区。Pipe文件在磁盘上没有定义。



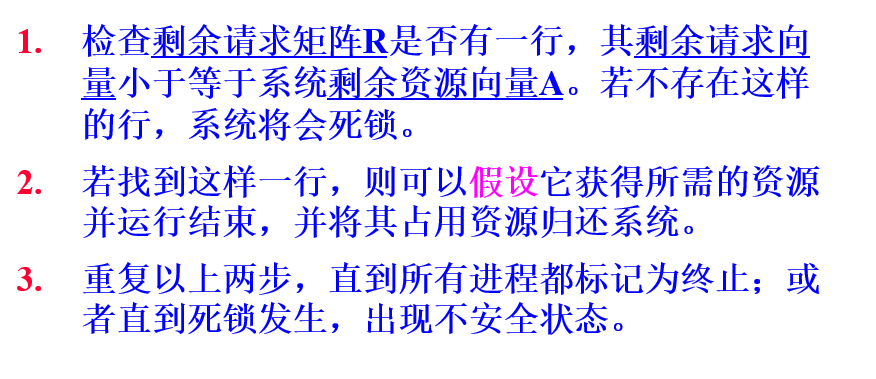
19、共享存储区：申请创建一块共享存储区，并将共享存储区映射到各自的地址空间。它不通过缓冲区进行交互，而是通过一个数据结构进行描述。

20、死锁产生的必要条件：**互斥条件、保持和等待条件、不剥夺条件、循环等待条件。**根本原因：对独占资源的共享，并发进程的同步不当。

21、解决死锁的方法：**鸵鸟算法，死锁的预防、死锁的避免、死锁的检测和恢复**。

死锁的预防即破坏其四个条件。

22、**银行家算法！！！！！！**



给出进程完成序列即可。

# 第4章 存储器管理

**各种方案的实现原理、采用的数据结构、分配和回收算法。**

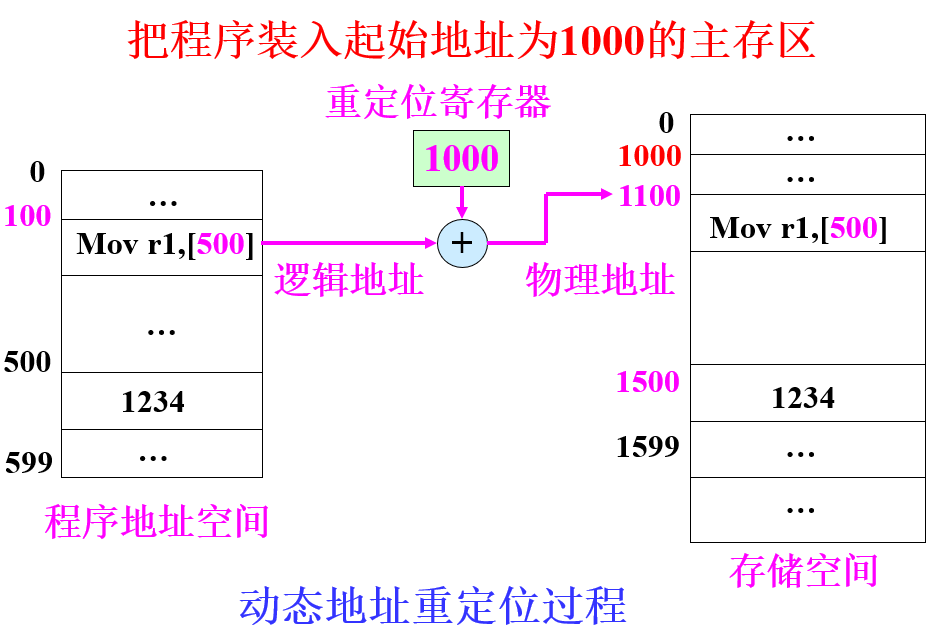
1、存储器管理的功能：**存储器分配、地址转换和重定位、存储器保护、存储器扩容、存储器共享（填空）**

2、逻辑地址空间：经编译链接后的程序大小所限定的空间，其各个地址以“0“为参考地址。

3、存储空间：物理存储器中的全部物理单元集合所限定的空间，一个程序只有从地址空间装入存储空间后才能运行。

4、地址重定位：

* 静态重定位：在执行进程前，将逻辑地址转入到物理地址。要求有连续的存储空间，而且程序在执行时不允许在主存移动，适合于**单道**批处理系统。
* 动态重定位：依靠重定位寄存器，进行地址转换。可移动程序，不需要有连续的主存空间，便于共享。用于多道批处理和分时系统。如多段编译，但每一步都是原样装入内存中。



5、程序链接方式：

* 静态链接：在构造可执行文件时，就链接在一起，形成一个**一维**的地址域。
* 动态链接：
  + **装入时**动态链接：边装入边链接；
  + **运行时**动态链接：运行时才链接，便于实现程序模块**共享**，减少系统开销。

6、存储器保护：其主要涉及两个方面：**地址越界**和**存取方式的保护（填空）**

7、存储器分为两个部分，一部分是系统占有区，另一部分是用户占有区。存储器管理指对**用户区**的管理。

8、存储器共享：允许多个进程共享同一个**主存区**，以提高存储器利用率。既可以是数据区，也可以是程序区。被共享的程序叫**可重入程序**，其代码无论执行多少遍，都保持不变。具有这种性质的程序又叫**纯代码**。

