操作系统

## 第一章 操作系统概论：

计算机系统

计算机系统是一种可以按用户的要求接收和存储信息，自动进行数据处理并输出结果的 系统。

组成：

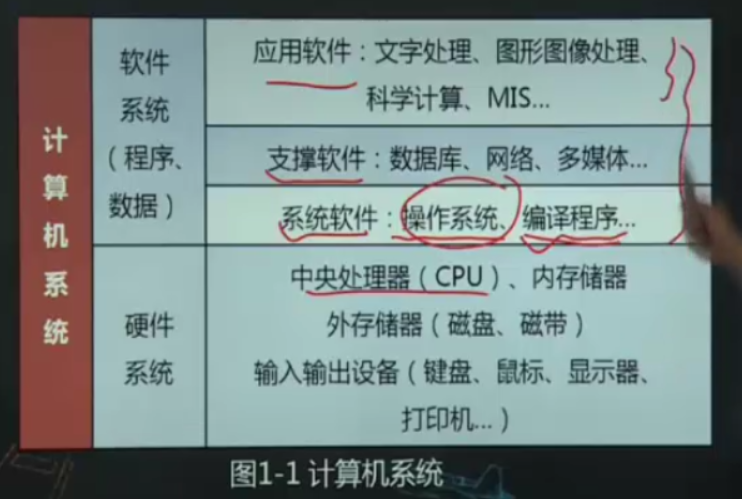
硬件子系统 软件子系统

操作系统:

在计算机系统中，集中了资源管理和程序控制功能的一种软件，成为操作系统。

计算机系统的资源：

硬件资源；软件资源。



1. 操作系统的定义：
2. 管理

组织和管理硬件资源和软件资源（表格登记，比如PCB,系统设备表）

有效

系统运行的效率，资源的利用率，CPU的利用率

合理公平

操作系统要 公平 对待不同的用户程序，保证系统不发生 死锁 和 饥饿 的现象

使用方便

易用性，易学性，易维护性

1. 操作系统的特征：

并发性：

计算机同时存在多个运行的程序，需要OS管理和调度

并发：在一个时间段内有多个程序运行(一个进程内多个线程)

并行：在一个时间点上有多个程序运行(多核CPU)

共享性：

操作系统需要与多个用户程序公用系统中的各种资源。

“同时” 访问

互斥共享

随机性：不可预知性

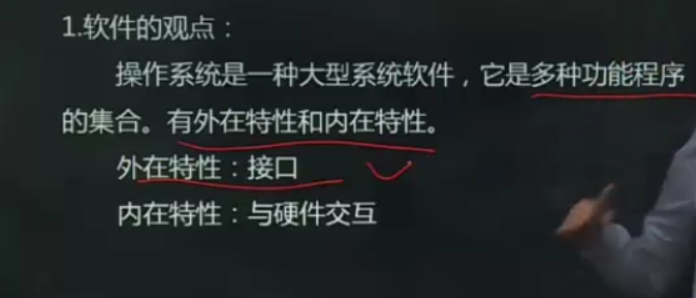
操作系统不能对所运行的程序的行为以及硬件设备的情况做出任何事先的假定。

程序的执行不是一贯到底，而是走走停停，向前推进的速度不可预知。异步的执行 过程。

但只要运行环境相同，OS需要保证程序运行的结果也要相同

1. 研究操作系统的观点：

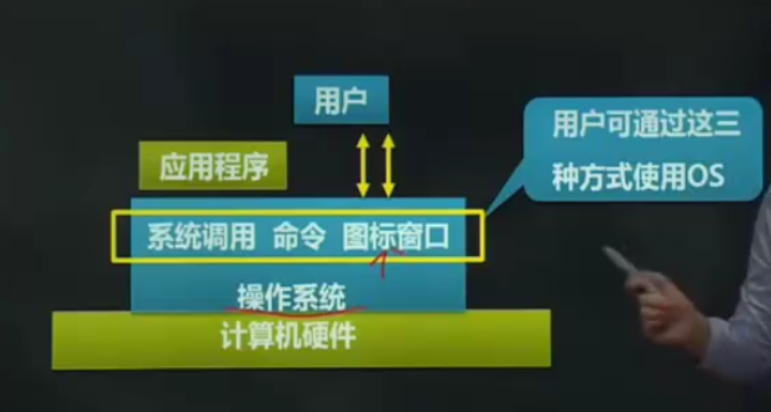
软件的观点：



硬件之上，应用程序之下。

外壳(Shell)，内核(Kernel)

