



全国本科计算机应用创新型人才培养规划教材

# 操作系统实用教程

主 编 范立南 刘 颖



北京大学出版社  
PEKING UNIVERSITY PRESS

全国本科计算机应用创新型人才培养规划教材

# 操作系统实用教程

主 编 范立南 刘 飒



北京大学出版社  
PEKING UNIVERSITY PRESS

## 内 容 简 介

操作系统是现代计算机系统中必不可少的基本软件，也是计算机及其相关专业的必修课，更是从事计算机相关职业人员应该具备的基础知识。

本书的主要内容包括：操作系统概述、Linux 操作系统简介、进程管理、进程控制、处理器调度、死锁、存储管理、设备管理、外存管理和文件系统。

本书深入浅出地对操作系统进行了描述，而且引入各种应用实例对基本原理进行讲解，同时对 Linux 操作系统的部分实现方法给予介绍。

本书可作为计算机专业或相关专业操作系统课程的教材，也可供有关科技人员自学或参考使用。

### 图书在版编目(CIP)数据

操作系统实用教程/范立南, 刘飒主编. —北京: 北京大学出版社, 2010.5

(全国本科计算机应用创新型人才培养规划教材)

ISBN 978-7-301-17087-8

I. 操… II. ①范…②刘… III. 操作系统—高等学校—教材 IV. TP316

中国版本图书馆 CIP 数据核字(2010)第 053140 号

书 名: 操作系统实用教程

著作责任者: 范立南 刘 飒 主编

策 划 编 辑: 张荣琴

责 任 编 辑: 程志强

标 准 书 号: ISBN 978-7-301-17087-8/TP · 1102

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电 子 邮 箱: [pup\\_6@163.com](mailto:pup_6@163.com)

印 刷 者:

发 行 者: 北京大学出版社

经 销 者: 新华书店

787mm×1092mm 16 开本 22 印张 510 千字

2010 年 5 月第 1 版 2010 年 5 月第 1 次印刷

定 价: 36.00 元

---

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有 侵权必究

举报电话: 010-62752024

电子邮箱: [fd@pup.pku.edu.cn](mailto:fd@pup.pku.edu.cn)

# 前 言

操作系统是配置在计算机硬件上的直接对硬件系统进行第一次扩充的软件，它在计算机中占据了极其重要的地位，其他所有系统软件及应用软件都依赖于操作系统的支持，都需要操作系统提供服务。

随着计算机体系结构的发展，计算机操作系统也得到了不断的发展。操作系统已从最初的监督程序，发展为现在熟悉的 DOS 磁盘操作系统、Windows 操作系统、Linux 操作系统和 UNIX 操作系统等，可以说新技术、新概念层出不穷。

只有掌握了操作系统的基本原理，才能理解和掌握各种操作系统的特性部分，才能深入了解操作系统内部特征，理解整个系统的工作流程是如何控制实现的、系统资源是如何分配和共享的、用户的任务是如何被接受和实现的，逐渐达到利用已掌握的知识分析、解决一般理论问题和实际问题，为进一步接受新型操作系统并应用操作系统基础开发自己的应用系统打下良好的基础。

本书结合编者多年的教学经验，依据教学大纲的要求编写而成。全书共 9 章，为了便于读者的学习，每章均附有习题，在附录中设置了实训练习。

本书第 1 章简要介绍了操作系统的简介、功能、分类和发展及实现。第 2 章主要介绍了 Linux 操作系统的简单使用方法。第 3、4 章介绍了进程管理和控制的有关概念与方法。第 5 章主要介绍了处理器调度和管理的策略。第 6 章介绍了死锁的基本概念、原因和解决办法。第 7 章介绍了存储管理，包括分区、分页、分段和段页式存储管理以及虚拟存储器的相关知识。第 8 章讲述了设备管理技术。第 9 章介绍了外存管理和文件系统。

本书第 1、7、8 章由范立南教授编写，第 2~6 章、第 9 章及附录由刘飒编写。

本书的讲授学时可以为 48~64 学时，各章分配如下：第 1 章 2~4 学时，第 2 章 2 学时，第 3 章 4~6 学时，第 4 章 8~10 学时，第 5 章 4~6 学时，第 6 章 6~8 学时，第 7 章 10~12 学时，第 8 章 6~8 学时，第 9 章 6~8 学时。教师可以根据教学计划自行安排学时。

由于编者水平有限，书中难免有不妥之处，恳请广大读者指正。

编 者  
2010 年 2 月

# 序

本套教材经过全国几十所高等学校老师一年多的努力，终于与广大读者见面了。我相信，它一定会受到全国高等学校计算机界老师和同学们的热烈欢迎。

随着信息技术的飞速发展，单一培养模式已经不能满足社会对计算机专业人才多样化的需求。应对这一变化的最佳办法，就是采用多种模式的培养方式。当前，高等学校的计算机教育正处于从过去的单一培养模式向多种培养模式的转变过程中，多种模式的培养方式将是必然的发展方向。