9、单用户单道程序的存储器分配，主存只有一个用户作业，通过对地址以及上下界的大小比较进行越界判断操作。

10、分区分配分为**固定式分区**和**可变式分区**。（**填空！！**）把主存分配**若干连续**的区域

11、固定式分区适合多道程序**最简单**的存储管理方式，把主存预先划分成几个**大小不等**的分区。系统设置**分区说明表**用以描述主存使用情况，别忘了最下面还有**操作系统分区**！

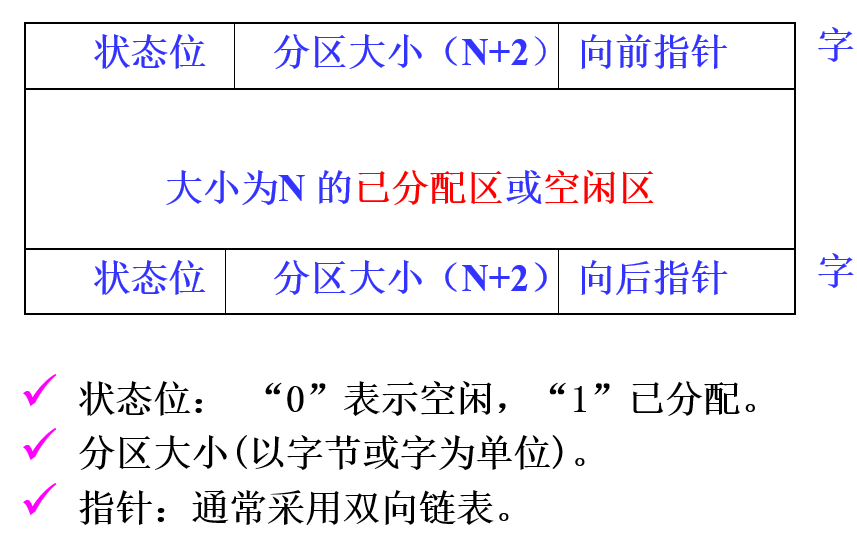
12、可变式分区——分区说明表：

* 分配主存：
  + 从**未分配区表**中找一个空闲区。将其一分为二，一部分分给作业，另一部分留在表中。
  + 在**已分配区表**中进行记录。
* 回收主存：
  + 将回收的分区登记在**未分配区表**中。
  + 将该作业占用的**已分配区表项置为空**。

缺点：通过结构数组进行构表，因为长度未知要么溢出，要么浪费。

13、可变式分区——空闲区链：

**已分配区在进程PCB中描述！**

****

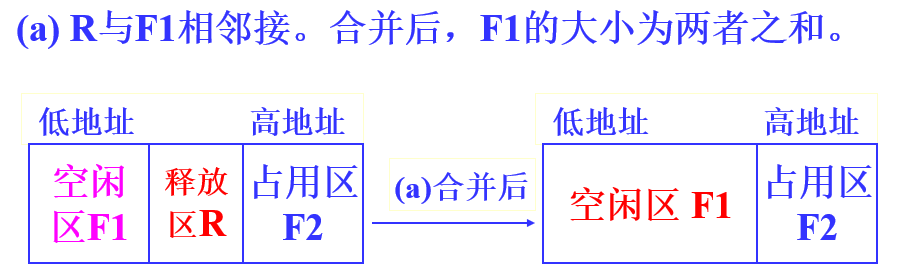
14、可变式分区——分配算法（记住**算法名词**和理解**其中方法！**！）

（1）**首次适应法**：存储程序从**起始地址最小**的地方开始扫描，直到找到足够大的分区

（2）**最佳适应法**：找到满足进程要求且为**最小**的空闲区。（需要查找所有分区，所以效率低下，会出现**无用的碎片**）。改进：从小到大先对分区排序

（3）**最坏适应法**：扫描**所有**空闲区，找到满足需求且**最大**的空闲区，将其分割开来。可能满足不了大进程的要求。改进：从大到小排序。

15、**回收算法**：若释放区与空闲区相邻接，要进行合并。





16、地址重定位和保护

* 固定分区
  + 静态重定位，进程运行时使用主存物理地址。
  + 设置**上、下界寄存器**来实现存储器保护
* 可变式分区
  + 动态重定位，进程运行时CPU给出的是程序的逻辑地址。
  + **基址+限长寄存器**。世界上第一台超级计算机CDC 6600使用该方案。