外在特性：接口；内在特性：与硬件交互



操作系统内部组件：

CPU调度器

物理内存管理

虚拟内存管理

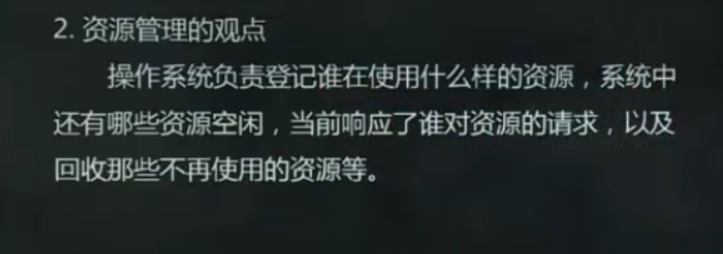
文件系统管理(disk)

中断处理与IO设备驱动

资源管理的观点：

操作系统通过多种方式对资源进行管理

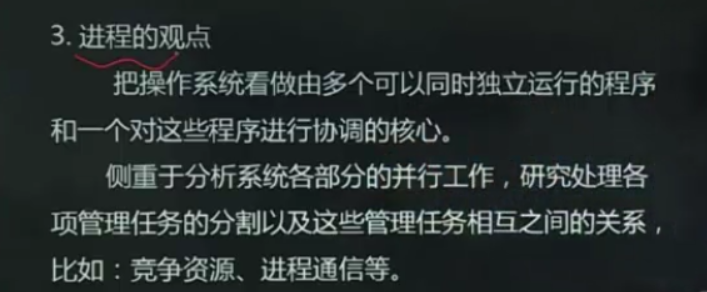
管理外部设备，分配资源



进程的观点：

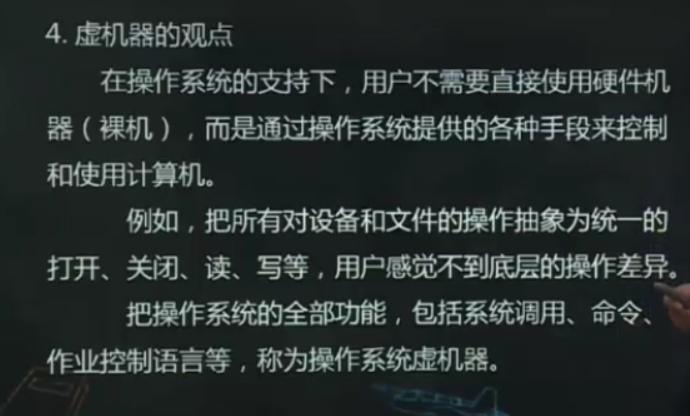
多个同时独立运行的程序和一个对这些程序进行协调的核心。

分析并行工作，竞争资源，进程通信

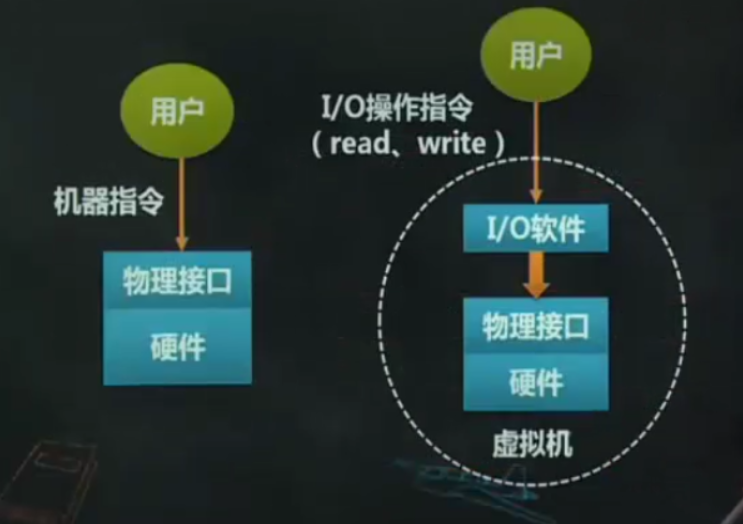


虚拟机的观点：

通过操作系统提供的各种手段来控制和使用外设。

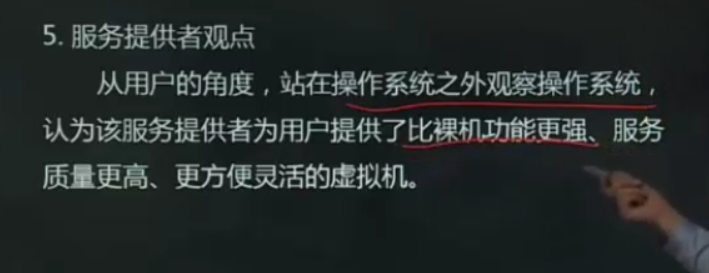


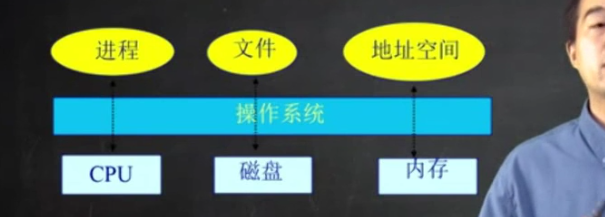
Windows操作系统的体系结构：



服务提供的观点：

从用户的角度，站在操作系统之外





1. 操作系统的功能：

进程管理：

