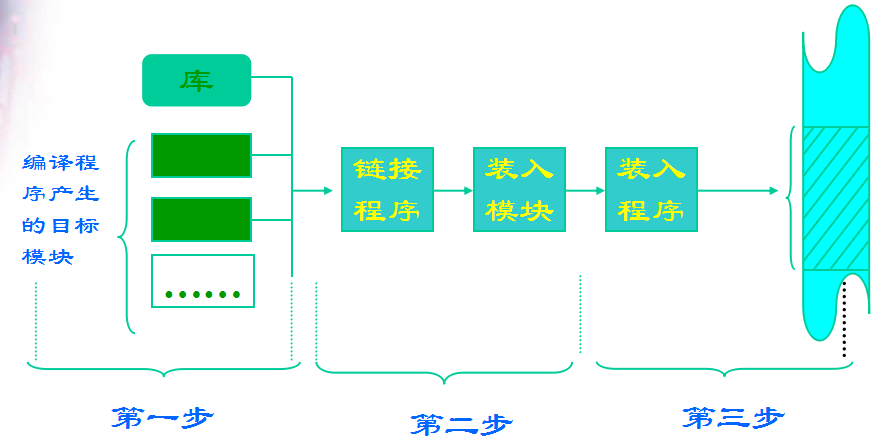
操作系统复习

选择20分 10道题 填空5道题 10分 10题判断题 20分 解答题 综合题

1. 出概念的题 操作系统概念（资源分配器、控制程序……）
2. 了解现代计算机体系结构 各个部分的工作 外设和内存之间交互的IO方式（中断 通道 DMA）
3. 操作系统体系结构 软件体系发展过程 微内核的结构
4. 什么是进程 进程和程序之间的不同 进程和线程之间的不同 进程的组成 进程的变化表现在状态的变化 线程是什么 为什么引入线程 对进程的管理 调度关键算法 进程调度基本思想 调度目的 协作（软件方法跳过） 信号量 重点掌握PV操作 PV怎样用信号量进行进程间的协作（同步与互斥）：分析有几个进程 进程之间有几个关系 涉及几个信号量 信号量的含义和初值 了解通讯的手段（概念上了解）死锁的特征 符合什么条件产生死锁 怎么样解决死锁
5. 分区管理 动态分区的方法 动态分区算法 LRU 先来先服务 离散分配方法 简单页式离散的思路 为什么做离散比连续好在哪里 1级页表2级页表 页式分配的优点 交换的技术是什么 虚拟存储的概念 段式的概念 段式和页式的不同 优缺点
6. 文件系统 文件属性FCB 文件的逻辑结构 物理结构（连续、索引……） 目录结构——树形结构 为什么要改进目录 带来的好处是什么 怎么改进 影响磁盘读取数据的时间 磁盘调度 调度方法会用 容错问题
7. 设备管理 设备的特点 怎么管理IO管理体系 与设备管理相关的硬件技术
8. 内存管理基础
9. 程序执行之前处理：编译、链接、装入
10. 编译：编译程序将用户代码编译程若干个目标模块
11. 链接：将编译后形成的一组目标模块，以及所需库函数链接在一起，形成完整装入模块。
12. 装入：将装入模块装入内存



5.地址表现形式

符号（源程序中）

可重定位的地址（目标模块）

绝对地址（内存映像）

1. 程序链接技术：静态链接、装入时动态链接、运行时动态链接
2. 静态链接：在程序运行之前，将各目标模块及所需函数链接成为一个完整的装入模块，以后不再拆开。
3. 装入时动态链接：在装入内存时，将目标模块边装入边链接。
4. 运行时动态链接：在程序执行过程中，需要该目标模块时才进行链接
5. 逻辑地址：CPU生成的地址
6. 物理地址：内存单元所看到的地址
7. 常用程序装入技术：地址再定位技术、绝对装入技术、可重定位装入技术
8. 地址再定位技术：将逻辑地址和物理内存地址对应起来的过程；地址再定位由操作系统中的装入程序来完成。
9. 绝对装入技术：也称为固定地址再定位；程序地址再定位在执行之前被确定，也就是在编译链接时直接制定程序在执行时访问的实际存储器地址。程序地址空间和内存地址空间是一一对应

优点：

装入过程简单

缺点：

过于依赖于硬件结构，不适于多道程序系统

1. 可重定位装入技术：可执行文件中列出各个需要重定位的地址单元和相对地址值，装入时再根据所定位的内存地址去修改每个重定位地址项，添加相应偏移量。
2. 两种地址再定位方式：静态再定位 动态再定位
3. 静态再定位：装入程序在程序执行之前进行地址再定位，一旦地址定位完成后，程序执行期间不会发生变化。

优点：易实现，无需硬件支持

缺点

程序再定位后不能移动，不利于内存的有效利用。

程序在存储空间中只能连续分配。

16.动态再定位：程序在装入内存时，不修改逻辑地址，在访问物理内存之前，再实时地将逻辑地址转换成物理地址。

优点

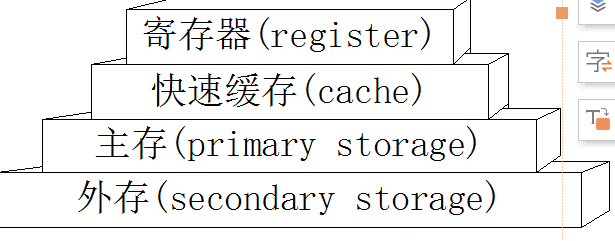
程序在执行过程中可以移动，有利于内存充分利用。

程序不必连续存放在内存中，可分散在内存若干个不同区域，只需增加几对基址一限长寄存器，每对寄存器对应一个区域。

缺点

需要附加硬件支持，实现存储管理的软件算法比较复杂。

9.2

1. 存储器的功能：保存数据，在存储技术和CPU寻址技术许可的范围内，组织合理的存储结构
2. 
3. 存储管理目的

充分利用内存

方便用户使用

解决程序空间比实际内存空间大的问题

存储保护与安全

共享与通信

实现的性能和代价

1. 存储管理任务

存储分配和回收

存储共享

存储保护

存储器扩充

1. 存储管理方案：连续内存分配方式（单一连续存储管理、分区存储管理）、离散内存分配方式（分页存储管理、分段存储管理、段页式存储管理）、虚拟存储器
2. 分区存储管理基本思想：把内存分为大小相等或不等的分区，每个进程占用一个或几个分区；操作系统占用其中一个分区
3. 特点

适用于多道程序系统和分时系统

支持多个程序并发执行

1. 问题

可能存在内碎片和外碎片；

难以进行内存分区的共享。

1. 固定分区存储管理：内存划分为若干个固定大小的连续分区

优点

内存利用率提高了

可以支持多道程序

实现简单

缺点

程序必须预先能够估计要占用多大的内存空间；

内碎片造成浪费；

分区总数固定，限制了并发执行的程序数目。

1. 动态分区存储管理：动态创建分区 优点内碎片 缺点有外碎片
2. 常用分区分配方法：最先适配算法、循环最先适配算法、最佳适配算法、最坏适配算法
3. 最先适配算法：按分区先后次序从头查找，找到符合要求的第一个分区。