多种模式的培养方式包括：培养人才的类型不同(研究型，应用型)；专业方向不同(计算机软件，计算机网络，信息安全，信息系统，计算机应用技术等)；课程设置的多样性等。

同时，高等教育对科技人才培养的要求是：不但要培养研究型科技人才，还要为国家培养更多的应用型科技人才(或称工程型科技人才)。也就是说，培养应用型科技人才是百分之九十以上的普通高等学校的主要任务。

本套教材正是为适应多种模式培养方式的要求，并且着重于培养计算机领域高级应用型科技人才的需求，而组织编写的。

本套教材具有如下特点。

## 1. 基础理论够用

计算机专业所需的基础理论知识以够用为准，不是盲目扩张。如数字系统的基础知识，计算机的基本组成原理和体系结构的基础知识，离散数学的基础知识，数据结构和算法的基础知识，操作系统的基础知识，程序设计的基础知识等，都进行了必要的讲解介绍。

## 2. 强调理论联系实际，学以致用

每本教材的编写都将“理论联系实际，学以致用”的原则贯彻始终。例如，《计算机组成原理和体系结构》结合现代的计算机讲解，使学生学完之后，确切掌握现代计算机的组成、结构和工作原理；又如，《程序设计》结合实例讲解，使学生学完之后，真正能够动手编写程序。

## 3. 强调教材的配套性

根据多年组织教材的经验，只有配套性好的教材才最受教师和学生们的欢迎。我们这套教材，尽量做到了课堂教材、实训教材和教学课件完全配套，以方便教学使用。

另外，本套教材提供的是一套应用创新型计算机教育系列教材，可供不同类型学校依照自己的教学计划，根据自身的需要进行选用。

现在把这套教材奉献给全国计算机界的朋友们，真诚希望大家能够喜欢。本套教材难免会有诸多缺点或不到之处，还希望得到大家的批评和指正。

全国高等学校计算机教育研究会课程与教材建设委员会主任

李大友

2009年3月

# 目 录

<b>第 1 章 操作系统概述</b> .....	1	2.2.4 vi 的用法 .....	44
1.1 操作系统简介 .....	1	2.2.5 实例——Linux 运行一个用户 程序 .....	46
1.1.1 计算机系统层次结构 .....	1	本章小结 .....	51
1.1.2 操作系统的定义 .....	3	习题 2 .....	51
1.1.3 操作系统的发展 .....	3	<b>第 3 章 进程管理</b> .....	53
1.2 操作系统的功能 .....	6	3.1 并发控制 .....	53
1.2.1 操作系统在计算机系统 中的地位和运行环境 .....	6	3.1.1 多道程序设计 .....	53
1.2.2 操作系统的特征 .....	6	3.1.2 程序的并发执行 .....	54
1.2.3 操作系统的功能 .....	8	3.2 进程的定义和特征 .....	56
1.2.4 操作系统的结构设计模式 .....	9	3.2.1 进程的定义 .....	56
1.2.5 操作系统的设计规范 .....	11	3.2.2 进程的特征 .....	56
1.3 操作系统的分类和发展 .....	12	3.2.3 进程与程序的区别和联系 .....	57
1.3.1 操作系统的分类 .....	12	3.2.4 进程与作业的区别 .....	58
1.3.2 操作系统的发展 .....	16	3.2.5 操作系统与进程的控制执行 .....	58
1.3.3 操作系统的实例 .....	18	3.3 内核中进程的实现 .....	61
1.4 操作系统的实现 .....	21	3.3.1 进程控制块 PCB .....	61
1.4.1 操作系统开发与升级 .....	21	3.3.2 进程上下文 .....	62
1.4.2 操作系统的安装 .....	21	3.3.3 进程三态转换分析 .....	63
1.4.3 系统启动与关机 .....	22	3.3.4 进程的组织 .....	65
本章小结 .....	22	3.3.5 内核创建进程 .....	66
习题 1 .....	23	3.4 线程模型 .....	69
<b>第 2 章 Linux 操作系统简介</b> .....	26	3.4.1 多线程 .....	70
2.1 Linux 操作系统概述 .....	26	3.4.2 内核线程和用户线程 .....	70
2.1.1 Linux 操作系统的历史 .....	26	3.4.3 线程与进程的主要区别 .....	71
2.1.2 Linux 操作系统的特点 .....	27	3.5 Linux 的进程管理 .....	71
2.1.3 Linux 操作系统的设计原理 .....	29	3.5.1 Linux 进程的组成 .....	71
2.1.4 Linux 操作系统的内核模块 .....	31	3.5.2 进程描述符 .....	72
2.2 Linux 操作系统的使用 .....	33	3.5.3 进程状态 .....	72
2.2.1 Linux 系统引导 .....	33	3.5.4 进程链表 .....	73
2.2.2 文件存取权限和所有权限 .....	41	3.5.5 进程控制 .....	74
2.2.3 常用命令 .....	42		

