操作系统的目标：方便性(用户)、有效性(软硬件)、可扩充性、开放性

操作系统发展的主要动力：提高计算机资源利用率、方便用户、器件的更新换代

操作系统发展过程：

人工操作方式：人机矛盾

缺点：用户独占全机、CPU等待人工操作

脱机I/O方式：减少了CPU的空闲时间、提高了I/O速度

批处理操作系统：提高CPU的利用率、不具有交互性

单道批处理系统：提高系统资源的利用率和吞吐量

特征：自动性、顺序性、单道性

多道批处理系统：提高CPU利用率、提高内存和I/O设备利用率、提高了系统吞吐量

特征：多道性、无序性、调度性(作业调度、进程调度)

缺点：平均周转时间长、无交互能力

分时系统：人机交互

特征：多路性、交互性、独立性、及时性

实时系统：规定时间处理事件

周期性和非周期性、硬实时任务和软实时任务

实时、分时比较：

分时系统交互性更高

实时系统可靠性更高

并行：是指两或多个事件在同一时刻发生

并发：是值两或多个事件在同一时间间隔发生

操作系统基本特征：共享性和并发性是操作系统两个最基本的特征(其余包括虚拟、异步)

OS主要功能：

处理机管理功能：(进程管理)

进程控制、进程同步、进程通信(直接、间接)、调度(作业、进程)

存储管理功能：(方便用户使用、提高存贮器利用率)

内存分配、内存保护、地址映射、内存扩充

设备管理功能：(提高I/O利用率和速度、方便用户)

缓冲管理、设备分配、设备处理、设备独立性和虚拟设备

文件管理功能：(方便用户、提供安全性)

文件存贮空间管理、目录管理、文件的读写管理和存取控制(保护)

用户接口：

命令接口(联机用户接口、脱机)、程序接口、图形接口

OS结构设计：无结构、模块式、层次式、微内核

无结构操作系统：

缺点：逻辑复杂、维护困难

模块化操作系统：

优点：提高了OS可维护性、提高了OS可适应性、加速了OS开发过程(并行开发模块)

缺点：接口不易确定、模块依赖关系可能复杂(大型软件)

分层式操作系统：(简化设计复杂度、下层为上层提供服务)

微内核操作系统：

客户/服务器结构：

提高了系统的灵活性和可扩充性、提高了软件的可靠性、适合于分布式系统

微内核结构：

进程管理、存储器管理、进程间通信和设备管理

程序顺序执行的特征：顺序性、封闭性、可再现性

程序并发执行的特征：间断性、失去封闭性(共享资源引起)、不可再现性

前趋图：有向无循环图

进程：程序的依此执行过程

特征：结构特征：由程序段、数据段及进程控制块三部分构成，总称“进程映像”

动态性、并发性、独立性、异步性(间断性)

进程状态：就绪状态、执行状态、等待状态(前三为基本状态)、新状态、终止状态

挂起状态(被换出内存的状态)

进程控制块：PCB(常驻内存)——链接、索引

进程控制：

进程的创建：进程图、引起创建进程的事件(用户登录、作业调度、提供服务、应用请求)、进程的创建

1.申请空白PCB（一个系统的PCB是有限的）

2.为新进程分配资源（不同于一般的分配，PCB-LIST在一个特殊区域）

3.初始化PCB

4.将新进程插入就绪队列。

进程的终止：引起进程终止的事件(正常结束、异常结束、外界干预)、进程的终止过程

进程的阻塞与唤醒：引起进程阻塞和唤醒的事件、进程阻塞过程、唤醒过程

进程的挂起与激活：进程的挂起过程、进程的激活过程

进程同步：并发进程在执行次序上的协调，以达到有效的资源共享和相互合作，使程序执行有可再现性。

制约关系：资源共享关系、相互合作关系

临界资源：一次仅允许一个进程访问的资源

临界区：进程访问临界资源的那段代码

同步机制应遵循的准则：空闲让进、忙则等待、有限等待、让权等待

死锁：是指两个或两个以上的进程在执行过程中，因争夺资源而造成的一种互相等待的现象，若无外力作用，它们都将无法推进下去。

管理机制：局部于管理的共享变量、对该数据结构进程操作的一组过程、对局部管程数据设置初值

进程通信：

缺点：效率低、通信对用户不透明

高级通信特点：效率高、通信实现细节对用户透明

类型：

共享存贮器系统、消息传递系统、管道通信

处理机调度：

高级调度(作业调度、长程调度、接纳调度)