算法实质：尽可能利用存储区低地址空闲区，尽量在高地址部分保存较大空闲区，以便一旦有分配大空闲区要求时，容易得到满足。

算法优点：分配简单，合并相邻空闲区也比较容易

算法缺点：查找总是从表首开始，前面空闲区往往被分割的很小，满足分配要求的可能性较小，查找次数较多。

1. 循环最先适配算法：按分区先后次序，从上次分配的分区起查找（到最后分区时再回到开头），找到符合要求的第一个分区

算法特点：算法的分配和释放的时间性能较好，使空闲分区分布得更均匀，但较大的空闲分区不易保留。

1. 最佳适配算法：在所有大于或者等于要求分配长度的空闲区中挑选一个最小的分区，即对该分区所要求分配的大小来说，是最合适的。分配后，所剩余的块会最小。

算法实现：空闲存储区管理表采用从小到大的顺序结构

优点

较大的空闲分区可以被保留。

缺点

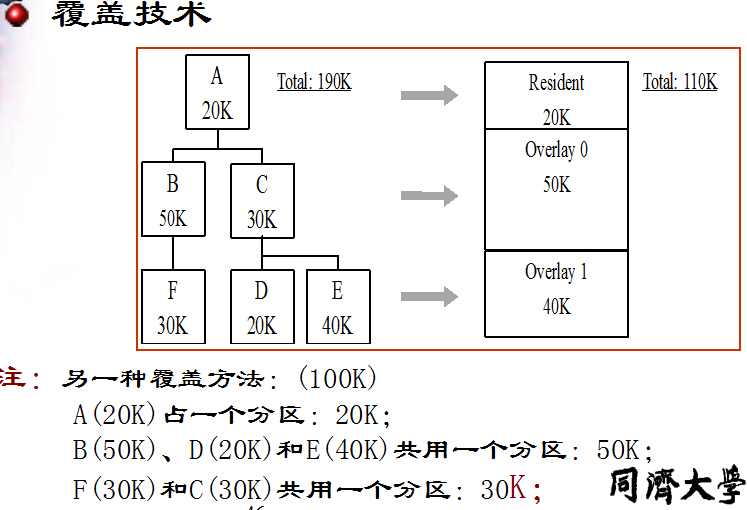
空闲区是按大小而不是按地址顺序排列的 ，因此释放时，要在整个链表上搜索地址相邻的空闲区，合并后，又要插入到合适的位置。

1. 最坏适配算法：分区时取所有空闲区中最大的一块，把剩余的块再变成一个新的小一点的空闲区。

算法实现：空闲区按由大到小排序。

优点：分配时，只需查找一次就可成功，分配算法很快。

缺点：最后剩余分区会越来越小，无法运行大程序

1. 常用分区分配算法存在的问题：碎片问题、分区保护问题
2. 内存扩充技术：借助大容量辅存在逻辑上实现内存扩充，以解决内存容量不足的问题，覆盖技术、交换技术
3. 
4. 覆盖技术缺点

编程时必须划分程序模块和确定程序模块之间的覆盖关系，增加编程复杂度。

从外存装入覆盖文件，以时间延长换取空间节省。

20.交换技术：交换方式

整体交换

交换以整个进程为单位，也称为进程交换；

目的解决内存紧张，进一步提高内存利用率。

部分交换

以分页、分段交换为基础，也称为页面交换，分段交换；

目的是支持虚拟存储系统

交换技术优点：增加并发运行的程序数目；给用户提供适当的响应时间；编写程序时不影响程序结构

交换技术缺点:换入和换出的控制增加处理机开销。

21.两种技术比较

覆盖技术主要用在早期操作系统中，交换技术广泛用于分时系统，其发展导致了虚存技术的出现；

覆盖发生在同一进程或作业内，交换发生在进程或作业之间；

覆盖只能覆盖那些与覆盖段无关的程序段，交换技术不要求用户给出程序段之间的逻辑覆盖结构。

1. 离散分配方式：页式存储管理、段式存储管理、段页式存储管理
2. 页式存储管理：

划分用户空间：把用户程序按逻辑页划分成大小相等的部分，称为页（虚页）；由系统自动完成的，一般，页大小为2的整数次幂

划分内存空间：按页的大小划分为大小相等的区域，称为内存块（物理页面，页框、实页）。

分配内存：以页为单位进行分配，并按任务页数多少来分配。逻辑上相邻的页，物理上不一定相邻。

1. 数据结构：

进程页表：系统为每个进程建立一个页表，页表给出逻辑页号和具体内存块号相应的关系；页表放在内存，属于进程的现场信息

物理页面表：整个系统有一个物理页面表，描述物理内存空间的分配使用状况。

请求表：整个系统有一个请求表，描述各个进程页表位置和大小，也可结合到各进程PCB里

25.内存分配过程：

计算一个任务所需的物理页面块数N；

查位示图，看看是否还有N个空闲页面块；

如果有足够空闲块，则页表长度设为N，可填入PCB中；申请页表区，把页表始址填入PCB；

依次分配N个空闲块，将块号和页号填入页表

修改位示图。

26.页式存储管理评价

优点：解决了碎片问题，便于管理

缺点：不易实现程序共享，不便于动态连接

1. 段式存储管理：

划分用户空间：

按程序自身的逻辑关系划分为若干个程序段；

每个程序段都有一个段名，且有一个段号；

段号从0开始，每一段从0开始编址，段内地址是连续的。

划分内存空间：

内存空间被动态的划分为若干个长度不相同的区域，这些区域被称为物理段;

每个物理段由起始地址和长度确定。

分配内存：

以段为单位分配内存，每一个段在内存中占据连续空间;