17、分区管理优缺点：

优：简单，实现多道程序的共享主存

缺：主存利用不够充分，存在碎片；没有实现主存扩容

18、覆盖与交换技术：覆盖与交换技术是解决**大进程**和**小主存**矛盾的两种存储器管理技术，其实质是对主存进行**逻辑扩充**。

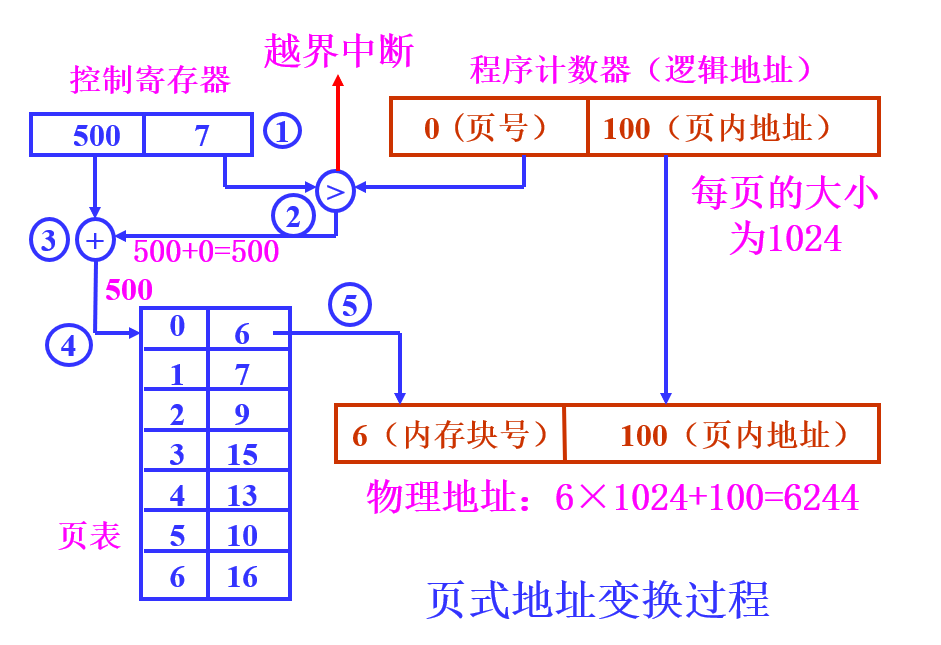
* 覆盖：让那些不会同时执行的程序段**共享一个主存区**。但是系统必须提供**覆盖管理控制程序**，用户提供**覆盖结构。**
* **交换内容**：暂时不运行的进程的部分或全部信息移到**外存**。**交换目的**：解决主存容量不够大的问题；保证**分时用户的合理响应**时间。**交换的时机：进程用完时间片或等待输入输出时；进程要求扩充其占用的存储区而得不到满足时**。**交换技术的关键**：设法减少每次交换的信息量。常将进程的副本保留在外存，每次换出时，仅换出那些修**改过的信息**即可。

19、页式存储器管理：占用**不连续**的存储空间，克服碎片。

1. **页框**：对主存区而言的，将其分成1024大小的若干块。
2. **逻辑页**：相对程序段而言的，将其分成与主存块大小相等的逻辑页；
3. **逻辑地址**为其中的页号与页内地址。
4. **页表**：页框-逻辑页的映像表。**页表存放在主存。**
5. **控制寄存器：**在页式管理中，系统为每个处理机设立一个控制寄存器，用以记录现运行进程的页表始址和页表长度**。**
6. **硬件动态地址转换机构MMU**负责将逻辑地址转换为物理地址。