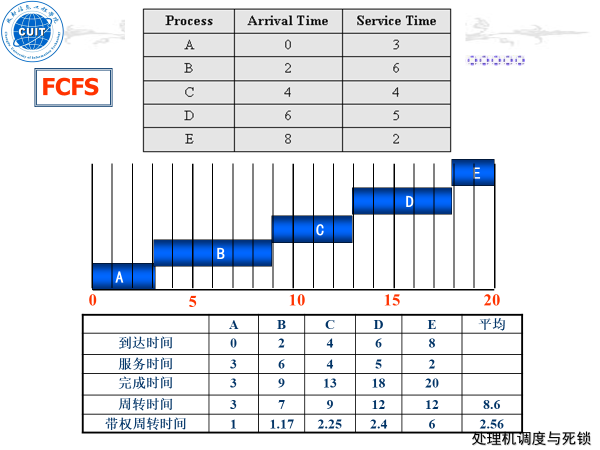
中级调度(中程调度)

低级调度(进程调度)

运行频率：低>中>高

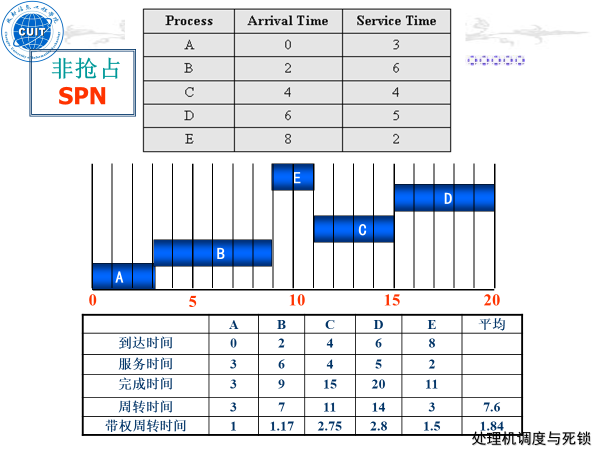
调度算法：

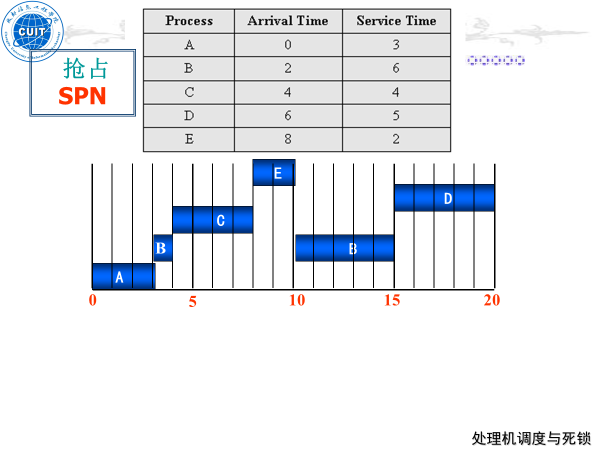
先来先服务调度算法：利于长作业、利于CPU繁忙型的作业



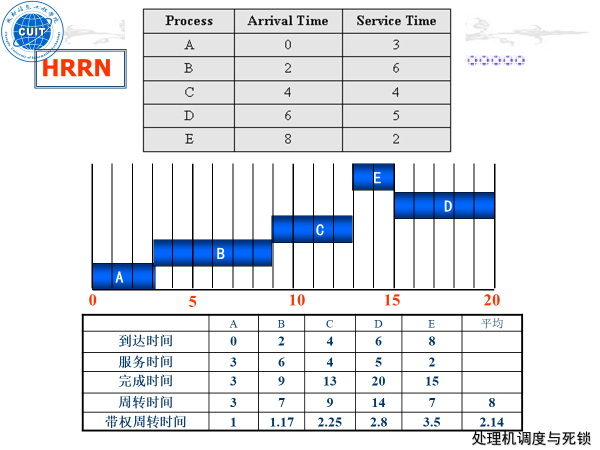
最短作业(进程)优先调度算法：抢占式、非抢占式，降低作业的平均等待时间和提高系统吞吐量