对中央处理器进行管理

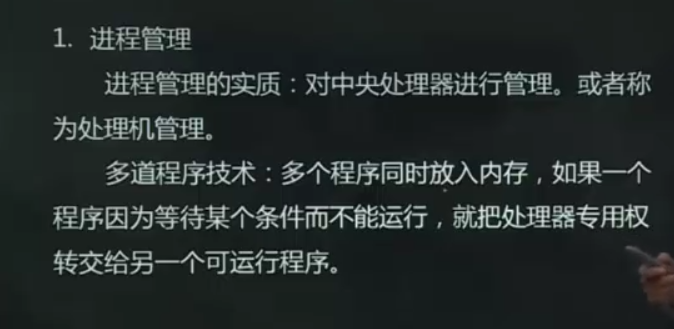
多道程序技术：

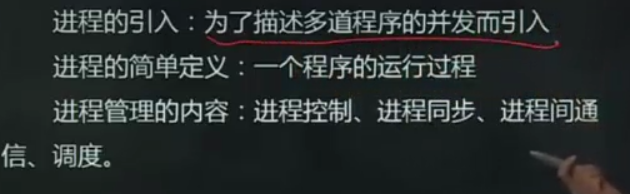
多个程序同时放入内存，如果某个程序因为等待某个条件而不能运行，就把处 理器使用权转交给另一个程序。

进程的引入：为了描述多道程序的并发而引入。PCB（进程控制块）？

进程的简单定义：

进程管理的内容：进程控制，进程同步，进程间通信，调度。





存储管理：

任务：管理计算机内存的资源

功能：

内存的分配与回收：

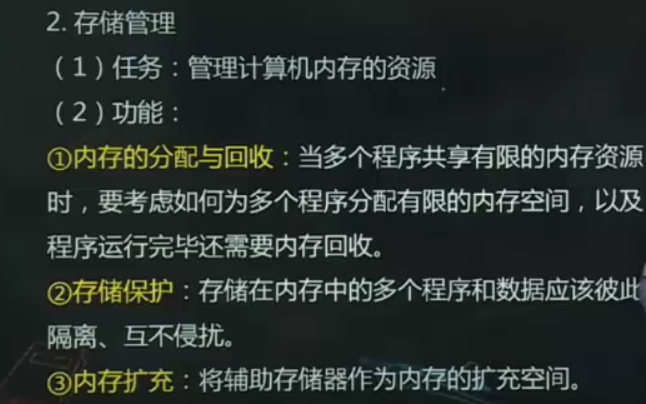
当多个程序共享有限的内存资源时，要考虑如何为多个程序分配有限的内 存空间以及程序运行完毕后还需要内存回收。

存储保护：彼此隔离、互不干扰

存储在内存中的多个程序和数据应该彼此隔离，互不侵扰。

内存扩充：

将辅助存储器作为内存的扩充空间



文件管理：

任务：

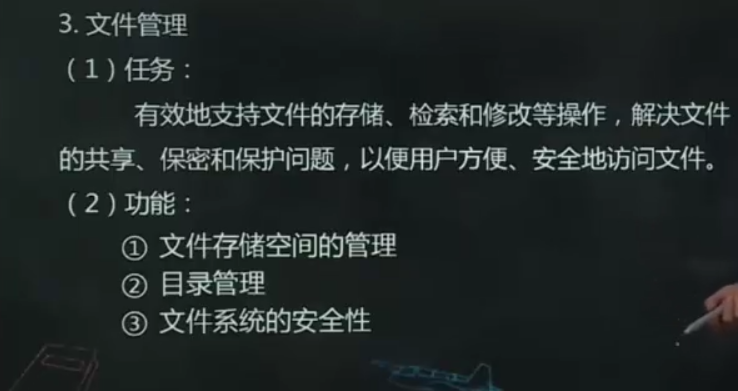
有效地支持文件的存储，检索和修改。解决文件的共享，保密和保护问题，以便用户方便，安全地访问文件系统。