各段之间可以不连续存放。

1. 数据结构：

进程段表：记录了段号、段的首（地）址、段长度；每一个进程设置一个段表，放在内存，属于进程现场信息。

系统段表：系统内所有占用段

空闲段表：记录空闲段起始地址和长度，可以结合到系统段表中。

优点

便于动态申请内存

管理和使用统一化

便于共享

便于动态链接

缺点

产生碎片

29.段式页式存储管理：进程空间：段式划分 内存空间：页式管理 页为单位进行分配

ch10

1. 虚拟内存工作过程：

程序装入时：只需将当前需要执行的部分页或段读入到内存。

程序执行中，

待执行的指令或访问的数据不在内存（缺页或缺段），则处理器通知操作系统将相应的页或段调入到内存，然后继续执行程序。

操作系统可将暂不使用的页或段调出保存在外存上，腾出空间存放将要装入的程序以及将要调入的页或段。

1. 益处：

存放大的程序

提供大的用户空间，提供给用户可用的虚拟内存空间大于物理内存。

更多程序并发执行

易于开发，与覆盖技术比较，不影响编程时的程序结构。

1. 虚拟存储特征：

不连续性：物理内存分配的不连续性；虚拟地址空间使用的不连续性。

部分交换

大空间：提供大范围的虚拟地址空间，其总容量不超过物理内存和外存交换区容量。

1. 虚拟页式机制：工作原理：

只在页面需要时，才将其载入内存；

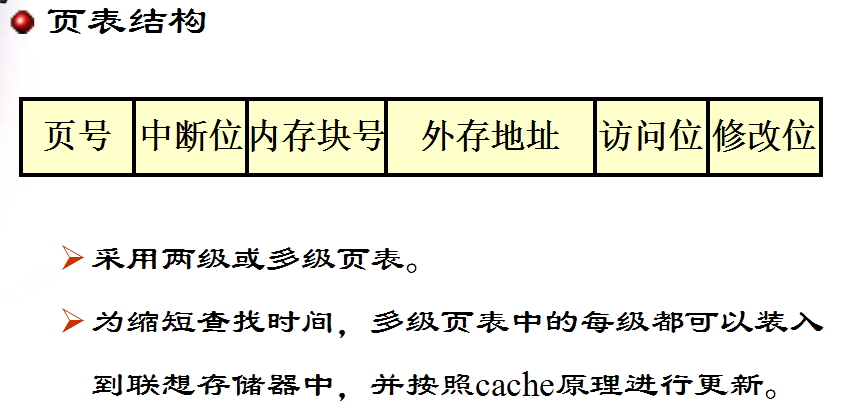
需要更少的输入输出

更小的内存

更快的响应

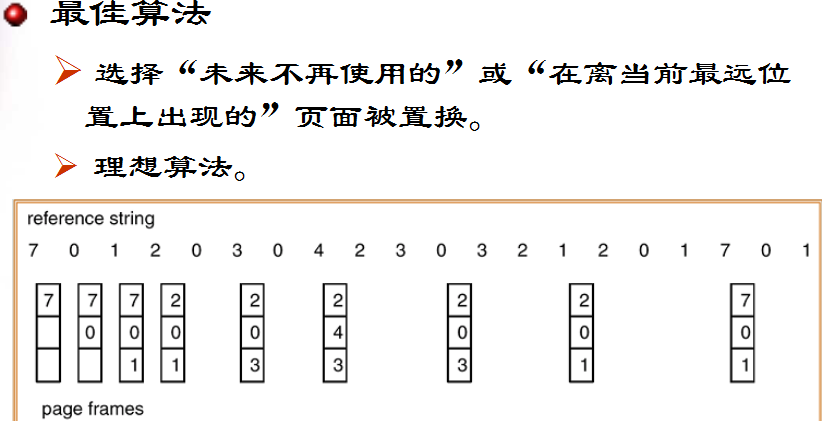
更多的用户

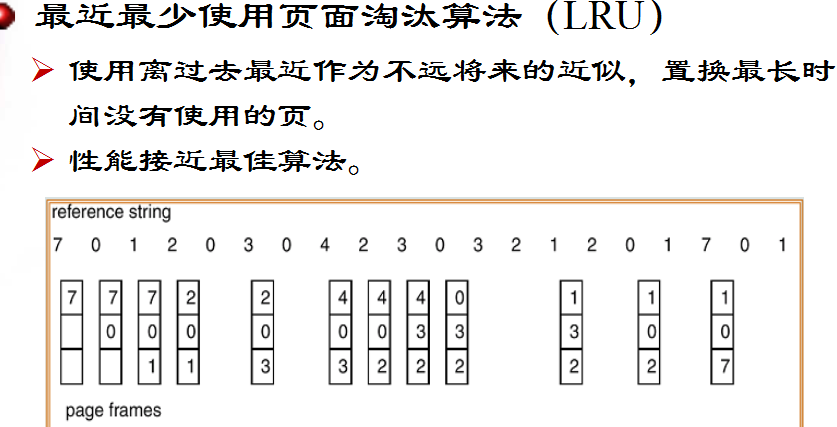
1. 数据结构：

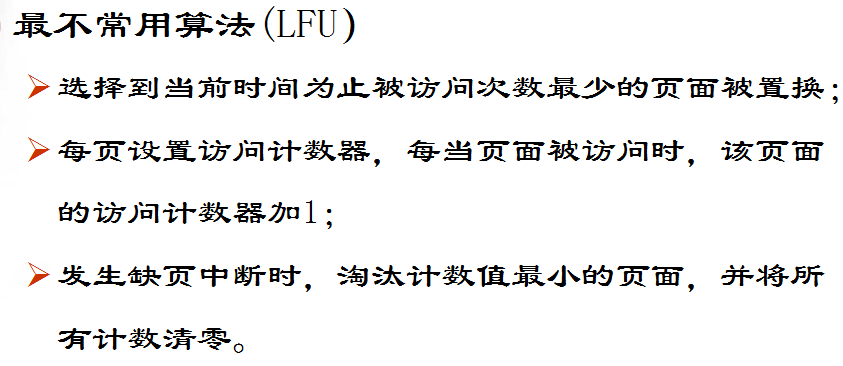


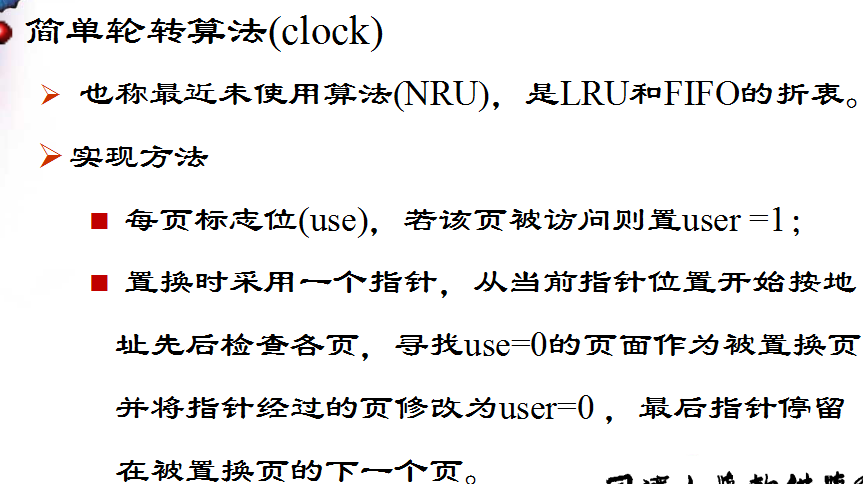
1. 页面置换算法：先进先出算法、最佳算法、最近最久未使用算法、最不常用算法、轮转算法。