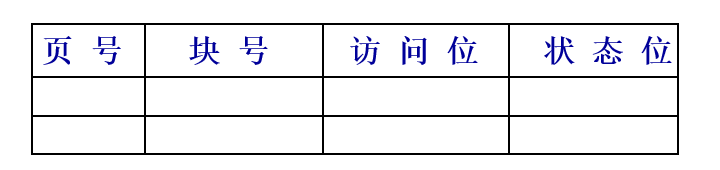
**20、页式动态地址转换（考点！！）**

块号\*1024+页内地址 = 物理地址

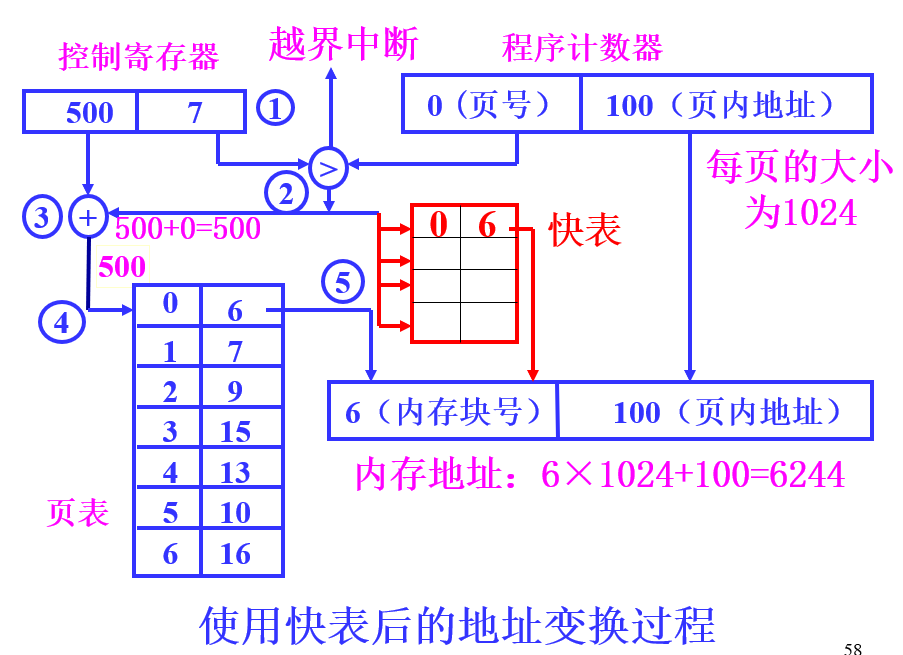


21、快表与联想存储器

**联想存储器**：MMU中增设一个具有并行查找能力的高速缓存（寄存器组）。用来存放最近访问过的页表项。包含页表项叫做**快表**。



**访问位**：最近是否访问过。**状态位**：是否正被占用。



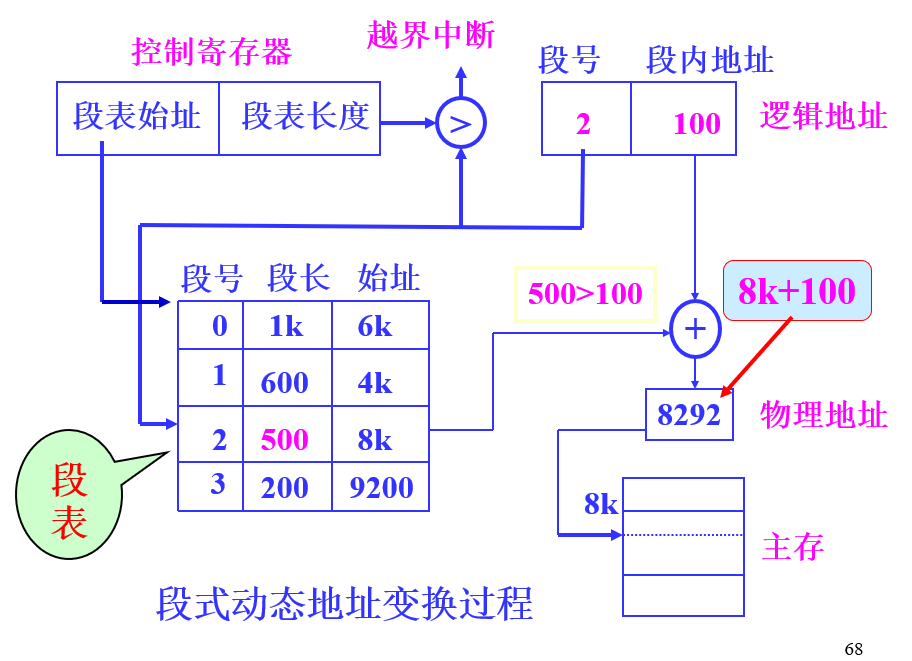
快表和页表同时查，快表中有待查找的页号时，立即停止正常访问主存页表的过程。快表中没有要查询的页。则继续正常的转换过程。还要把从主存页表中取出的块号和CPU给出的页号一起写入快表中。**（找空闲快表项；无时再找最近没被访问的快表项）**

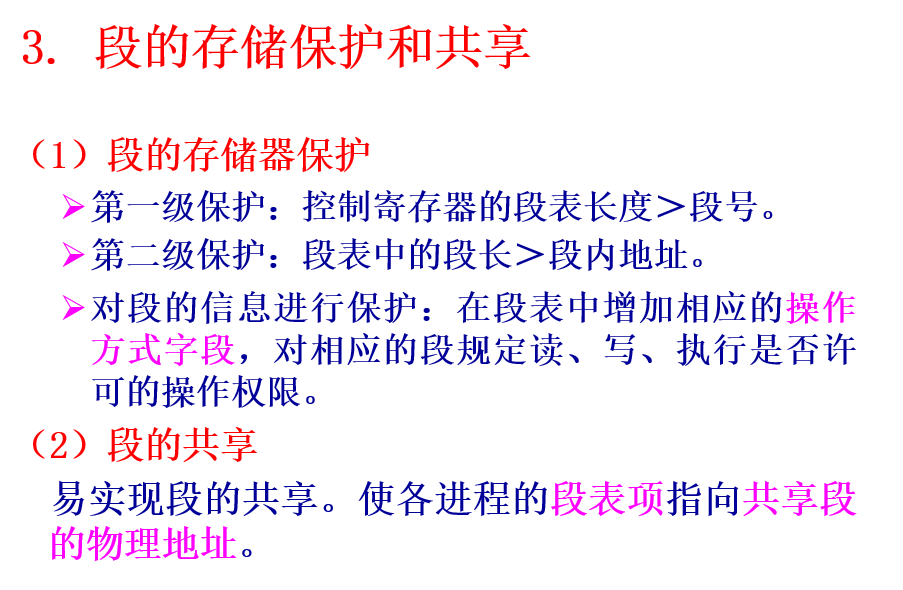
21、进程分配：

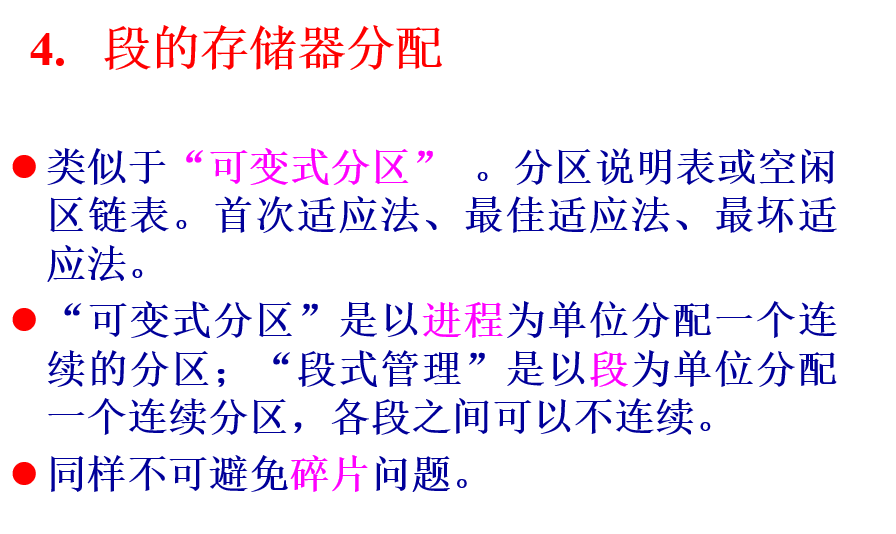
1. 页表：放在内存中
2. 进程控制块！
3. 存储空间使用情况（**背名词必考！！**）
   1. **存储分块表：**记录存储器中的块的占用情况。表的第一项指出当前空闲块总数，第二项为指向第一个空闲块的**指针**，各空闲块通过单向链链接在一起。分配主存时，查存储分块表。先查空闲块总数**是否够用**，再分配。同时为进程建立页表。进程完成时，回收主存块。
   2. **位示图：**每个主存块对应位示图中的一位。0表示空闲，1表示被占用。块越小，位示图越大；块越大，位示图越小。