功能：

存储空间管理

目录管理

文件系统的安全性

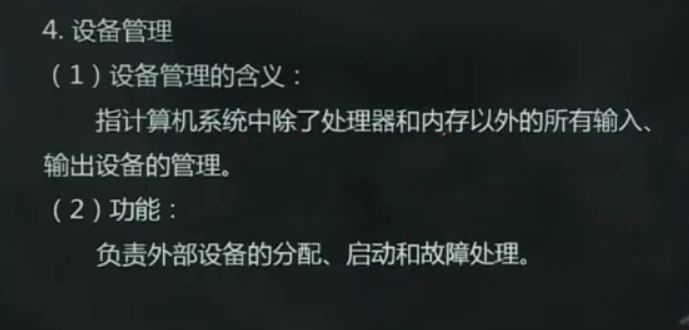


设备管理：

含义：除了处理器和内存以外的所有输入，输出设备的管理。

功能：负责外部设备的分配、启动和故障处理

采用的技术：中断技术、通道技术、虚拟设备技术、缓存技术。尽可能发挥设备和 主机的并行能力

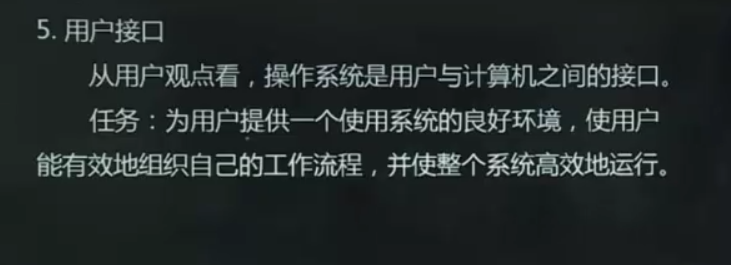


用户接口(从用户观点)：

操作系统是用户与计算机之间的接口

任务

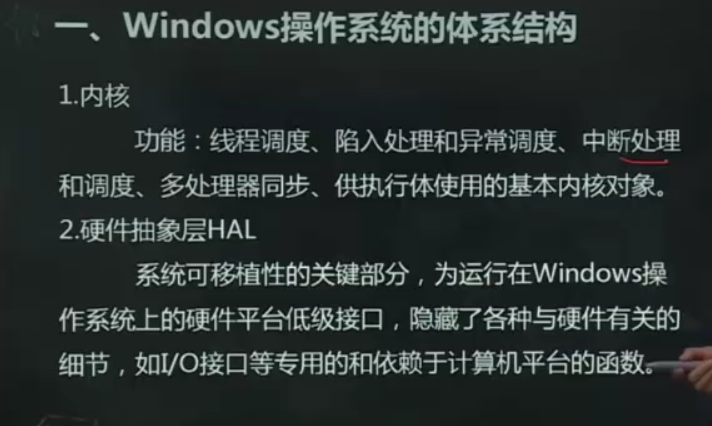
为用户提供一个使用系统的良好环境，使用户能够有效地组织自己的工作流程，并使整个系统高效运行。