7.虚拟存储策略调入策略：用于确定在外存中的页面调入时机；

8.常用策略：请求调页、预调页。

9.请求调页：只调入发生缺页时所需的页面

优点：容易实现。

缺点：对外存I/O次数多，开销较大

10.预调页：发生缺页需要调入某页时，一次调入该页以及相邻的几个页；

常用于程序装入时的调页。

优点：提高调页的I/O效率

缺点：基于预测，若调入的页在以后很少被访问，则效率低。

11.页面调入来源

交换区

进程装入时，将全部页面复制到交换区，以后总是从交换区调入。

调入速度快，要求交换区空间较大。

文件区

未被修改的页面，直接从文件区读入，被置换时不需调出；

已被修改的页面，被置换时需调出到交换区，以后从交换区调入。

1. 调出策略：确定何时将已修改页面调出到外存上。
2. 常用调出策略：预调出、请求调出

请求调出：页面被置换时才调出。

缺点：调入所缺页面之前还要调出已修改页面，缺页进程等待时间较长，

预调出：页被置换之前就调出，因而可以成批调出多个页面。

缺点：可能形成不必要的开销。

14，常驻集：虚拟页式管理中给进程分配的物理页面数目

常驻集确定方式：

固定分配：常驻集大小固定。如：各进程平均分配、根据程序大小按比例分配、按照优先权。

可变分配：常驻集大小可变，按照缺页率动态调整（高或低－>增大或减小常驻集）。性能较好，增加算法运行的开销。

15.置换范围

局部置换：容易进行性能分析

全局置换：更为简单，容易实现，运行开销小。

16.常驻集大小和置换范围的配合策略

固定分配+局部置换

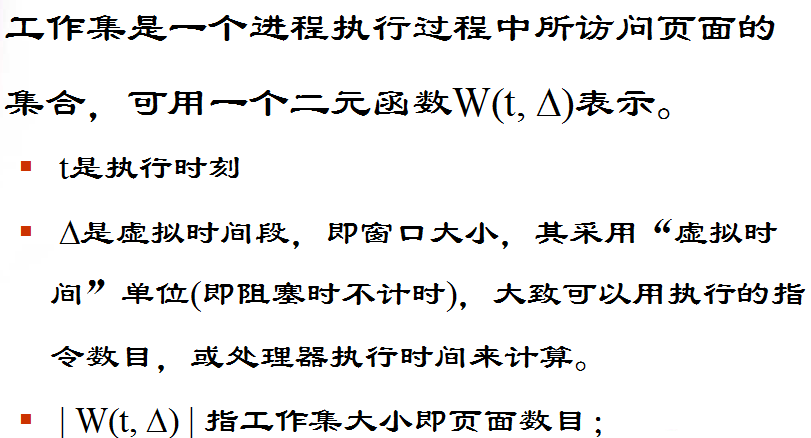
可变分配+局部置换

可变分配+全局置换

OS会一直维持一定数目的空闲页面，以便快速置换；

主要问题是置换策略的选择，如何决定哪个进程的页面将被调出。

17.引入工作集目的是依据进程在过去一段时间内访问的页面来调整常驻集大小。

18.

19.利用工作集进行常驻集调整的策略：

记录一个进程的工作集变化，定期从常驻集中删除不在工作集中的页面；

总是让常驻集包含工作集。

20.常用控制策略：基于工作集策略；“L = S判据”策略；基于轮转置换算法

21.基于工作集策略的算法：

如PFF等，这些算法隐含负载控制策略，只有那些常驻集足够大的进程才能运行，从而实现对负载的自动和动态控制。

“L = S判据”策略：

让缺页平均间隔时间等于对每次缺页的处理时间（即缺页率保持在最佳水平），这时CPU利用率达到最大。

基于轮转置换算法的负载控制策略：

定义一个轮转计数，描述轮转速率（即扫描环形页面链的速率）。

当轮转计数少于一定阈值时，表明缺页较少或存在足够的空闲页面

当轮转计数大于阈值时，隐含系统的进程并发水平过高，需降低系统负载。

Ch11

1. 文件系统由两个不同的部分组成：文件和目录结构
2. 文件：信息以一种单元，即文件形式存储在磁盘或其他外部介质上。

文件是一组带标识的、在逻辑上有完整意义的信息项的序列。

文件是通过操作系统来管理的，文件内容由文件建立者和使用者解释

1. 文件系统

用户观点：文件系统如何呈现在其面前，如：文件由什么组成、如何命名，如何保护文件、可进行何种操作等。

操作系统观点：文件目录怎样实现、怎样管理存储空间、文件存储位置、磁盘实际运作方式(与设备管理的接口)等。

1. 文件系统功能：是操作系统中统一管理信息资源的一种软件；管理文件的存储、检索、更新，提供安全可靠的共享和保护手段，并且方便用户使用。

具体来说：

统一管理文件的存储空间，实施存储空间的分配与回收。

实现文件的按名存取

实现文件信息的共享，并提供文件保护和保密措施；

向用户提供方便使用的接口

系统维护及向用户提供有关信息

文件系统的执行效率

提供与I/O的统一接口

1. 文件属性

名称

类型：由OS和程序定义

位置：指向设备和设备上文件位置的指针

大小

保护

时间、日期和用户标识

1. 文件分类目的

对不同文件进行管理,提高系统效率；

提高用户界面友好性

7.具体分类

按信息保存期限分类

临时文件；永久文件；档案文件

按文件保护方式分类

只读文件；读写文件；可执行文件

按文件性质和用途分类

系统文件、用户文件、库文件

按文件的逻辑结构分类

流式文件；记录式文件

按文件的物理结构分类

顺序（连续）文件；链接文件；索引文件

1. 文件逻辑结构：

无结构文件：流式文件

构成文件的基本单位是字符，文件是有逻辑意义的、无结构的一串字符的集合。

好处:提供很大的灵活性

有结构文件：记录文件

文件是由若干个记录组成，是一个固定长度记录的序列，每条记录有其内部结构每个记录有一个键，可按键进行查找。

9. 物理结构

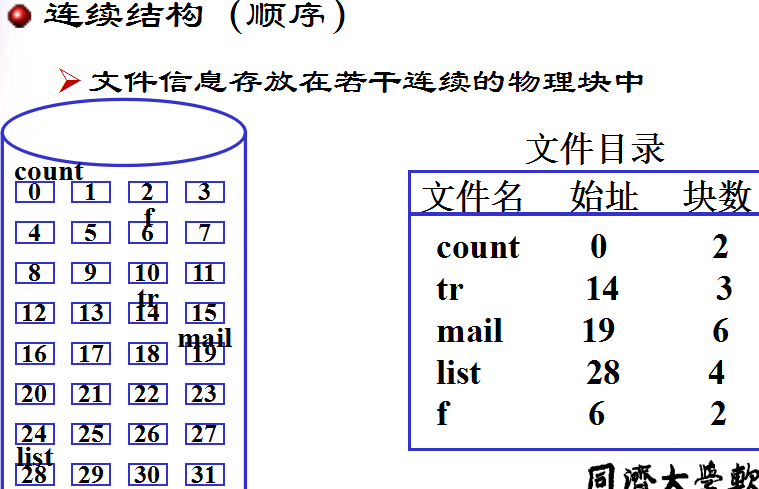
从系统角度来看文件，从文件在物理介质上的存放方式来研究文件.