22、页式管理**共享受限**。不好共享程序中的部分代码。保护：可在页表中增加对该页的操作方式位，以表示是**可读写/只读/只执行（背）**等。

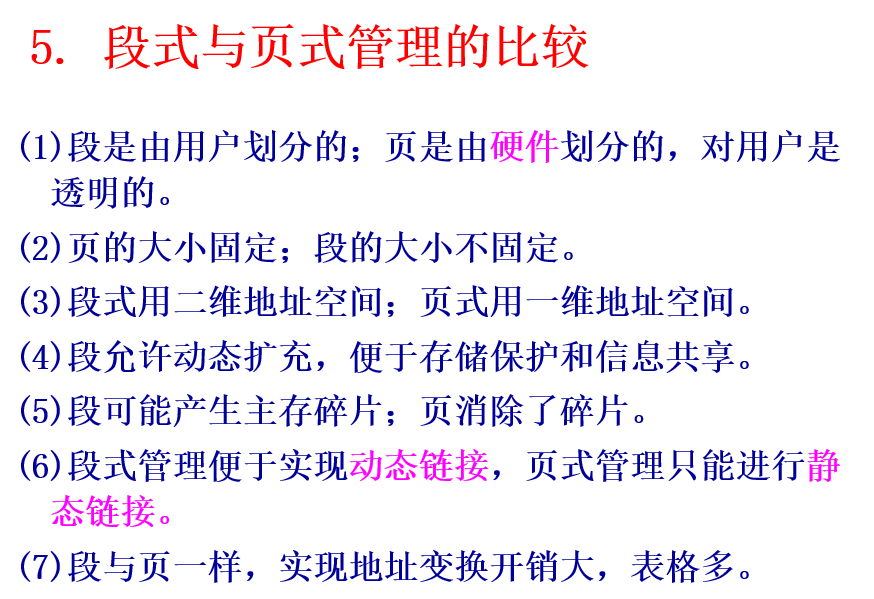
23、段式存储器管理：解决**共享问题**。二维地址空间，段名和段内地址。







**必考考点！！！！！！！！！！！**

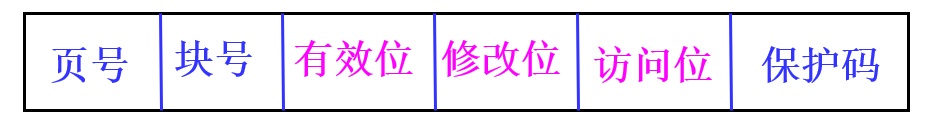


24、虚拟存储技术：进程运行时，其执行实体不必完全在主存中。程序可以比物理主存大。32位机就是2^32（4GB）虚拟内存，64位机同理。

25、程序访问的**局部性原理（填空必考）**：时间局部性+空间局部性。

26、地址转换机构MMU，保有很大的外存容量的同时，尽可能保持高速的主存的存取速度。

27、页式虚拟存储管理



(1) **有效位（状态位）**：用来指示某页是否在主存。

* 为1表示该页在主存，完成正常的地址变换；
* 为0表示该页不在主存，由硬件发出一个**缺页中断**，转操作系统进行缺页处理。产生缺页中断时，一条指令没执行完，OS进行缺页中断处理后，**应重新执行被中断的指令。（和CPU中断不同！！！）**