3.5.6 Linux 线程.....	74	4.6.5 共享内存.....	114
3.5.7 Linux 进程的创建与撤销.....	75	本章小结.....	114
3.5.8 实例——使用库函数 clone() 与 pthread_create()在 Linux 环境下创建两种不同执行模式 的线程程序.....	77	习题 4.....	115
3.5.9 实例——Linux 操作系统下 守护进程的编程方法.....	80	<b>第 5 章 处理器调度</b> .....	118
本章小结.....	83	5.1 处理器调度的基本概念.....	118
习题 3.....	84	5.1.1 作业的状态及三级调度的 概念.....	118
<b>第 4 章 进程控制</b> .....	88	5.1.2 调度队列模型.....	120
4.1 同步和互斥.....	88	5.1.3 选择调度方式和算法的 若干原则.....	121
4.1.1 并发问题.....	88	5.2 调度算法.....	124
4.1.2 进程同步和互斥的基本概念.....	90	5.2.1 先来先服务和短作业(进程) 优先调度算法.....	124
4.1.3 信号量机制.....	92	5.2.2 高优先权优先调度算法.....	125
4.1.4 信号量的应用.....	93	5.2.3 基于时间片的轮转调度 算法.....	128
4.2 经典的进程同步问题.....	95	5.2.4 反馈调度算法.....	128
4.2.1 生产者—消费者问题.....	96	5.2.5 Linux 中的进程调度.....	130
4.2.2 哲学家进餐问题.....	97	5.2.6 实例——批处理操作系统中 作业调度.....	131
4.2.3 读者—写者问题.....	98	本章小结.....	134
4.3 管程机制.....	100	习题 5.....	134
4.3.1 管程的定义.....	100	<b>第 6 章 死锁</b> .....	139
4.3.2 利用管程解决进程之间的 同步与互斥.....	101	6.1 死锁的定义和死锁产生的必要条件.....	139
4.4 进程通信.....	102	6.1.1 死锁的定义.....	139
4.4.1 进程通信的类型.....	103	6.1.2 死锁产生的必要条件.....	142
4.4.2 消息缓冲通信.....	106	6.1.3 资源分配图.....	145
4.4.3 信箱通信机制.....	108	6.2 死锁的预防.....	146
4.5 线程间的同步和通信.....	109	6.3 死锁的避免.....	148
4.5.1 互斥锁.....	109	6.3.1 系统安全状态.....	148
4.5.2 条件变量.....	109	6.3.2 银行家算法.....	149
4.5.3 信号量机制.....	110	6.4 死锁的检测和恢复.....	153
4.6 Linux 进程通信机制.....	110	6.4.1 死锁的检测.....	153
4.6.1 信号.....	110	6.4.2 死锁的解除.....	154
4.6.2 管道.....	111	6.5 实例——银行家算法的模拟.....	155
4.6.3 消息队列.....	111	本章小结.....	158
4.6.4 信号量.....	113	习题 6.....	159

<b>第 7 章 存储管理</b> .....	164	8.1.3 I/O 通道 .....	229
7.1 存储管理概述 .....	164	8.1.4 总线系统 .....	230
7.1.1 存储管理的基本概念 .....	164	8.2 I/O 控制方式 .....	231
7.1.2 程序的装入和链接 .....	165	8.2.1 程序 I/O 方式 .....	231
7.1.3 重定位技术 .....	168	8.2.2 中断控制方式 .....	231
7.1.4 存储管理的目的和功能 .....	168	8.2.3 DMA 控制方式 .....	232
7.2 连续模式 .....	170	8.2.4 通道控制方式 .....	233
7.2.1 覆盖与交换技术 .....	170	8.2.5 中断技术 .....	235
7.2.2 单一连续区 .....	171	8.3 缓冲管理 .....	238
7.2.3 固定式分区 .....	172	8.3.1 缓冲技术的引入 .....	238
7.2.4 可变式分区和动态存储 分配技术 .....	174	8.3.2 缓冲区的分类 .....	239
7.3 页式管理 .....	180	8.4 设备分配 .....	242
7.3.1 基本的分页存储管理 .....	180	8.4.1 设备分配中的数据结构 .....	243
7.3.2 虚拟存储器 .....	186	8.4.2 设备分配的原则 .....	244
7.3.3 请求分页存储管理 .....	188	8.4.3 设备分配的独立性 .....	246
7.3.4 页面置换 .....	192	8.4.4 设备分配的流程 .....	247
7.4 分段管理 .....	198	8.4.5 SPOOLing 技术 .....	248
7.4.1 分段存储管理 .....	198	8.5 设备驱动程序 .....	250
7.4.2 段页式管理 .....	200	8.6 磁盘存储器管理 .....	251
7.4.3 请求分段存储管理方式 .....	201	8.6.1 磁盘调度策略 .....	252
7.5 实例——请求页式存储管理中 页面置换算法的模拟 .....	203	8.6.2 磁盘容错技术 .....	256
7.6 Linux 内存管理 .....	209	8.6.3 磁盘高速缓存 .....	260
7.6.1 Linux 的内存管理原理 .....	209	8.7 实例——独占设备的分配和 回收模拟 .....	261
7.6.2 i386 的页式主存管理机制 .....	210	8.8 Linux 设备管理 .....	265
7.6.3 Linux 主存管理的基本框架 .....	211	8.8.1 设备驱动程序环境 .....	265
7.6.4 存储器管理中的几个重要 数据结构 .....	212	8.8.2 设备文件 .....	268
7.6.5 主存页框的调度 .....	216	8.8.3 设备驱动程序概述 .....	268
7.6.6 缺页异常处理 .....	217	本章小结 .....	269
本章小结 .....	217	习题 8 .....	270
习题 7 .....	218	<b>第 9 章 外存管理和文件系统</b> .....	276
<b>第 8 章 设备管理</b> .....	226	9.1 文件的管理 .....	276
8.1 I/O 系统 .....	226	9.1.1 文件基本概念 .....	276
8.1.1 I/O 设备 .....	226	9.1.2 文件系统基本概念 .....	278
8.1.2 设备控制器 .....	227	9.1.3 文件的组织与存取 .....	279
		9.1.4 目录管理 .....	284
		9.1.5 外存空间空闲存储块的 管理 .....	290