连续（顺序）结构

链接结构

索引结构

1. 连续（顺序）结构



优点

简单

支持顺序存取和随机存取

顺序存取速度快

所需的磁盘寻道次数和寻道时间最少

缺点

文件不能动态增长

预留空间:浪费

重新分配和移动

不利于文件插入和删除

外部碎片问题

存储压缩技术

11.



12.优点

提高了磁盘空间利用率,不存在外部碎片问题；

有利于文件插入和删除；

有利于文件动态扩充。

缺点

存取速度慢，不适于随机存取；

可靠性问题，如指针出错；

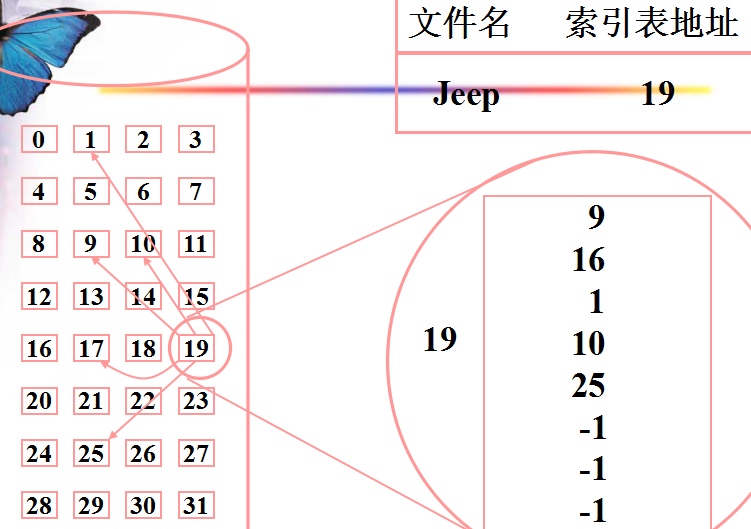
更多寻道次数和寻道时间；

链接指针占用一定的空间。

13.索引结构

文件信息存放在若干不连续物理块中，系统为每个文件建立一个专用数据结构--索引表，并将这些块的块号存放在一个索引表中；

一个索引表就是磁盘块地址数组,其中第i个条目指向文件的第i块。



优点

即能顺序存取,又能随机存取；

满足了文件动态增长、插入删除要求；

充分利用外存空间。

缺点

较多的寻道次数和寻道时间

索引表本身带来了系统开销，

如：内外存空间，存取时间

14.索引表组织

链接模式：一个盘块一个索引表,多个索引表链接起来；

多级索引：将一个大文件的所有索引表（二级索引)的地址放在另一个索引表（一级索引)中。

综合模式：UNIX文件系统采用多级索引结构(综合模式)。

每个文件索引表为13个索引项，每项2个字节。最前面10项直接登记存放文件信息的物理块号（直接寻址）；

如果文件大于10块，则利用第11项指向一个物理块，该块中最多可放256个文件物理块的块号（一次间接寻址）。对于更大的文件还可利用第12和第13项作为二次和三次间接寻址；

15.文件控制块（FCB）：文件控制块是操作系统为管理文件而设置的数据结构，存放了为管理文件所需的所有有关信息（文件属性）.文件控制块是文件存在的标志；

16.文件目录：把所有FCB组织在一起，就构成了文件目录，即文件控制块的有序集合。

17.目录项：构成文件目录的项目（目录项就是FCB）

18.目录文件：为实现对文件目录的管理，通常将文件目录以文件形式保存在外存，这个文件就叫目录文件。

19.一级目录结构

优点：简单、易实现

缺点：命名问题，逻辑组

1. 二级目录结构：

优点：解决名称冲突问题

缺点：用户需要在某个任务上进行合作和访问其他文件时是个缺点。有时系统简单得不允许本地用户文件被其他用户访问。

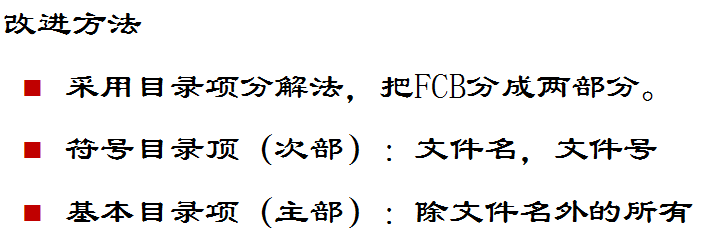
1. 树状目录结构：
2. 路径名的2种形式：绝对路径名 相对路径名。
3. 绝对路径名：从根开始并给出路径上的目录名直到所指定的文件
4. 相对路径名：从当前目录开始定义路径。
5. 文件访问方法
6. 目录检索：用户给出文件名，按名寻找目录项

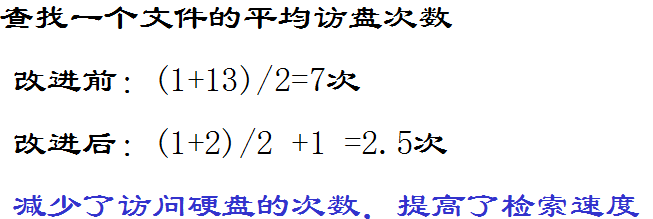
根据路径名检索：

全路径名：从根开始

相对路径：从当前目录开始

1. 文件寻址：根据FCB中文件物理地址等信息，求出文件的任意记录或字符在存取介质上的地址，称为文件寻址。
2. 目录结构改进：目的：加快检索





29.空闲块表 ：所有空闲块记录在一个表中

空闲块链表：把所有空闲块链成一个链

30.位示图 ：用一串二进制位反映磁盘空间中的分配使用 情况, 每个物理块对应一位, 分配物理块为1，否则为0。

申请物理块时，在位示图中查找为0的位，

返回对应物理块号；

归还时，将对应位转置0

描述能力强，适合各种物理结构。

1. 文件系统数据结构：系统文件表：系统打开文件表（整个系统一张）放在内存，用于保存已打开文件的FCB文件号、共享计数、修改标志
2. 可恢复性： 一致性检查：两张表，每块对应一个表中的计数器，初值为0。