缺点：对长作业不利、不能保证紧迫性作业或进程会得到及时处理、不一定能真正做到短作业优先

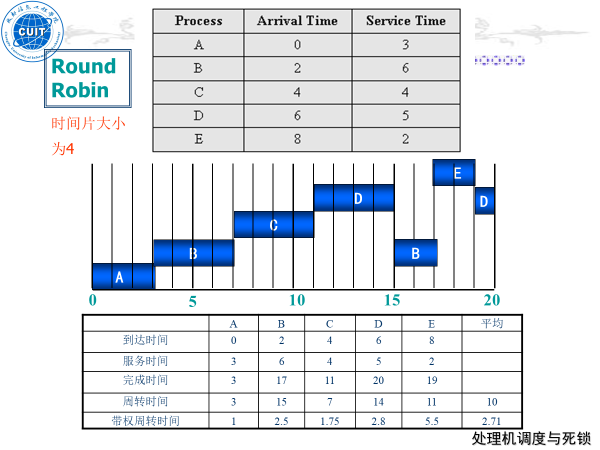




响应比Rp=(等待时间+服务时间)/服务时间



时间片轮转法：



死锁：当某进程提出资源申请后，使得系统中一些进程处于无休止的阻塞状态，在无外力作用下，永远不能再继续前进

死锁的起因：竞争资源(系统资源不足)、进程推进顺序非法

竞争资源引起的死锁：可剥夺性资源(CPU、主存)、非剥夺性资源(打印机、磁带机)、竞争非剥夺性资源

产生死锁的必要条件：互斥条件、请求和保持、不剥夺条件、环路条件

处理死锁的基本方法：预防死锁(破坏4个条件之一)、避免死锁、检测死锁、解除死锁

预防死锁：

1.互斥条件是资源固有属性，不能避免

2.摒弃请求和保持条件

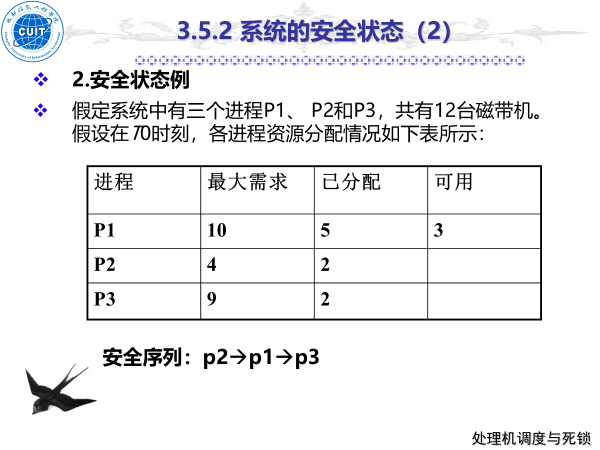
3.摒弃“不剥夺”条件

4.摒弃“环路等待”条件

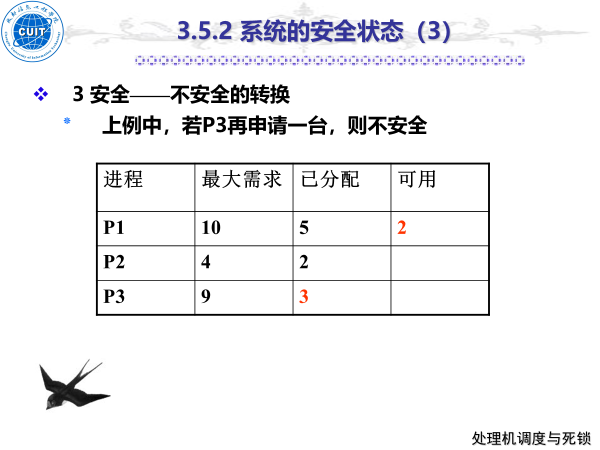
避免死锁：

1.安全状态

2.安全状态例



3.安全——不安全的转换



存储管理即存储器管理，它既包括内存管理也包括外存资源管理

CPU 最快

主存 快

磁盘(虚拟内存) 慢

地址空间：

物理地址空间：硬件所支持的地址空间

起始地址0到max

物理地址也称绝对地址或实地址

逻辑地址空间：一个运行的进程所拥有的内存范围

起始地址0到max

逻辑地址也称相对地址或虚地址

程序的装入：绝对装入方式(只使用单道)、可重定位装入方式、动态运行时装入方式

程序的连接：静态链接方式、装入时动态链接、运行时动态链接

连续分配方式：

单一连续分配：用于单用户、单任务

固定分区分配：分区大小相等、分区大小不等(方法)