9.1.6 文件的安全管理.....	294	9.2.4 对虚拟文件系统的管理.....	303
9.1.7 文件的系统基本模型.....	298	9.3 实例——文件系统模拟 .....	304
9.2 虚拟文件系统.....	301	本章小结 .....	313
9.2.1 虚拟文件系统的设计思路 .....	301	习题 9 .....	313
9.2.2 虚拟文件系统 VFS 框架.....	302	附录 实训练习 .....	318
9.2.3 Linux 虚拟文件系统的 数据结构.....	302	参考文献.....	340

# 第 1 章 操作系统概述

计算机系统由硬件和软件两部分组成。从功能上，可以将整个计算机系统划分为硬件、其他操作系统、系统软件和应用软件 4 个层次，操作系统(Operating System, OS)是配置在计算机硬件上的第一层软件，是对硬件系统的首次扩充，它在计算机系统中占据了极其重要的地位，其他所有的软件如汇编程序、编译程序、数据库管理系统等系统软件以及大量的应用软件，都依赖于操作系统的支持。

操作系统已经成为从大型机到微机都必须配置的软件。关于什么是操作系统，它是如何构成的，它在计算机中的地位是怎样的，操作系统的特征和功能又是什么，它有哪些类型，每种类型又有什么特点，这些问题都将在本章做相应的阐述。

## 1.1 操作系统简介

### 1.1.1 计算机系统层次结构

只有硬件组成的计算机称为裸机。对于一台裸机，无论它在硬件配置上有多高，性能有多强，也很难被使用。这是因为在裸机上操作，必须用机器指令编制程序，去控制计算机上的部件，用户不仅要用机器代码编写自己的应用程序，而且还要编写控制使用硬件的程序，这种程序代码动辄数千条，十分不方便，而操作系统的出现使得这一切变得简单起来。

操作系统是软件系统中最重要、最基本的系统，它为系统中的其他一切软件提供了与硬件配合完成计算机各项任务的友好界面。如图 1.1 所示，可见操作系统在整个计算机系统中所处的地位非常重要。

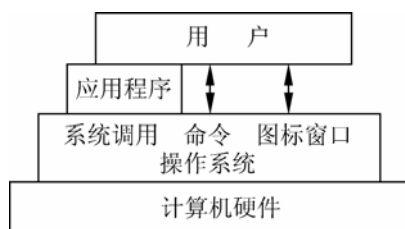


图 1.1 计算机系统的层次视图

由图 1.1 可以看出，其他软件(包括实用软件)以操作系统为支撑环境，换言之，操作系统为应用程序提供服务。操作系统紧贴裸机，把裸机改造成功能更强大、使用更方便的虚拟机。操作系统本身又分为内核和外核两部分。操作系统本身的活动在外核是并发的，而在内核通常是顺序执行的。例如，UNIX 的核心程序在执行期间，只要它自己不调用 sleep 原语就不会引起处理器的切换，这样做的好处是简化核心程序的编制。传统的内核包括中断处理、设备驱动、CPU 切换以及实施进程控制与通信等功能，这些功能由原语实现，内核微型化既能提高操作系统的并发程度又能使操作系统方便地被移植到不同芯片的计算机系统上，这就

是所谓的微核(Microkernel)技术。

近年来,大型软件都是采用层次式结构进行构建的,也就是将一个软件分成若干个层次。在此,同样可以用一个层次式的 OS 模型来描述操作系统,该模型分为 3 个层次(如图 1.2 所示),包括:用户接口(命令接口、程序接口、图形用户接口);对对象操纵和管理的软件集合(处理器管理软件、存储器管理软件、设备管理软件、文件管理软件);操作系统对象(处理器、存储器、设备、文件和作业)。