安全性写：对写操作进行逐个排序的写方式 延迟写

可恢复文件系统：采用事务日志来实现文件系统的写入 既考虑安全性，又考虑速度性能

Ch12设备管理

1.I/O设备特点

I/O性能经常成为系统性能的瓶颈；

外设种类繁多，结构各异；

输入输出数据信号类型不同；

速度差异很大；

与其他功能联系密切，特别是文件系统。

IO设备分类

2.按照使用特性分

存储型设备

输入型设备（外设主机）

输出型设备（主机外设）

输入输出型设备（交互型设备）

3.按数据组织分

块设备：以数据块为单位存储、传输信息

字符设备：以字符为单位存储、传输信息

4.按外部设备的从属关系分

系统设备：系统生成时，登记在系统中的标准设备

用户设备：系统生成时，未登记在系统中的非标准设备。

5.按资源分配角度分

独占设备

共享设备

虚拟设备

按程序使用角度分

逻辑设备

物理设备

按数据传输率分

高速设备

低速设备

6.I/O设备管理目标

完成设备与内存数据交换，完成用户I/O请求。

向用户提供方便的使用外部设备的接口；

提高CPU与设备、设备与设备间的并行工作能力；

多个进程竞争使用设备时，按一定策略分配和管理各种设备，使系统能有条不紊的工作；

保证设备数据是安全的、不被破坏的、保密的；

与设备无关性（设备独立性）

7. I/O软件设计目标

设备独立性

统一命名

出错处理

可共享设备和独占设备的处理

8.I/O软件分层思想

中断处理程序

设备驱动程序

设备独立的I/O软件

用户空间的I/O软件

9.I/O硬件：I/O设备、I/O过程、设备连接

10.I/O设备特点：操作异步性、设备自治性、接口通用性

11.I/O设备组成 :物理设备、设备控制器

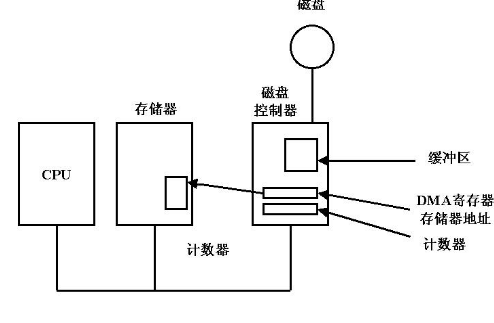
12.I/O过程：准备、启动、测试和等待、结果检查和错误处理

13.设备相关技术：DMA技术、缓冲技术、总线技术、SCSI接口技术

14. DMA技术：与中断区别

工作原理：窃取总线控制权

DMA工作流程



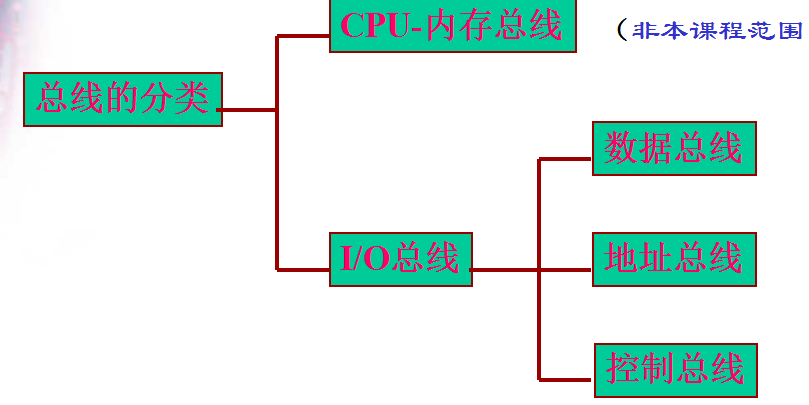
15.引入目的：缓解CPU与外部设备之间速度不匹配的矛盾，提高资源利用率；

缓冲区设置

硬缓冲：在设备中设置缓冲区，由硬件实现；

软缓冲：在内存中开辟一个空间，用作缓冲区

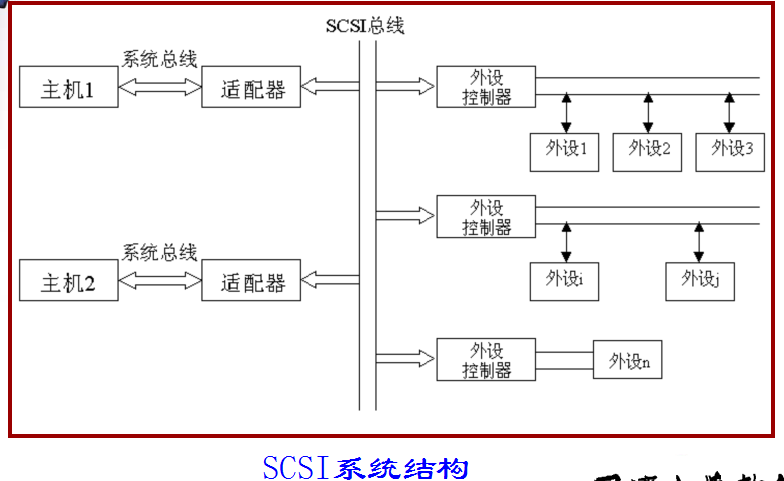
16.总线技术：在计算机系统内各种子系统，构建公用信号或数据传输通道，这种可共享连接的传输通道称为总线。