动态分区分配：空闲分区表、空闲分区链(数据结构)

首次适应算法：每个空白区按地址递增的顺序链接在一起

循环首次适应算法

最佳适应算法：空白区按大小递增的顺序链在一起

最差适应算法：空白区按容量递减次序排列

对换：

将阻塞进程，暂时不用的程序，数据换出

将具备运行条件的进程换入

类型：整体对换、部分对换

基本分页存储管理：连续分配引起——碎片

物理内存划分成固定大小的存储块(帧)

逻辑地址空间划分成若干大小相等的页

命中率是指在地址变换工程中，在快表中(或内存等)查找成功的次数占总的访问页面数的比例

分段存储管理方式：方便编程、信息共享、信息保护、动态增长、动态链接

分段：每个段定义了一组逻辑信息，主程序段、子程序段、数据段等

段表、地址变换机构

可重入代码(Reentrant Code)又称为“纯代码”(Pure Code)是一种允许多进程同时访问的代码

常规存储器：一次性、驻留性

局部性(时间局部性、空间局部性)

虚拟存储器：多次性、对换性、虚拟性

分页请求系统(硬件支持、实现请求分页的软件)

请求分页存储器管理方式：

请求分页中的硬件支持：页表机制、缺页中断机制、地址变换机构

内存分配策略和分配算法：

最小物理块数的确定：保证进程正常运行所需的最小物理块数

物理块的分配策略

物理快分配算法：平均分配算法、按比例分配算法、考虑优先权的分配算法

调页策略：

1.何时调入页面：预调页策略、请求调页策略

2.从何处调入页面

3.页面调入过程

最佳置换算法：



先进先出置换算法(FIFO)：



最近最久未使用置换算法(LRU)：



设备是指计算机系统中除CPU，内存和系统控制台以外的所有设备

设备管理程序是用于对这类设备进行控制和管理的一组程序

设备驱动程序是I/O进程与设备控制器之间的通信程序

I/O设备的类型：

按传输速率分类：低速设备、中速设备、高速设备

按信息交换的单位分类：块设备、字符设备

按设备的共享属性分类：独占设备、共享设备、虚拟设备

按使用特性分类：存储型设备、输入型设备、输出型设备、输入输出型设备

按从属关系：系统设备、用户设备

设备控制器：

功能：接收CPU命令，控制I/O设备工作，解放CPU

1.接收和识别命令

2.数据交换

3.设备状态的了解和报告

4.地址识别

5.数据缓冲

6.差错控制

组成：



数据传送(I/O)控制方式：

选择和权衡控制方式的原则：速度高且安全、系统开销小、使I/O设备尽量忙，CPU等待时间少

四个阶段：程序I/O——中断I/O——DMA控制——通道控制(提高并行度)

程序直接控制(程序I/O)方式：控制简单硬件要求少

缺点：CPU与外围设备只能串行工作、设备间不能并行操作、安全性差

中断I/O：CPU不必反复测试寄存器状态，节约了时间

缺点：中断控制方式仍然消耗大量的CPU时间，因为每个字的数据传输都必须经过CPU寄存器转发

DMA：在外围设备和内存之间开辟直接的数据交换通路。信息的传送方向，传送的源地址和目的地址及传送长度都由CPU控制。

特点：数据传输的基本单位是数据块、所传送的数据是从设备直接送入内存的，或者相反、仅在传送一个或多个数据块的开始和结束时，才需CPU干预，整块数据的传送是在控制器的控制下完成的