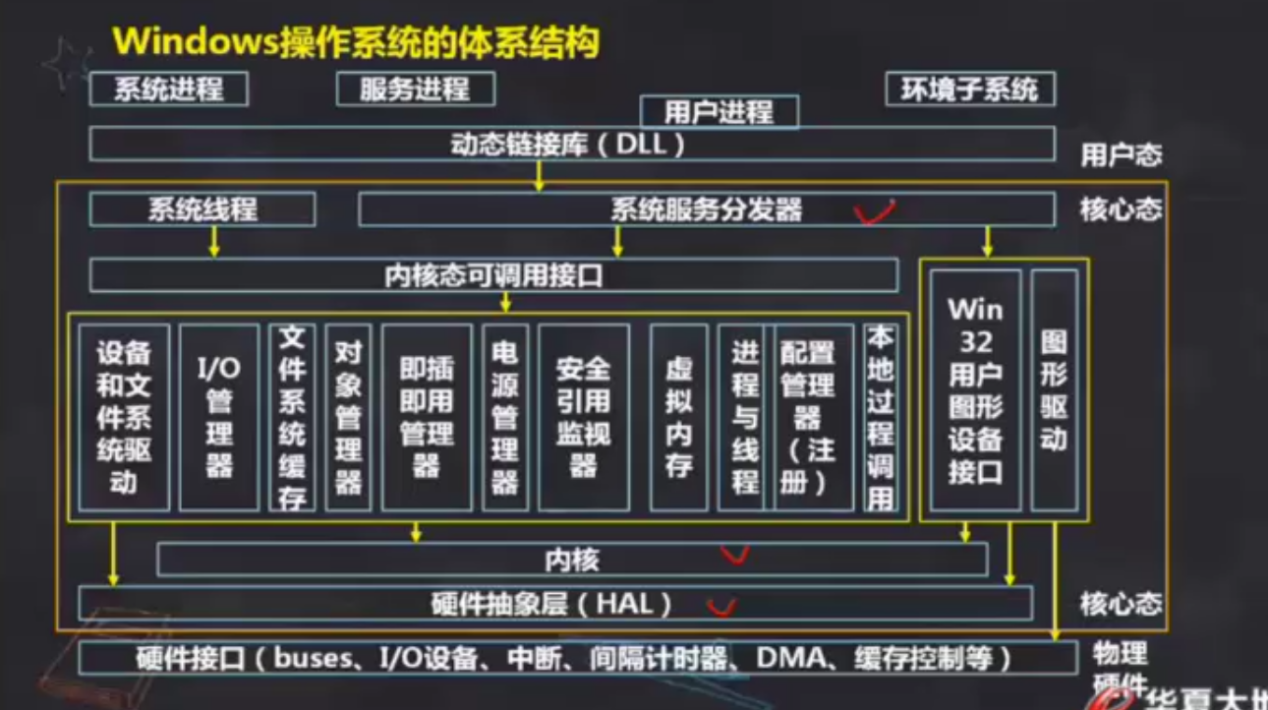


### 第二节：操作系统的体系结构

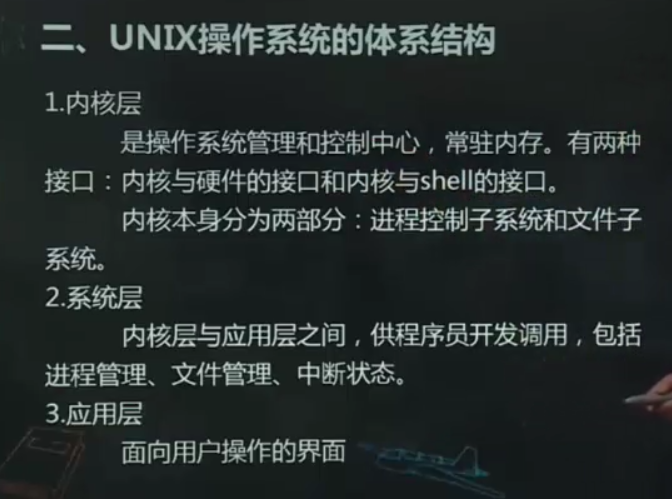
一、Windows OS体系结构

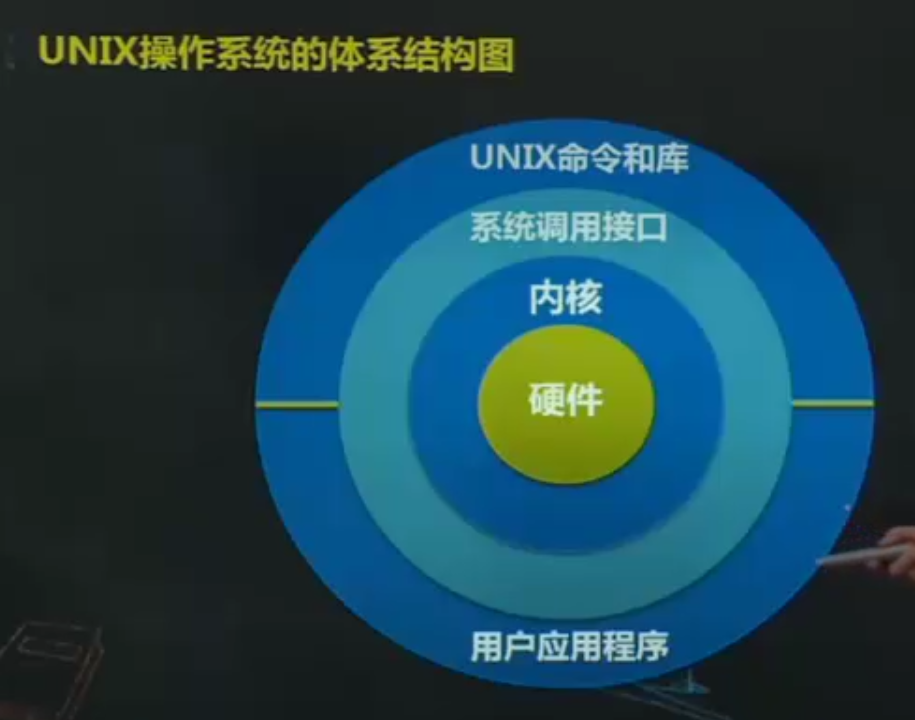
1. 内核
2. 硬件抽象层HAL
3. 执行体
4. 系统进程和系统线程



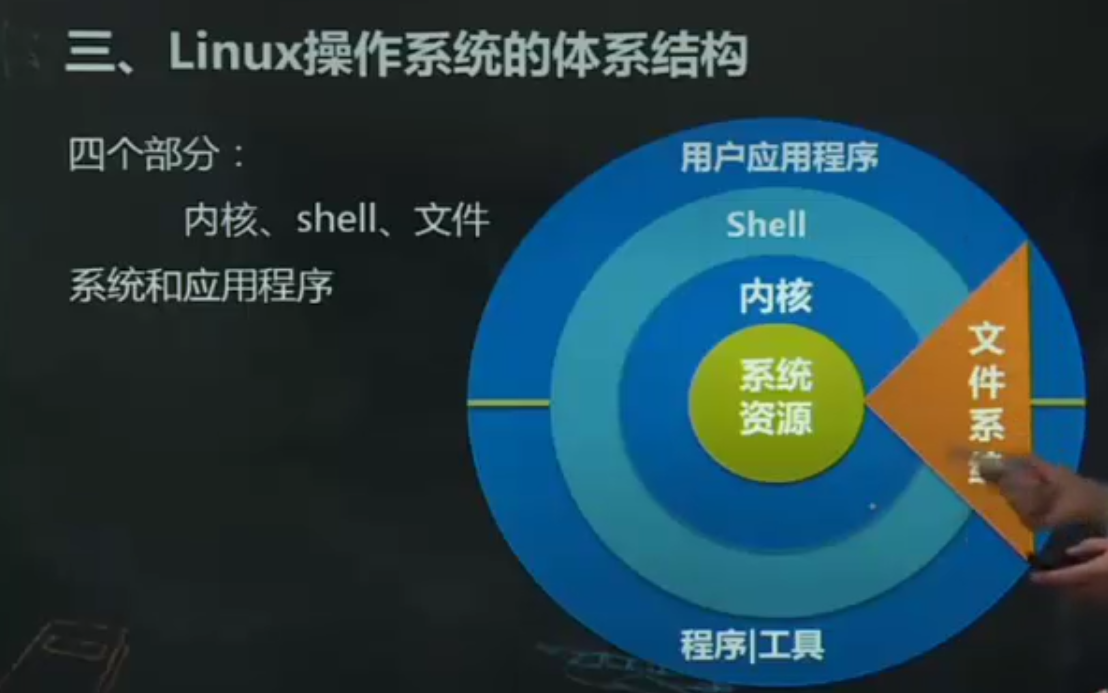


1. UNIX OS体系结构
2. 内核层
3. 系统层
4. 应用层