17.SCSI接口技术：小型计算机系统接口（Small Computer System Interface）

SCSI I/O设备控制器可将新型高速I/O设备增加到计算机系统中；

SCSI 设备控制器的智能化I/O控制，降低了计算机系统的负担，使计算机系统具有更高的I/O能力。



Ch13

1.外存：容量大，断电后可保存信息，速度较慢，成本较低

由两部分组成：驱动部分+存储介质

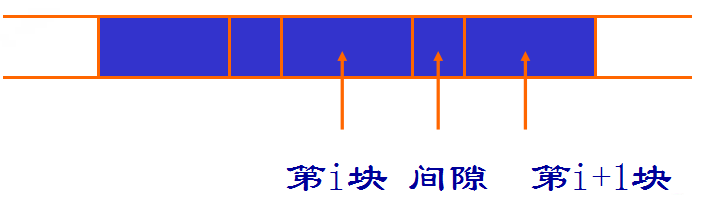
种类很多

外存空间组织与存取方式复杂

2.磁带：永久保存大容量数据

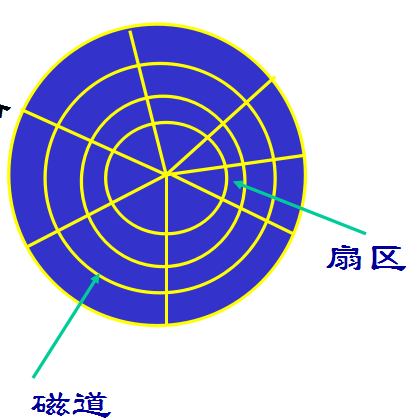
顺序存取设备

存取速度较慢，主要用于后备存储



3.磁盘：直接（随机）存取设备

物理地址形式：磁头号（盘面号）；磁道号（柱面号）；扇区号



4.硬盘分类

固定头磁盘：

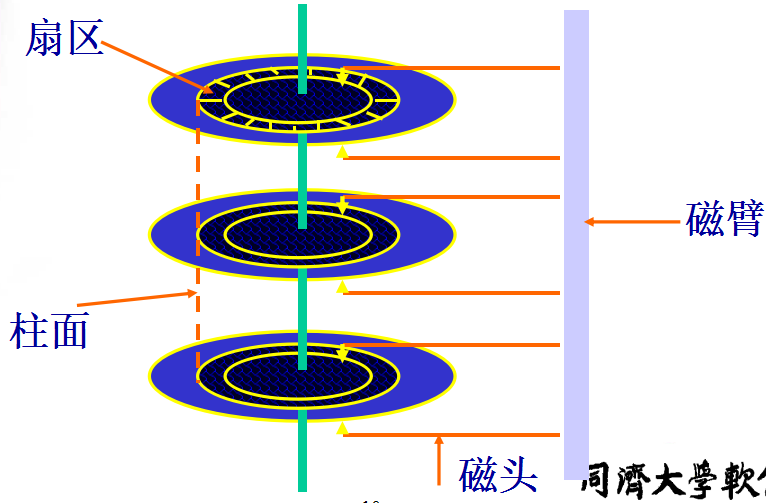
每个磁道设置一个磁头，变换磁道时不需要磁头的机械移动；

速度快但成本高。

移动头磁盘：

一个盘面只有一个磁头，变换磁道时需要移动磁头；

速度慢但成本低。



5.磁盘读取时间：磁盘系统由磁盘本身和驱动控制设备组成，实际存取读写的动作过程是由磁盘驱动控制设备按照主机要求完成的。

寻道：磁头移动定位到指定磁道

旋转延迟：等待指定扇区从磁头下旋转经过

数据传输：数据在磁盘与内存之间的实际传输

6.存取时间：一次访盘时间 = 寻道时间+旋转延迟时间+存取时间

7.光盘特点 ：容量大，速度快，价格便宜；

一般不可写，可读写光盘驱动器价格贵，写过程很麻烦。

光盘的空间结构与磁盘类似。

8.用户对外存的使用：读写外存数据

用户对外存的要求：方便、效率、安全

9.调度方法：先来先服务、最短寻道时间优先、扫描算法（电梯算法）、单向扫描算法

10.先来先服务（FCFS）：按访问请求到达的先后次序服务

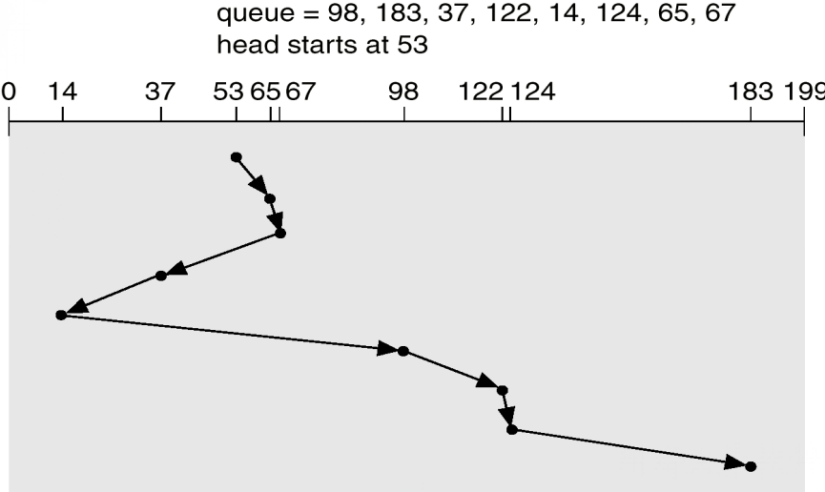
优点：简单，公平

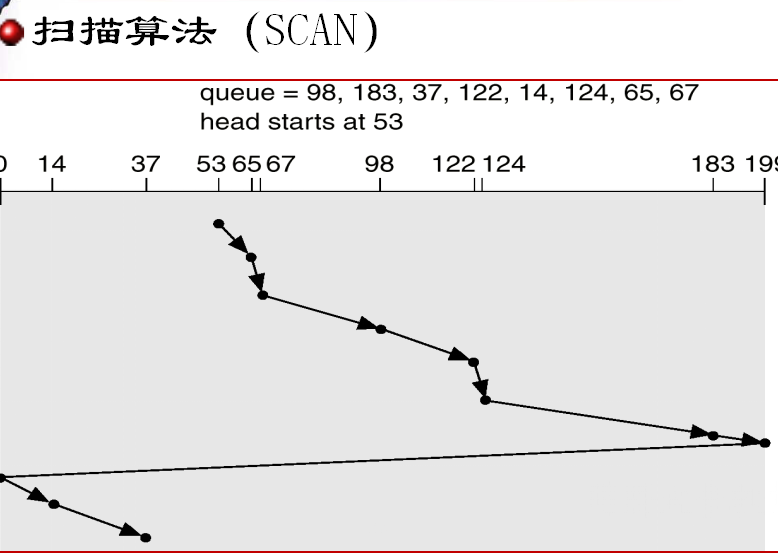
缺点：效率不高，相邻两次请求可能会造成最内到最外的柱面寻道，使磁头反复移动，增加了服务时间，对机械也不利。

11.最短寻道时间优先(SSTF)：优先选择距当前磁头最近的访问请求进行服务，主要考虑寻道优先。

优点：改善了磁盘平均服务时间。

缺点：造成某些访问请求长期等待得不到服务。



1. 
2. 磁盘管理：格式化、引导区、坏块
3. 格式化：低级格式化

将磁盘分成磁盘控制器能读与写的扇区；

每个扇区数据结构由头、数据区域和尾部组成；头部和尾部包含一些磁盘控制器所使用的信息，如扇区号和纠错代码等

磁盘出厂的时候就做好了

高级格式化：

分区

逻辑格式化（创建文件系统）

1. 引导块：绝大多数系统只在启动ROM中保留一个很小的自举装入程序，其作用是进一步从磁盘上调入更为完整的启动程序（保存在磁盘的启动块上）
2. 坏块区

对于简单磁盘，坏扇区可手工处理。Format、Chkdsk

对于复杂的磁盘，对坏块的处理更为聪明。扇区备用或转寄

1. 磁盘容错技术：通过增加冗余的磁盘驱动器、磁盘控制器等来提高磁盘系统的可靠性。

Ok习题1

1. 单处理系统每次只允许一个进程运行，其他必须等待，直到CPU空闲能被调度为止
2. 在一个单处理机系统中，存在5个进程，则最多有(A)个进程处于就绪队列，(B)个进程处于阻塞队列 4,5
3. 在将CPU的执行状态分为用户态和核心态的系统中，应该在核心态下执行的指令依次为(A)、(B)和(C)。而从用户状态转换到系统状态时通过(D)实现的。

屏蔽所有中断、设置时钟的值、停机。设置时钟的值

1. 若进程A与进程B必须互斥地进入自己的临界区，则进程A处于对应的临界区内时，仍有可能被进程B中断
2. 对于记录型信号量，在执行一次wait操作时，信号量的值应当（A），当其值为（B）时，进程应阻塞。在执行signal操作时，信号量的值应（C），当其值为（D）时，应唤醒阻塞对列中的进程

-1，小于0 +1 小于等于0

1. 当进程由执行状态变为就绪状态时，CPU现场信息必须被保存在PCB中
2. 进程被挂起后，状态变为阻塞状态 错误
3. 信号量的初值不能为负数。
4. 线程是CPU调度的单位，但不是资源分配的基本单位
5. wait、signal操作可以解决一切互斥问题