通道控制方式：CPU只需发送I/O命令给通道，通道通过调用内存中的相应通道程序完成任务

通道类型：字节多路通路、数组选择通道、数组多道通道

缓冲技术：

引入原因：提高CPU，通道，I/O设备的并行能力、改善CPU和I/O设备间速度不匹配的情况、可减少中断CPU的次数，放宽CPU对中断的响应时间

循环缓冲：用几个缓冲区组成一个环形缓冲区，供输入/出共同使用

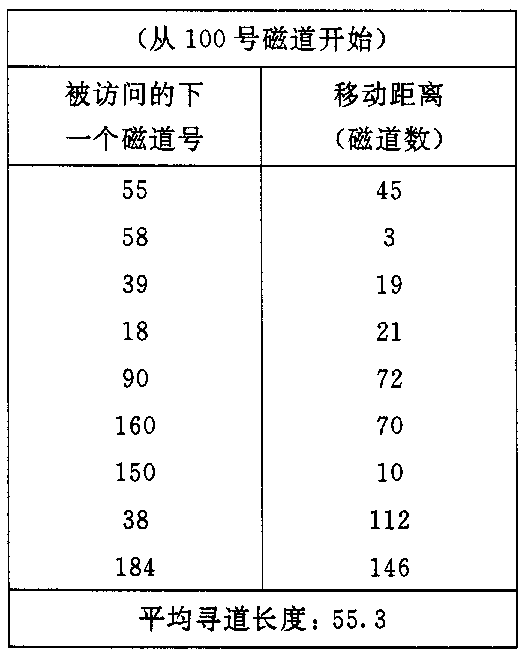
缓冲池：把输入和输出多缓冲器结合起来，形成一个既能用于输入，又能用于输出的缓冲器

SPOOLING技术：假脱机技术，在联机情况下同时出现外围操作一道程序，来模拟脱机输入时的外围控制机功能，把低速I/O设备上的数据传送到高速磁盘上；再用另一道程序来模拟脱机输出时外围控制机的功能，把数据从磁盘传送到低速输出设备上