(2) **修改位**：指示该页调入主存后是否被修改过。“1”表示修改过，“0”表示未修改过。

(3) **访问位（引用位）**：指示该页最近是否被访问过，“1”表示最近访问过，“0”表示最近未访问

28、页面淘汰算法：**（4种算法必须全部掌握！！！）**

（1）最佳置换（OPT）

选择以后不再访问的页或经很长时间之后才可能访问的页进行置换

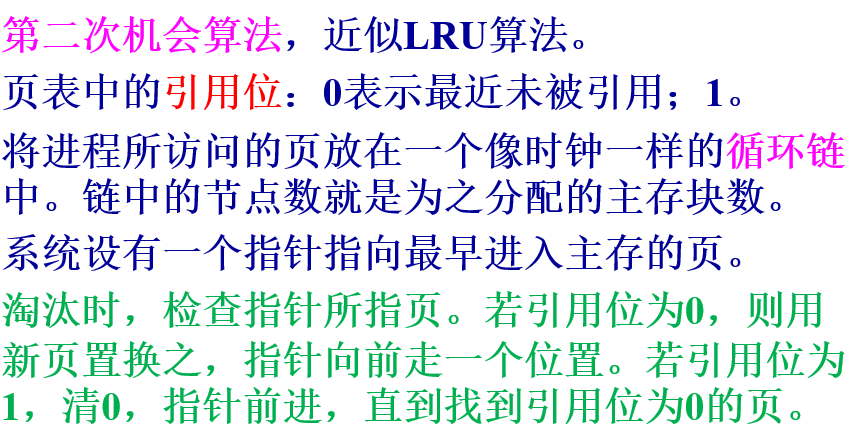
（2）先进先出（FIFO）

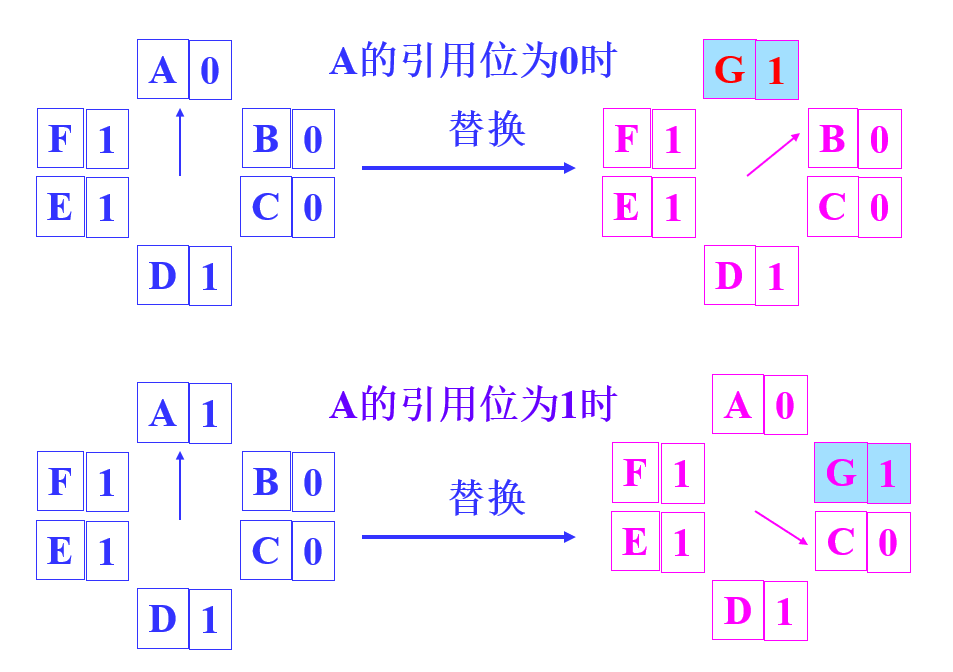
有可能**出现抖动**和**Belady异常。**

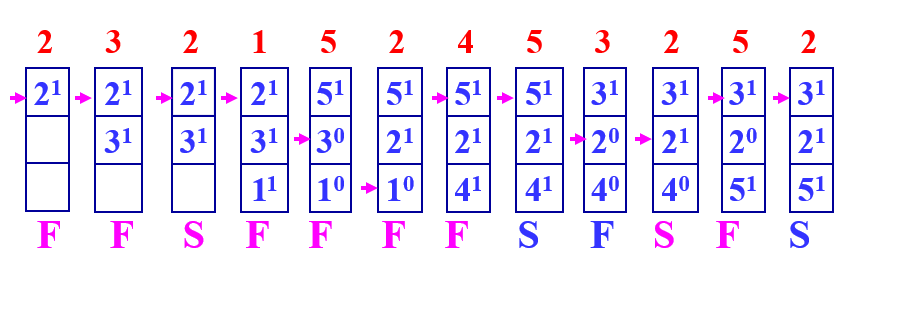
（3）LRU页面置换算法

根据**局部性原理**，形成**进程的工作集**，将最近被调用过的进程放置最顶端。数据结构为**栈**，用**双向链表**链接。**在任何时刻，主存块为m+1时，存于主存中的一串页面必然包含主存块为m时存于主存中的各页。决不会出现Belady异常。**

（4）时钟页面置换算法（**必考！！！！！**）







29、页面尺寸

页尺寸大-页内部碎片多；页尺寸小-页表长。

30、**写时复制技术**（背！填空考）

若没有进程向共享主存页写时，两个进程就共享之。若有进程要写某页，系统就把此页复制到主存的另一个页框中，并更新该进程的页表，使之指向此复制的页框，且设置该页为可读/写

31、多级页表

注意理解以下的定义：

* **大页表：(地址空间4GB)/(页4KB)=1M个页表项。若每个页表项占4B，则最大的页表为4MB，需占用1K个页框（有时最多1k个）。采用多级页表结构，页表在内存不必连续存放。**