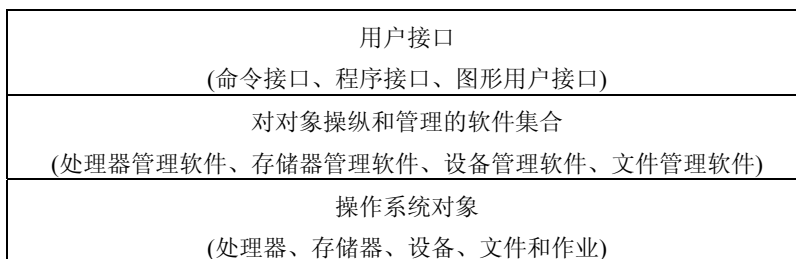


图 1.2 OS 层次模型

### 1. 用户接口

为了方便用户使用 OS, OS 通常向用户提供以下 3 种类型的接口。

(1) 命令接口,即用户与 OS 的接口。用户可以通过命令来取得 OS 的服务。

(2) 程序接口,即应用程序与 OS 的接口。用户通过在程序中进行系统调用来取得 OS 的相关服务。

所谓系统调用是操作系统提供给编程人员的唯一接口。编程人员可以利用系统调用,在源程序一级动态请求和释放系统资源,并调用系统中已有的系统功能来完成那些与机器硬件部分相关的工作以及控制程序的执行速度等。因此,系统调用对于用户而言,就像一个黑箱子那样,屏蔽了操作系统的具体动作而只提供有关的功能。事实上,人们使用计算机时接触到的命令控制界面(如菜单、命令按钮等所对应的操作命令),也是在系统调用的基础上开发成的。

(3) 图形用户接口。操作系统的设计者将系统常用的各项功能和应用程序,以各种直观的图形逼真地表示出来,方便用户操作。

### 2. 对对象操纵和管理的软件集合

对对象操纵和管理的软件集合部分通常是操作系统的核心部分,它们集中了操作系统五大功能中的四大功能,分别用于对 4 类对象进行操控和管理。此外,还包括了一部分对作业进行管理的功能。

### 3. 操作系统对象

操作系统对象是 OS 操纵和管理的对象。由于 OS 作为计算机硬件和软件间的界面,主要用于管理和调度各种系统资源,因此,OS 的对象主要有以下几类:处理器、存储器、各种 I/O 设备、文件和作业。

其中,用户接口在最外层,是用户与计算机之间的桥梁,而控制和管理的软件集合是整个操作系统的核心部分,也称之为操作系统内核(Kernel)。

## 1.1.2 操作系统的定义

当前,任何一种计算机都装有操作系统(OS)。计算机系统主要由硬件和软件两部分组成,计算机硬件主要包括中央处理器、主存储器、外围设备等物理设备,系统中的硬件配置可以因为机器使用用途的不同而有所不同。但要使其高效、高速地运行,就要对有限的 CPU 及主存资源充分利用。图 1.3 是计算机系统结构图,操作系统及其他软件放在磁盘上,要运行的时候放到内存里。

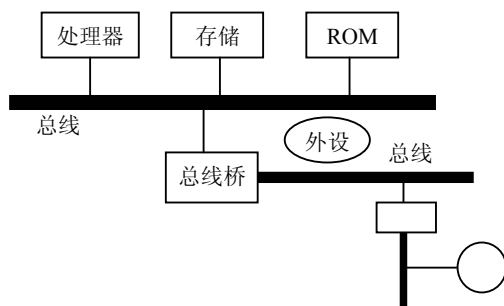


图 1.3 计算机系统结构图

为此,就需要协调、分配这些资源,所有这些问题的解决就构成了对计算机系统的管理、控制工程(在管理学上,就是追求  $1+1>2$ )。这些都是通过操作系统来实现的。

下面举个例子:一个学校的食堂相当于一台计算机,里面的所有物品相当于计算机的硬件,食堂中存在的人员相当于软件(食堂的自有人员为系统软件,其中的管理人员为操作系统,来食堂吃饭的人为应用软件)。原来可以负担 100 人午餐的食堂,通过加强管理,如科学的管理所有工作流程,尽量满足就餐人员的各项需求,缩短工作时间,减少空闲时间,加大各类资源利用率,提高并发性和共享性等,可以达到负担 200 人甚至更多人午餐。在此,操作系统起到了决定性的作用。

可见操作系统是计算机系统中的一个系统软件,是一些程序模块的集合,它们能以尽量有效、合理的方式组织和管理计算机的软硬件资源,为用户提供各种服务功能,使用户能够灵活、方便有效地使用计算机,使整个计算机系统能高效地运行。

狭义操作系统包含的程序有:核心态运行程序,用户态运行的命令解释器和系统调用库。广义操作系统包含的程序有:除上述外,许多提供系统常用功能的实用程序,库程序。因此,操作系统可以被定义为集如下三方面功能为一体的程序集合。

- (1) 控制和管理计算机系统的硬件与软件资源。
- (2) 合理地组织计算机的工作流程。
- (3) 方便用户使用。

管理目标:使硬件、软件资源的利用率最高。

服务宗旨:给用户尽可能多的服务和最大的方便。

服务项目:程序界面——由系统调用命令组成,给用户编程提供方便;联机用户界面——由键盘命令、屏幕命令组成;脱机用户界面——由作业控制命令实现。

## 1.1.3 操作系统的发展

伴随着计算机技术的发展,操作系统的功能也在不断地变化和完善。从计算机体系结构

的观点上讲,计算机的某项功能既可以用硬件实现,也可以用软件实现(用硬件实现的特点是速度快、价格高、可维护性差;用软件实现的特点是速度慢、价格低、可维护性高,因此,某项功能采用何种方式实现由性价比等因素决定)。随着计算机体系结构的发展,软硬件技术的变革,操作系统也在发生着日新月异的变革。具体地讲操作系统的发展经历了人工操作阶段、批处理阶段和操作系统的形成阶段。伴随着不断出现的实际功能需求的增长,操作系统还处于动态发展过程中。