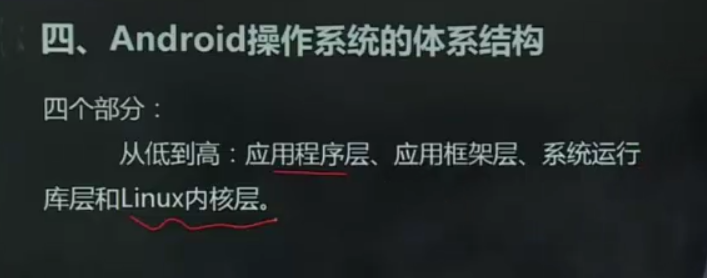




1. Linux OS体系结构
2. 内核
3. Shell
4. 文件系统
5. 应用程序



1. Android OS 体系结构



### 第三节：操作系统的发展

1. 手工阶段
2. 监控程序
3. 多道批处理
4. 分时与实时操作系统
5. UNIX通用操作系统
6. 个人计算机操作系统
7. Android操作系统：从低到高：应用程序层、应用框架层、系统运行库层、Linux内核层。

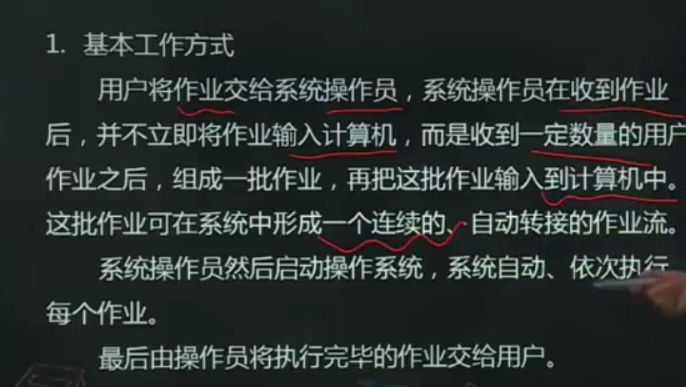
### 第四节：操作系统的分类(重点)