其中，页也称虚页是对进程进行划分的，常见为4KB/4MB，对应划分数量向上取整，页框的数量决定页表项数量与页数量。

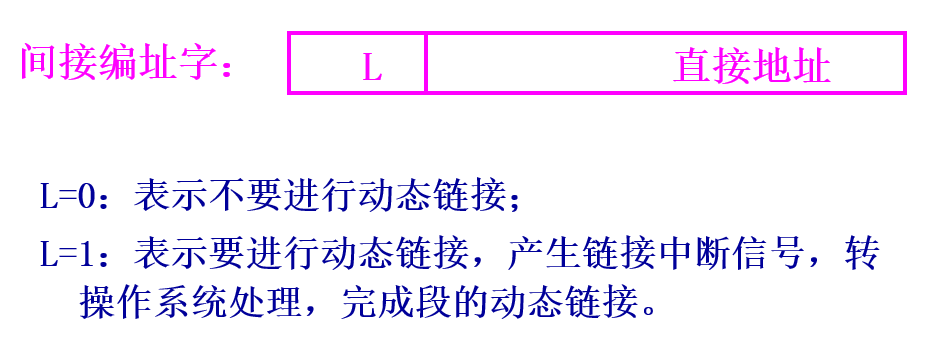
32、段式虚拟存储：

增加**动态增长位**：是为动态数组和动态表设计的。

段中断。

33、段的动态链接和装入

增加两个硬件功能：**间接编址字**、**连接中断位**



34、利用段的**动态链接**很容易实现共享，需要建立一个**共享段段表。**

35、**段页式管理**的基本思想：程序的逻辑结构按段划分，每段再按页划分。

# 第5章 文件系统

1、文件由两部分组成：**文件控制块**和**文件体**。操作系统将文件控制块放到**文件目录**的结构当中，目录也是以文件形式存在**外存**上的。

2、DOS文件目录项共32B，占用32个字节。

3、文件分类分为：**普通文件，目录文件和特别文件**。

4、文件系统：OS中管理文件的软件机构。包括管理文件所需的数据结构、相应的管理软件和被管理的文件。引入文件系统的主要目的是实现**按名存取文件**。

5、文件目录可分为：**一级目录结构、二级目录结构和多级目录结构**。

6、树形目录：/c/c4/c43/c432共打开了3次！！！

7、无环图目录结构允许**共享**文件和子目录，**避免创建环**，通常采用**硬链接**和**符号链接**来实现

8、文件的存取方式有：

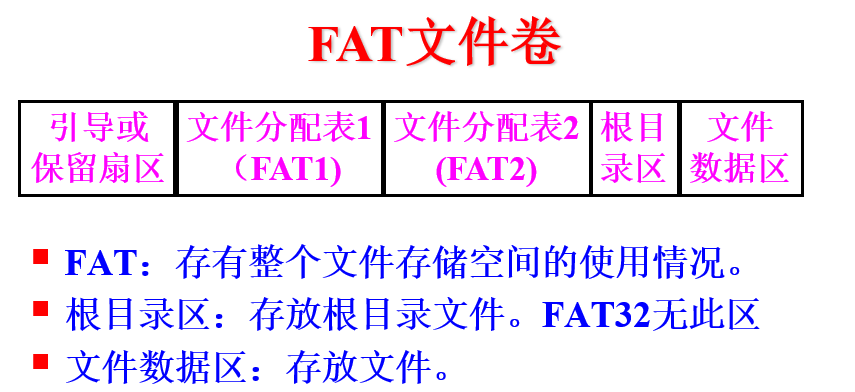
（1）**顺序存取**：按照文件信息的**逻辑顺序**依次存取。适用于顺序访问设备（磁带）和随机访问设备（磁盘）。

（2）**直接存取**（随机存取）：基于文件的磁盘模型，磁盘允许对任意文件块进行随机读和写。

对于变长记录文件，通常采用顺序存取法。

9、文件的物理结构分为：**连续文件、链接文件、索引文件、索引顺序文件**。

10、MS-DOS就是使用**文件分配表（FAT）**来分配和管理磁盘空间的。克服链接文件不能随机存取的缺点。



11、在访问文件时，先将索引表调入内存，避免访问文件时两次访问外存。多级索引结构。当索引表超过磁盘号之后，需要进行建立2级索引。



12、磁盘：寻道时间，旋转延迟时间，读写传输时间

13、把多个逻辑记录存放在一个物理块中的工作叫做记录的**组块**，把一个块中存放的记录的个数叫做**块因子**。进行组块操作时，必须使用**主缓存区**。信息交换是以**块**为单位进行的。用户使用记录时，先将包含该记录的物理块读入内存缓冲区，然后进行**记录分解**。

14、文件存储器存储空间的管理方法：**空白文件目录、空闲块链表、位映像表或示意图**。

1）空白文件目录：系统为所有这些空白文件建立一张表。每个空白文件占用一个表目。适合于文件的**静态分配**（连续文件的分配）。

2）空闲块链表：把所有空闲块连接成一个链表。简单且适合文件动态分配。工作效率低。分配和回收多个盘块时需要多次访问磁盘才能完成。改进：**空闲块成组链表**，即利用盘空闲块来管理盘上的空闲块。改进的方法适合连续结构，也适合连接和索引结构。