在三种基本类型的操作系统中，都设置了（3），在批处理系统中还应该设置（2），在分时系统中，除了（3）外，通常还设置了（4），在多处理机系统中，则还需要设置（5）

1. 剥夺调度（2）作业调度（3）进程调度（4）中级调度（5）多处理机调度

11.在动态优先权中，随着进程执行时间的增加，其优先权降低

12.系统产生死锁是指（若干进程等待被其他进程占用而又不能被释放的资源），产生死锁的基本原因是（资源不足）和（进程在使用资源的时候推进速度和顺序不当），产生死锁的四个必要条件是（资源互斥性的访问、资源使用的不可剥夺性、环路等待、请求并保持）。

13.为什么要引入挂起状态？ 考虑中级调度

引入挂起状态处于五种不同的需要:  终端用户需要，父进程需要，操作系统需要，对换需要和负荷调节需要。处于挂起状态的进程不能接收处理机调度。

1. 说明引起进程创建的事件

用户登录

作业调度

提供服务

应用请求

1. 引入线程的思想：将进程资源分配和调度分开，引入线程
2. 进程调度的主要功能？高级调度和中级调度主要任务是什么？为什么要引入中级调度？

高级调度：

也称为作业调度或宏观调度；

高级调度的时间尺度通常是分钟、小时或天。

中级调度：

涉及进程在内外存间的交换;

从存储器资源管理的角度来看，把进程部分或全部换出到外存上，可为当前运行进程的执行提供所需内存空间，将当前进程所需部分换入到内存。指令和数据必须在内存里才能被处理机直接访问

1. 如果为每个作业只建立一个进程，为了照顾短作业用户，采用什么调度算法？为了照顾紧急作业用户，应采用什么算法？为了实现人机交互，应采用什么算法？为了让短作业、长作业和交互作业用户都满意，应该采用什么算法？

提示：分别为：短作业先行、优先级调度、分时、多级反馈队列

* 1. 用信号量解决“独木桥”问题：同一方向的行人可连续过桥，当某一方向有人过桥时，另一个方向的行人必须等待；当某一方向无人过桥时，另一方向的行人可以过桥。

答案：

将独木桥的两个方向标记为A和B，用变量countA和conutB表示A和B方向上已经在独木桥上的行人数目，初值为0；再设置三个互斥变量，初值都为1：SA实现对countA互斥修改，SB实现对countB变量的互斥修改，mutex用来实现两个方向的行人对独木桥的互斥使用。

以A方向行人的动作为例，来描述行为如下：

P(SA);

If(countA=0) then P(mutex);

countA=countA+1;

S(SA);

通过独木桥

P(SA);

countA=countA+1;

If(countA=0) then V(mutex);

S(SA);

1. 某系统中仅有4个并发进程竞争某类资源，并都需要该类资源3个，那么该类资源至少(       )个，这个系统不会发生死锁 9
2. 为了对交互式作业进行控制，操作系统为用户提供了一些常用的操作使用接口，不属于操作使用接口的是 系统调用
3. 用户程序在管态下使用特权指令将引起的中断是属于 程序中断
4. 作业与进程的主要区别：前者是由用户提交，后者是由系统自动生成  
   前者以用户任务为单位，后者是操作系统控制的单位
5. 利用P/V操作可以：实现资源的互斥使用 协调进程的执行速度
6. 虚拟存储系统中，完成地址转换工作的是 硬件
7. 在请求分页管理中，若采用先进先出（FIFO）页面置换算法，可能会产生“FIFO异常”, “FIFO异常”指的是 分配的页面数增加，缺页中断的次数也可能增加
8. 静态重定位是在作业（2）中进行，而动态重定位是在作业（4）中进行

装入过程、执行过程

10.由连续分配方式发展到分页存储管理方式的主要动力是（1）；由分页系统发展到分段系统，进而发展到段页式系统的主要动力是（4）和（5）

提高内存利用率、更好的满足多道程序运行的需要、既满足用户需求，又提高内存利用率

11.对外存交换区的管理应以（4）为主要目标，外存文件区的管理应以（2）为主要目标。

提供换入换出速度；提供存储空间的利用率

12.一个计算机系统中，虚拟存储器的最大容量是由（5）确定的，其实际容量是由（4）确定的。

计算机的地址结构；内存和硬盘交换区容量之和

1. 在请求调页系统中，内存分配有两种策略：（3）和（4），（3）的缺点是可能导致频繁地出现缺页中断而造成cpu利用率下降。

固定分配；可变分配；固定分配

14.在保护机构中，操作系统应该处于（1）内，一般应用程序应该处于（2）内，并遵循下面的规则：一个程序可以访问驻留在（5）中的数据；一个程序可以调用驻留在（4）中的服务。

最高特权；最低特权；最低和相同特权；最高和相同特权

1. 虚拟存储器特征 大空间、不连续性、部分交换、容量确定
2. 说明请求分页系统中的缺页中断过程：在请求分段系统中，每当发现运行进程所要访问的段尚未调入内存时，便由缺段中断机构产生一缺段中断信号，进入操作系统后由缺段中断处理程序将所需的段调入内存。缺段中断机构与缺页中断机构类似，它同样需要在一条指令的执行期间，产生和处理中断，以及在一条指令执行期间，可能产生多次缺段中断

17.某系统采用页式存储管理策略，拥有逻辑空间32页，每页是2k，拥有物理空间为1M。

（1）写出逻辑地址的格式

（2）如果不考虑访问权限等，进程的页表有多少项？每项至少有多少位？

（3）如果物理地址空间减少一半，页表结构应如何改变?

解答：

1. 位页号 +１１位页内偏移
2. 进程的页表最多是32项，每项为9位
3. 进程页表项不变，还为32项目，每项为8位。
4. 通道是一种特殊的（3），具有（5）能力。主机的cpu与通道可以并行工作，并通过（7）实现彼此之间的通讯和同步

处理机、执行IO指令集、IO指令和IO中断

19.在I/O设备控制发展过程中，最主要的推动因素是（3）。提高I/O速度和设备利用率，在OS中主要依靠（6）功能。使用户所编制的程序与实际使用的物理设备无关是由（7）功能实现的。

减少主机对I/0控制的干预、缓冲管理、设备管理

20.文件系统最基本的目标是（1），它主要是通过（8）功能实现的，文件系统所追求的最重要的目标是（4）。

按名存取、目录管理、提高对文件的存取速度

1. 在树型目录结构中，用户对某个文件的首次访问通常都采用（2），文件被打开后，对文件的访问通常采用（4） 文件路径名、用户文件的描述符
2. 一个文件在同一系统中、不同的存储介质上的拷贝，应采用同一种物理结构 错误

23.为实现磁盘镜像功能，需要在系统中配置（4），而为实现磁盘双工功能，需要在系统中配置（3）。 两台磁盘驱动器、两台磁盘控制器