### 1. 人工操作阶段

在第一代计算机时期,构成计算机的主要元件是电子管。计算机运行速度很慢,资源管理是用户自己编制的管理程序完成的,不能被称为操作系统(没有实现并发性和共享性,见 1.2 节)。用户都是直接用汇编语言编制程序,并在上机时独自占用全部系统资源,用户既是程序员,又是操作员,上机完全是手工操作,整个过程漫长,十分不便。

到了 20 世纪 50 年代后期,计算机运行速度有了明显的提高,手工操作方式严重降低了计算机资源的利用率,手工操作的慢速度和计算机的高速度之间形成了矛盾,即所谓的手工操作方式与机器利用率的矛盾,简称人机矛盾。随着 CPU 速度的提高、系统规模的扩大,人机矛盾也就变得日趋严重,只有摆脱人的手工操作,实现作业的自动过渡,才能解决此矛盾,这样就出现了批处理。

### 2. 早期的批处理

计算机发展的早期阶段,由于没有任何管理软件,所有的运行管理和具体操作都由用户自己来做,任何一步错误的操作都有可能造成任务的从头开始,当时的计算机价格又是非常的昂贵,尽可能提高 CPU 的利用率变得十分迫切。

为了解决上述的问题,批处理出现于 20 世纪 50 年代末到 60 年代初,它是为了提高主机的使用效率,在解决人机矛盾的过程中发展起来的。但是这时计算机系统运行的特征是单道顺序地处理作业(计算机一次只能为一个用户的一次计算任务服务),这样就会使计算机的资源利用率仍然不高,为了提高计算机性能,操作系统进入了多道程序阶段,多道程序合理搭配交替运行,充分利用资源,从而提高了效率。

### 3. 多道程序系统

在单道批处理系统中,内存中仅有一道作业,它无法充分利用系统中所有的资源(CPU 和内存、I/O 设备不可能同时都工作),致使系统性能较差,为了进一步提高系统吞吐量和资源的利用率,引入了多道程序设计技术。

多道程序设计技术是指在内存中同时存放若干个作业(系统吞吐率增大),使它们共享系统资源并同时运行(系统资源利用率提高)的技术。在单处理器环境下,这些作业在宏观上同时运行(并行),而在微观上交替运行(并发)。

多道程序系统的出现标志着操作系统逐渐趋向成熟的阶段,先后出现了作业调度管理、处理器管理、存储器管理、外部设备管理、文件系统管理等功能。至此,操作系统的管理结构相对完整了,操作系统概念形成。

### 4. 分时操作系统

为了解决批处理系统无法进行人机交互的问题,并使多个用户能同时使用主机资源,又

引入了分时操作系统。

在批处理方式下，用户以脱机操作方式使用计算机，用户在提交作业以后就完全脱离了自己的作业，在作业运行过程中，只有等该批作业处理结束，用户才能得到计算结果，如果结果有错，还得继续计算后，再做下一步处理。这种方式的优点是计算机效率高，但用户又对手工操作阶段的联机工作方式很喜欢。这种联机方式可以独占计算机、控制程序运行，但这种方式会造成资源效率低及出现错误修改不及时的问题。为了既能保证计算机效率，又能方便用户使用，20 世纪 60 年代中期，由于 CPU 速度的提高和分时技术的发展，使得分时交互成为可能。

所谓分时操作系统，就是使一台计算机同时为几个、几十个甚至几百个用户服务的一种操作系统。把计算机与许多终端用户连接起来，分时操作系统将系统处理器时间与内存空间按一定的时间间隔，轮流地切换给各终端用户的程序使用，由于时间间隔很短，每个用户的感受就像他独占计算机一样。分时操作系统的优点是可有效增加资源的使用率，例如，UNIX 系统就采用剥夺式动态优先的 CPU 调度，有力地支持分时操作。分时系统一般不考虑作业的概念。

产生分时操作系统是为了满足用户需求所形成的一种新型操作系统。它与多道程序系统之间，有着截然不同的性能差别，具体表现在以下几个方面：人一机交互、共享主机、便于用户上机。分时系统实现中的关键问题为：及时接收，及时处理。

### 5. 实时操作系统

20 世纪 60 年代中期计算机进入第三代，计算机的性能和可靠性有了很大的提高，计算机的应用十分广泛，用于工业生产过程控制、军事实时控制等领域的计算机由于采用多道程序系统和分时操作系统不能满足实时控制、实时信息处理的需求而引入实时操作系统，形成了各种实时处理系统。

实时操作系统是一种能在限定时间内对输入数据进行快速处理并做出响应的计算机处理系统。根据响应时间限定的严格程度，实时系统又可分为硬实时系统和软实时系统。硬实时系统又称实时控制系统，主要用于工业生产过程控制、航天系统的跟踪及控制、武器的制导等。这类系统要求响应时间快，工作要求极其安全可靠。软实时系统又称实时处理系统，主要用于响应速度在秒以上数量级的系统中，常见的有订票系统，情报、资料检索系统，银行、财务处理系统，信用卡记账、取款系统以及仓库管理系统等。这些系统一般配有大型数据库，对系统安全性、可靠性和保密性要求较高。