3）位映像表或示意图：是适合文件静态分配和动态分配的最简单方法。位映像表（位向量）：每一个二进制位对应一个物理盘块。为1时表示块已分配，为0时空闲。

**优点**：易找到一个或几个连续的空闲块。其尺寸固定，可以保存在主存，便于分配和回收。

**[例]一个10G的磁盘，每个盘块为4k字节时，它要求一个2.5M位的映像表，这个表需占用2.5M/8/4k = 78个盘块。**

15、文件共享通过**硬链接**和**符号链接**来完成。

16、文件保护通过**文件复制**和**增设防护设施**来完成。

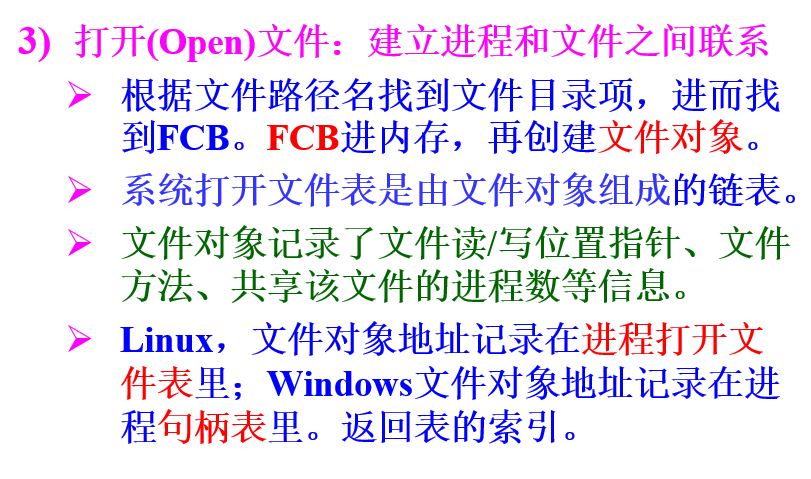
17、常用的文件存取控制方法：**保护域、存取控制表（ACL）**。

1）保护域规定了进程对一组对象的存取权限。用**存取控制矩阵**来记录各个域对对象的存取权限。存取控制矩阵很少用，因为空项很多，**浪费存储空间**。

2）存取控制表：为存取控制矩阵中的**每一列**建立一张**存取控制表(ACL)**，用一**有序对(域, 权集)**表示。**文件的ACL记录在FCB中。R-READ,W-Write, X-eXecute。**



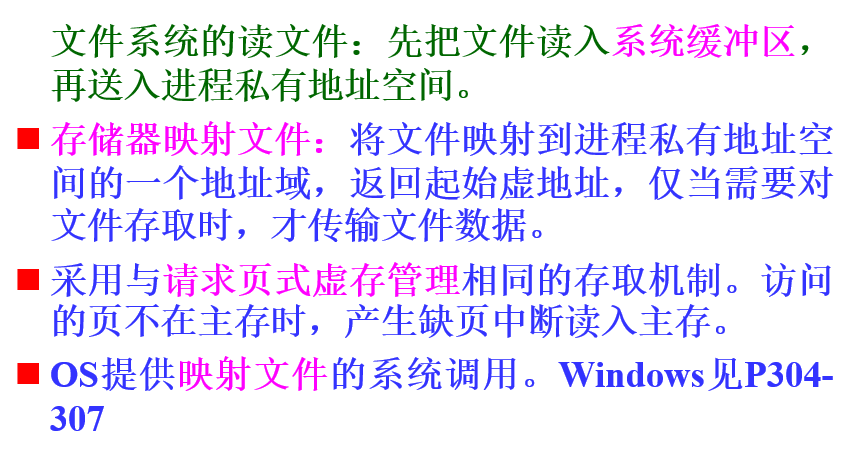
18、文件操作命令

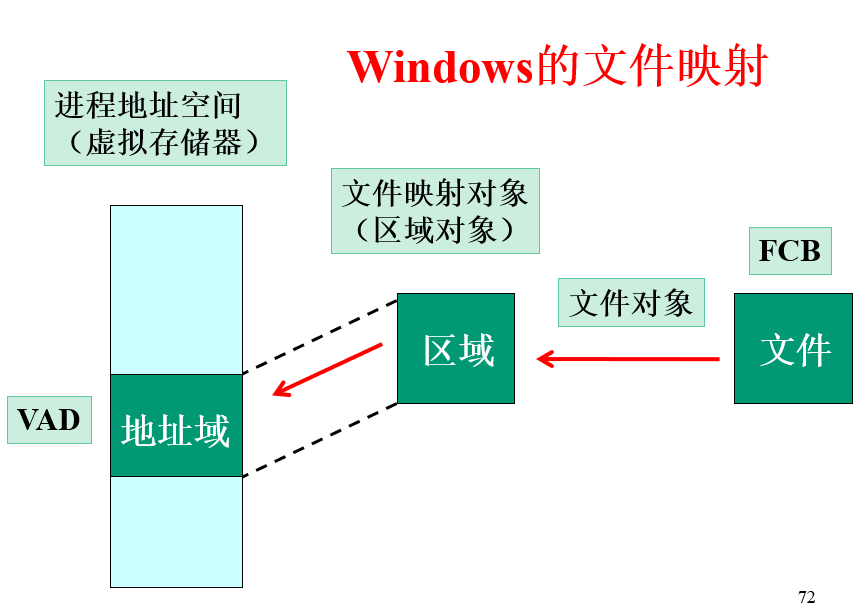


19、文件系统的组织结构

应用程序接口、逻辑文件系统、文件组织模块、基本文件系统、I/O调度及控制模块（由设备驱动和中断处理模块组成）。

20、存储器映射文件





21、小结

**文件系统是OS的信息管理部分，提供文件的按名存取。负责文件的存储、检索和存取保护。**

**文件是具有符号名的相关信息的集合。**

**文件的逻辑结构：无结构的字节流式文件，有结构的记录式文件。**

**文件的存取方法：顺序存取、直接存取。**

**文件物理结构：连续、链接、索引、索引顺序。**

**文件目录是文件系统提供按名存取文件的重要数据结构。单级、二级和多级树状目录。二级和多级目录较好地解决文件的重名和共享。**

**文件存储空间的管理：空白文件目录、位映像表（或位示图）及空白块链。**

**对文件的存取控制：存取保护域、存取控制矩阵、存取控制表(ACL)、口令和密码等。**

**文件的操作命令：文件系统向用户提供的接口。**

**文件系统的层次结构：应用程序接口、逻辑文件系统、文件组织模块、基本文件系统、I/O调度和控制模块。**