三种基本类型：

批处理系统，分时系统，实时系统

1. 批处理系统

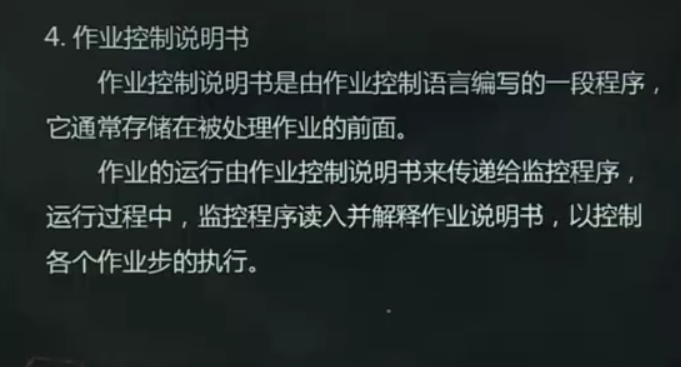
基本工作方式：生成作业，依次各个执行作业。



1. 特点：成批处理。用户不能干预自己作业的运行
2. 目标：资源利用率高，作业吞吐率高
3. 分类：简单批处理，多道批处理



1. 作业控制说明书：作业控制语言编写的程序



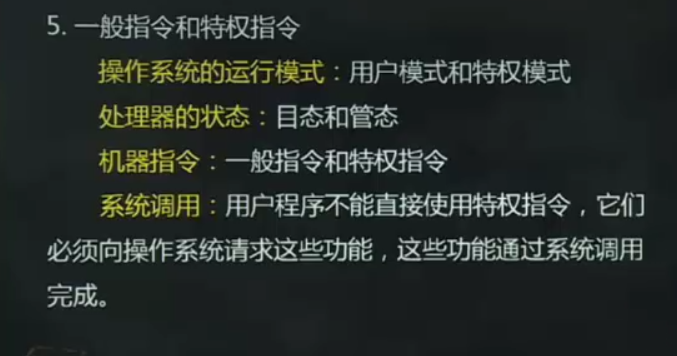
1. 一般指令和特权指令：

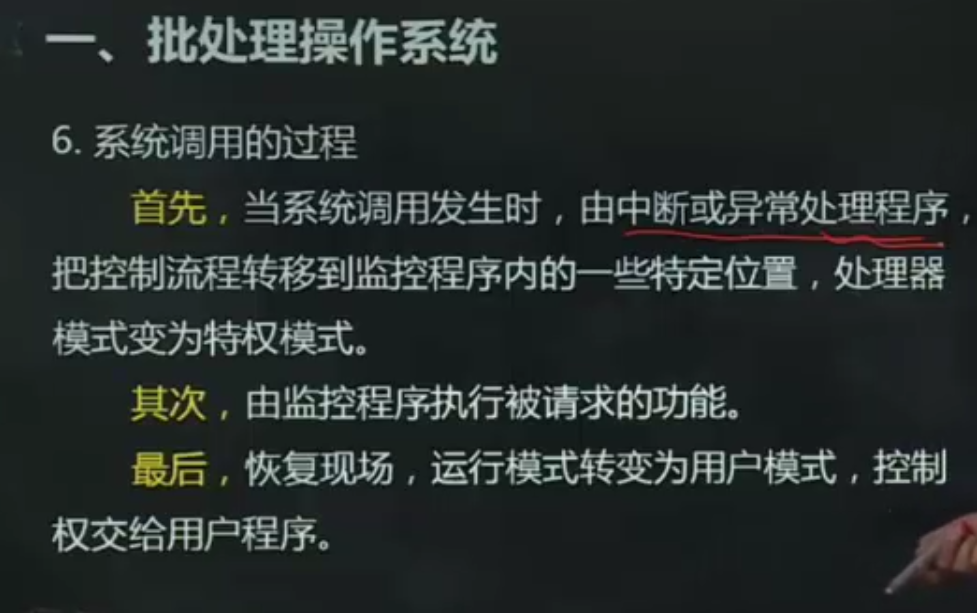
运行模式：用户模式，特权模式

处理器状态：目态(用户态)、管态(核心态)