### 6. 通用操作系统

多道程序系统和分时操作系统的不断改进、实时操作系统的出现以及其应用日益广泛，致使操作系统日益完善。在此基础上，出现了通用操作系统，它可以同时兼有多道批处理、分时、实时处理的功能，或其中两种以上的功能。例如，当对于某一计算机系统采用实时+批处理操作系统时，把实时处理作为前台作业，批处理作为后台作业。前、后台作业的区别在于：只有前台作业不需要使用 CPU 时，后台作业才能得到 CPU 的控制权，一旦前台作业可以开始工作时，后台作业就需立即让出 CPU 供其使用。在此种情况下，批处理作业往往作为后台作业。

以上介绍的批处理、分时、实时操作系统被称为传统操作系统。至此，操作系统的基本概念、功能、基本结构和组成都已形成并逐渐趋于完善。关于现代操作系统将在 1.3 节介绍。

## 1.2 操作系统的功能

### 1.2.1 操作系统在计算机系统中的地位及运行环境

#### 1. 操作系统在计算机系统中的地位

操作系统在计算机系统中具有以下重要作用。

- (1) 计算机系统管理中心(相当于社会中的政府地位)。
- (2) 最复杂、最精确的人工开发的管理系统。
- (3) 在计算机系统中不可缺少，如果缺少操作系统，用户不可能上机使用计算机资源。

#### 2. 操作系统的运行环境

计算机硬件所提供的支持构成现代操作系统的硬件环境，其中最为重要的是中断机构。事件引发中断，中断必须加以处理，操作系统由此被驱动。操作系统是一个众多程序模块的集合，根据运行环境，这些模块大致分为 3 类：第 1 类是在系统初启时便与用户程序一起主动参与并发运行的，如作业管理程序、输入/输出程序等，它们由时钟中断、外设中断所驱动。第 2 类是直接面对用户态(亦称常态、或目态)程序的，这是一些“被动”地为用户服务的程序。这类程序的每一个模块都与一条系统调用指令对应，仅当用户执行系统调用指令时，对应的程序模块才被调用、被执行。系统调用指令的执行是经过陷入中断机构处理的。因此从这个意义上说，第 2 类程序也是由中断驱动的。第 3 类是那些既不主动运行也不直接面对用户程序的、隐藏在操作系统内部的、由前 2 类程序调用的模块。既然前两类程序是由中断驱动的，那么第 3 类程序也是由中断驱动的。应当注意，操作系统本身的代码运行在核心态(亦称管态、特态)，从用户态进入核心态的唯一途径是中断。

### 1.2.2 操作系统的特征

操作系统具有五大特征，其中前两项是操作系统的基本特征，不具备这两项的管理程序不能被称为操作系统。

#### 1. 并发性

并发性是从宏观上看到多个程序在同时活动，微观上这些程序在计算机系统内的各个硬件上交替运行，由操作系统负责这些程序之间的运行切换；从外部宏观上，人们看到的是多个程序都在系统中运行。多个程序的并发执行是通过引入进程来实现的，程序本身是静态实体，系统必须分别为每个程序建立进程。进程是指在系统中能独立运行并作为资源分配的基本单位，它由一组机器指令、数据和堆栈等组成，是一个活动实体。

并行性和并发性是两个既相似又有区别的不同概念。并发是指在某一时间间隔内，计算机系统内部有多个程序活动。并行是指在同一时刻计算机系统内的多个程序都在运行，这些程序的运行依赖于 CPU。所以，基本部件同一时刻都处于工作状态时称为并行工作。在单机系统中，多个程序是不能并行的。

进程和并发是现代 OS 中最基本的概念，也是 OS 运行的基础。长期以来，进程都是 OS 中可以拥有资源和独立运行的基本单位。线程的概念在 20 世纪 80 年代中期才出现，它是比进程更小的单位。通常一个进程中包含若干线程。线程基本上不拥有资源(也称为轻载进程)，运行起来更轻松，能够更好地提示系统内多个程序间并发执行的程度。

### 2. 共享性

共享性是指多个用户或程序对软、硬件资源的共同占用或使用。操作系统通过充分的资源共享来提高整个资源利用率，所以资源共享是现代操作系统的一个重要特征。

共享方式分为互斥共享和同时共享。互斥共享的资源不允许两个以上进程或线程同时使用，只能一个作业用完并释放以后，另一个作业串行使用。另一种共享方式是同时共享，允许多个作业交替使用，一个作业暂时不使用时，可以让另一个作业使用。宏观上看仍是多个作业同时使用，即同时共享。互斥共享的资源有打印机、绘图仪等，同时共享的资源有磁盘等。

软件资源的共享也可以分为互斥共享和同时共享两种。其中，只读的数据、数据结构、只读文件及纯的可执行文件是同时共享的。而可写的数据、数据结构及文件均为互斥共享的。

并发和共享是 OS 中的两个最基本的特征，又是互为存在的条件。一方面，程序(进程)的并发执行是资源共享的条件，系统若不允许程序并发执行，就不能存在资源共享问题；另一方面，若系统不能对资源共享实施有效管理、对共享资源的诸进程进行协调，就必然影响程序并发执行的程度，以至于无法并发执行。