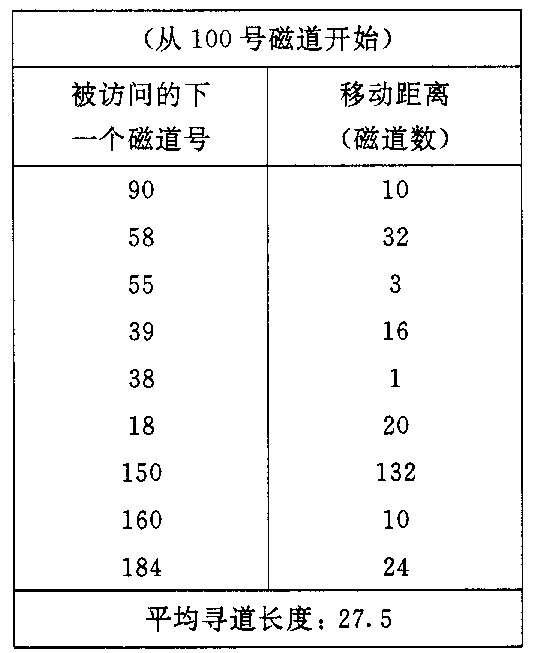
作用：通过缓冲方式，将独占设备改造为共享设备

组成：输入井、输入缓冲区、输入程序

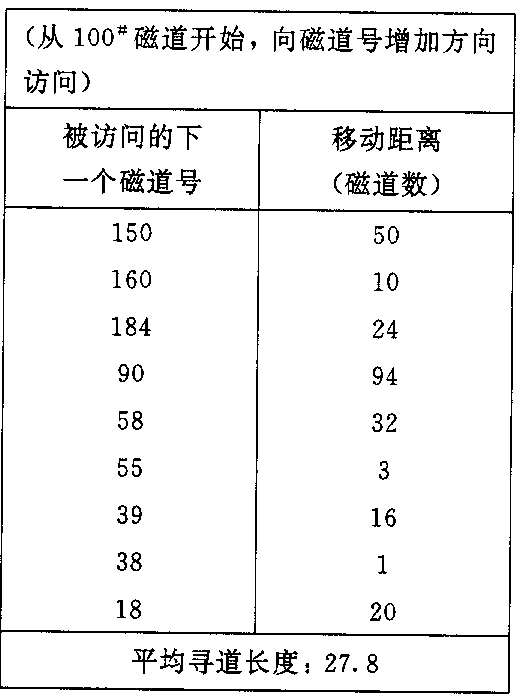
先来先服务FCFS：公平、简单、平局寻道时间长



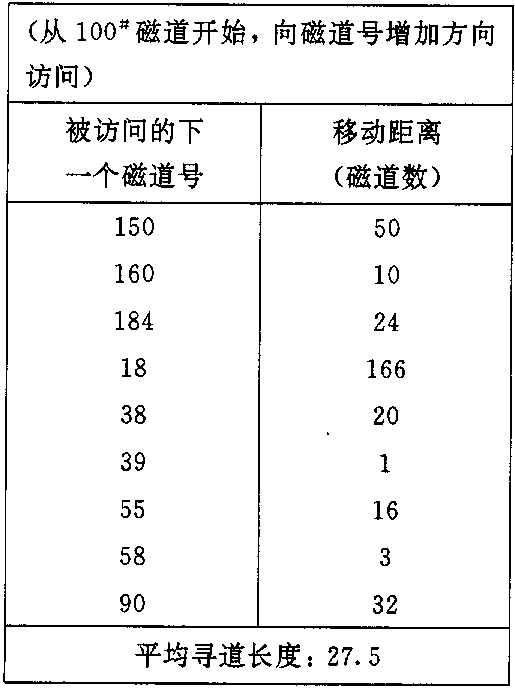
最短寻道时间优先SSTF：寻道时间最短，但导致某些进程发生“饥饿”现象



扫描(SCAN)算法：寻道性能较好，避免了进程“饥饿”现象



循环扫描算法(CSCAN)：规定磁头单向移动



程序和数据以文件的形式保留着在外存中

数据项(最低级的数据组织形式)：

基本数据项：又称为数据元素或字段

组合数据项

记录：是一组相关数据项的集合，用于描述一个对现在该某方面的属性

文件：有创建者所定义的、具有文件名和一组相关元素的集合

分为：有结构文件(记录式文件)、无结构文件(流式文件)

属性可以包括：文件类型、文件长度、文件的物理位置、文件的建立时间

文件类型：

按用途分类：系统文件、用户文件、库文件

按文件中的数据的形式分类：源文件、目标文件、可执行文件

按存取控制属性分类：只执行文件、只读文件、读写文件

(按文件的物理结构：顺序文件、链接文件、索引文件、散列(Hash)文件、索引顺序文件

按文件的存取方式：顺序存取文件、随机存取文件

按文件内容：普通文件、目录文件、特殊文件)

文件系统模型：文件和对文件进行操纵和管理的软件集合(文件->文件操作->文件访问接口)

对象及其属性：文件、目录、磁盘(磁带)存储空间

对对象操纵和管理的软件集合：对文件存储空间的管理、对文件目录的管理、用于将文件的逻辑地址转换为物理地址的机制、对文件读和写的管理，以及对文件的共享与保护等功能

文件系统的接口：命令接口、程序接口

文件操作：

对文件最基本操作：创建、删除、读、写、截断、设置文件的读写位置

对文件“打开”和“关闭”

逻辑结构类型：

有结构文件(记录式文件)：定长记录、变长记录

形式：顺序文件、索引文件、索引顺序文件

无结构文件(流式文件)：字节为单位，利用读/写指针进行访问

顺序文件：

逻辑记录的排序：串结构、顺序结构

对顺序文件的读/写操作

外存分配方法(文件物理组织)：文件的物理结构，又称为文件的存储结构，是指文件在外存上的存储组织形式

外存分配方式：连续分配、链接分配、索引分配(每个文件分配一个索引块)

文件共享：是指不同用户可以共同使用某文件

绕道法、连访法(链接法)、基本文件目录表

设备：是指计算机系统中除CPU，内存和系统控制台以外的所有设备

UNIX系统中包含两类设备：块设备(磁盘、光盘等)、字符设备(键盘、打印机)

缓冲是一个广泛采用的技术：单缓冲、双缓冲、多缓冲、缓冲池

缓冲的目的是为了信息交换双方的速度匹配