机器指令：一般指令和特权指令

系统调用：用户程序不能直接使用特权指令，必须通过系统调用

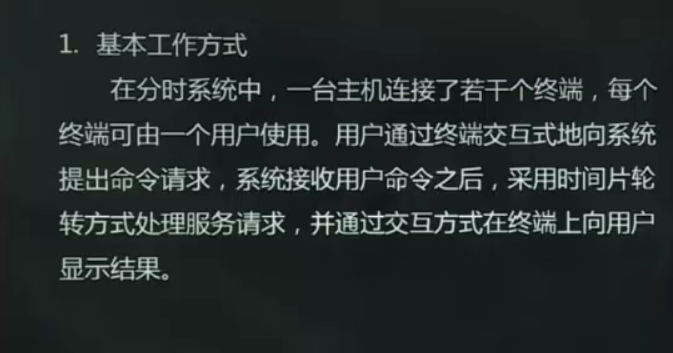


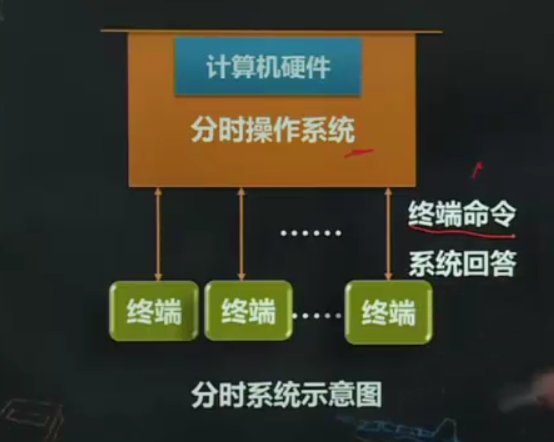


SPOOLing技术：

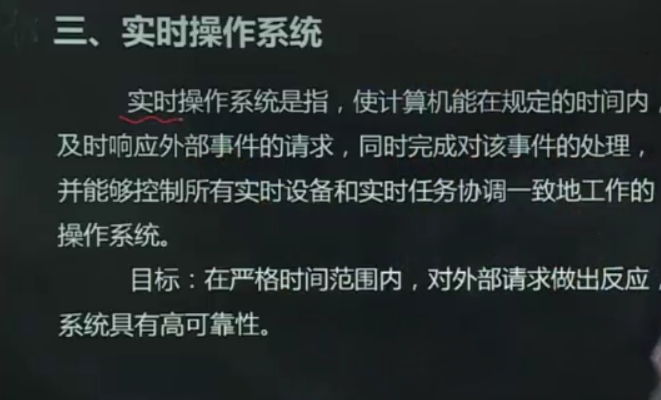
是多道程序设计的关键技术之一，也成为假脱机技术。

1. 分时系统
2. 基本工作方式：





1. 特点：多路性、交互性、独占性（分时轮流使用计算机资源）、及时性
2. 能力
3. 实时系统



1. 概念：
2. 目标：实时性，高可靠性
3. 分类：硬实时系统和软实时系统
4. 能力：

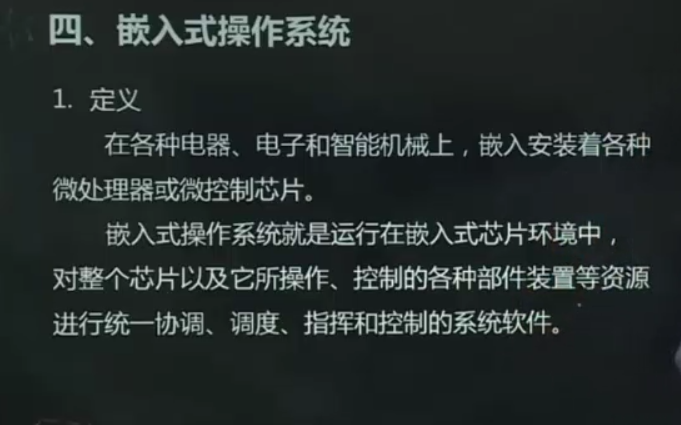
除了多道程序的基本能力外，还有以下功能：

实时时钟管理

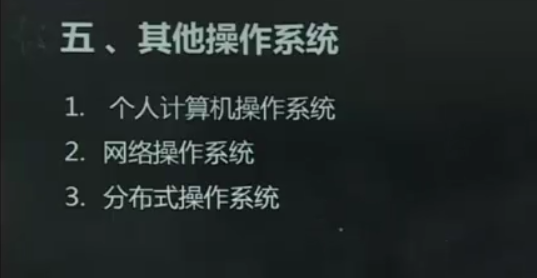
过载防护

高可靠性

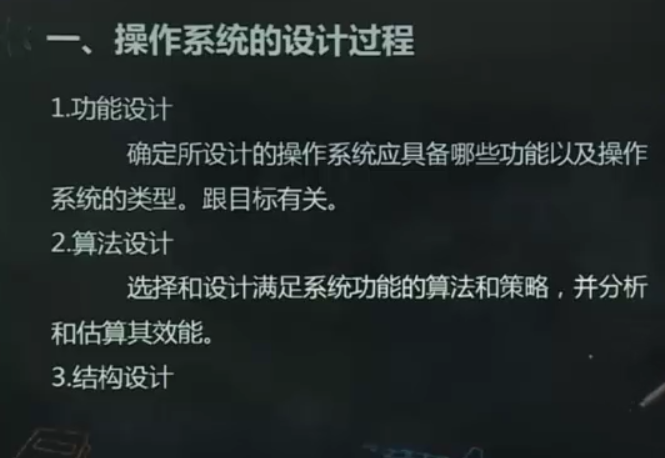
1. 嵌入式操作系统



1. 定义：
2. 特点：微型化、实时性



### 操作系统的设计过程



1. 功能设计

确定所设计的操作系统应具备哪些功能以及操作系统的类型。跟目标有关。

1. 算法设计

选择和设计满足系统功能的算法和策略，并分析和估算其效能。

1. 结构设计

结构研究目标：系统模块化，模块标准化，通信规范化

常见的操作系统结构：

整体式结构

层次性结构

微内核（客户/服务器）

操作系统的设计目标

1. 可靠性
2. 高效性
3. 易维护性
4. 可移植性
5. 安全性
6. 简明性