### 3. 虚拟性

在操作系统中所谓的“虚拟”，是指通过某种技术把一个物理机器虚拟成若干个逻辑上的机器，使得每个用户都感觉是“独占”机器。在多道分时系统中，虽然只有一个 CPU，但每个终端用户却都认为是有 CPU 在专门为他服务，即利用多道程序技术可以把一台物理上的 CPU 虚拟为多台逻辑上的 CPU，称为虚拟处理器。类似地，也可以把一台物理 I/O 设备虚拟为多台逻辑上的 I/O 设备，称为虚拟设备。在操作系统中虚拟的实现，主要是通过分时使用的方法来实现的。

### 4. 异步性

异步性又称不确定性，是指进程以人们不可预知的速度向前推进。在多道程序系统下，多个进程允许并发执行，但由于资源等因素的限制，通常，进程的执行并非“一气呵成”，而是以“走走停停”的方式运行。内存中的每个进程在何时执行，何时暂停，以怎样的速度向前推进，每道程序总共需多少时间才能完成，都是不可预知的，很可能是先进入内存的作业后完成；而后进入内存的作业先完成。或者说，进程是以异步方式运行的。尽管如此，但只要运行环境相同，作业经多次运行，都会获得完全相同的结果，因此，异步运行方式是允许的。此即进程的异步性，是操作系统的一个重要特征。

### 5. 可重构性

可重构性支持对计算机系统可靠性和配置动态化的考虑。在多处理器系统中应使操作系统具有这样的能力：当系统中某台处理器或存储模块等资源发生故障时，系统能自动切除故障资源，换上备份资源，对系统进行重构使之继续工作。现代操作系统允许在系统运行过程

中动态地改变硬件的配置。当配置变更时，操作系统截获变更信息，然后自动地重构系统配置信息。例如，Windows 95 的即插即用(Plug and Play)方案允许用户将便携式系统从台式停靠站卸下并迅速移到另一位置。在此期间用户不必关闭计算机。当用户再度用机时，即插即用子系统已经重构好了 Windows 95 的新配置，用户马上就可以继续他原来的工作。现在，对于动态配置的要求越来越强烈。如用户有时用一个 PCMCIA 卡上的调制解调器作为 COM1 端口插入到便携机上，而有时又需要将便携机安装到停靠站上。又如，无线网络要支持可移动工作站，也必须能够进行动态配置。即插即用是一个受到关注的方案。

### 1.2.3 操作系统的功能

从资源管理的角度看，操作系统主要功能分四大模块，包括处理器管理，存储器管理，设备管理，文件、作业管理、通信事务管理。此外，为了方便用户使用操作系统，还给用户提供了一个友好的用户接口。

#### 1. 处理器管理

处理器管理是指操作系统根据一定的调度算法分配处理器。在单处理器系统中，进程管理也属于这一范围。在采用多道程序设计技术的操作系统中，处理器在操作系统的控制下交替地被多道程序所占用。在多处理器系统中，处理器的管理要复杂一些。一方面必须了解每台处理器的功能，以便把适合于该处理器的任务分配给它；另一方面，也应确切地知道一个作业中各个任务之间的串、并行关系。处理器集的概念较好地解决了多处理器调度问题。

#### 2. 存储器管理

存储器管理是指分配、收回与保护存储单元。一道程序在启动执行前必须装入主存，操作系统应当根据程序的大小和当前主存空间的实际情况，为每一道程序分配使之能运行的必要的存储空间。当程序执行完后，操作系统把该程序所占用的全部存储空间收回，以做他用。显然，一道程序未必要全部装入主存才能运行。如果主存空间充足，可以把程序全部装入。如果主存空间紧缺，则把程序的一部分装入，在运行过程中需要访问未装入的程序片段时，再把这些程序片段装入主存，由此产生了虚拟存储管理算法。一般认为，主存管理算法可推广到外存(磁盘、磁带、光盘)空间的管理上。在计算机系统中，外存属于外围设备。

#### 3. 设备管理

设备管理主要是对设备进行分配、回收与控制。通信设备在工作过程中的控制尤为重要。用户程序使用外围设备的步骤是先申请、再使用、最后释放。与此相应，操作系统必须把有限的设备进行优化调度以满足多个用户程序的要求，对分配出去的设备进行有效控制以发挥设备的最大能力，最后还要对用完的设备及时收回。为了提高设备利用效率，设备管理中引入了虚拟设备的概念和 Spooling 技术。

#### 4. 文件、作业管理、通信事务管理

文件管理涉及文件存储空间的分配与回收、文件目录管理、文件读/写与保护等。由于现代计算机系统与信息资源常以文件的形式出现，因此管理好文件就能使软件资源发挥更好的效益。目前，一个全方位的以文档为中心的资源管理方法受到用户的欢迎。用户希望只需要对文档进行操作，而不必关心被操作的文档究竟是文件、程序、目录、声像数据、磁盘卷