

高等院校精品课程系列教材 · 国家级

操作系统原理 与Linux系统实验

庞丽萍 郑 然 ©编著



*Principle of Operating System
and Linux Experiments*



机械工业出版社
China Machine Press

高等院校精品课程系列教材·国家级

操作系统原理 与Linux系统实验

庞丽萍 郑 然◎编著



*Principle of Operating System
and Linux Experiments*



机械工业出版社
China Machine Press

本书全面、系统地阐述了现代操作系统的基本原理、主要功能及实现技术，重点论述多用户、多任务操作系统的运行机制，系统资源管理的策略和方法，操作系统提供的用户界面。本书给出了一个操作系统实验体系。该实验体系涉及操作系统的核心实现技术和一些重要的实际应用，包括进程控制、增加新的系统功能调用、增加新的设备驱动程序等操作系统功能模块的实现方法；还包括 Linux 文件系统、服务器配置与应用的范例。每个实验都是一个完整的小体系，包括：预备知识、实验指导、实验目的和内容以及实验中的程序范例。

本书既可作为高等院校计算机和信息类本科、专科各专业和其他相关专业的教材，亦可供从事计算机科学、工程和应用等方面工作的科技人员参考。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目（CIP）数据

操作系统原理与 Linux 系统实验 / 庞丽萍，郑然编著. —北京：机械工业出版社，2011.4
（高等院校精品课程系列教材）

ISBN 978-7-111-33568-9

I. 操… II. ①庞… ②郑… III. Linux 操作系统—高等学校—教材 IV. TP316.89

中国版本图书馆 CIP 数据核字（2011）第 031581 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：张少波

北京市荣盛彩色印刷有限公司印刷

2011 年 5 月第 1 版第 1 次印刷

185mm×260mm·16 印张

标准书号：ISBN 978-7-111-33568-9

定价：29.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：（010）88378991；88361066

购书热线：（010）68326294；88379649；68995259

投稿热线：（010）88379604

读者信箱：hzjsj@hzbook.com

前言

操作系统是计算机系统的核心软件，在计算机系统中占有重要的地位。学习操作系统就是要理解并熟练地运用它，为深入掌握计算机专业知识，进一步提升软件开发能力乃至系统软件开发能力打下坚实的基础。而要真正学懂操作系统却非易事，一方面需要有一本好的教材，另一方面则需要学习者的勤奋与努力。

好的教材应能抓住操作系统原理的精髓，并能深入浅出地加以勾画与描述；还要有一套涉及操作系统核心的实验体系，使学生能方便地体验操作系统功能模块的实现方法，掌握与实际应用有关的技术。

本教材针对操作系统内容庞杂、涉及面广的特点，在教材内容的选取上注重基础性、实质性、先进性，在框架的设计上注重逻辑性、完整性，将操作系统的内容组织成一个完整的逻辑整体；提炼现代多用户、多任务操作系统的本质特征，始终贯穿并发、共享的主线；深入浅出地描述现代操作系统的动态特征，论述操作系统的重要概念——进程、支持多进程并发活动必需的数据结构、进程控制与进程调度功能的实现技术，讨论系统资源的共享、资源管理的策略与方法。

本教材针对操作系统实践性强的特征，以当前流行的 Linux 操作系统为例，给出了一个操作系统实验体系。该实验体系涉及操作系统的核心实验和一些重要的实际应用，其内容包括 Linux 系统的使用、Linux 文件系统、服务器的配置与应用、proc 文件系统、进程（线程）控制、进程间通信、系统功能调用的增加、设备驱动程序的加载、Linux 主存管理等。每个实验都是一个完整的小体系，包括：预备知识、实验指导、实验目的和内容以及实验中的程序范例。

在教材内容上，操作系统原理的论述和实验内容的讨论相互关联、相互印证，读者能以操作系统原理为指导，针对每一个实验去思考自己要做什么，并知道如何去做。这样，每次实验都会有收获，能更好地理解操作系统原理中有关的概念、操作系统实现的机制和策略，体验操作系统功能模块的实现技术。

本教材的论述力求深入浅出，通俗易懂，使读者便于阅读和理解。实验内容经过精心的选择，实验中的程序范例都做了验证。

我们在教学和编写教材的过程中，学习、参考了有关操作系统、UNIX 系统、Linux 系统方面的好的教材；在实验指导的写作过程中，参考了大量网络资源。我们对所有帮助过我们的老师、同行表示深深的感谢。由于篇幅所限，在参考文献中只列出了部分参考教材和网络资源所在的网站首页，请老师和同行谅解。另外，对机械工业出版社的领导及有关同志深表谢意，他们对此书的出版和发行做了大量的工作。

恳切地希望能继续得到同行和读者的批评和帮助，以便使此书的质量能不断地提高。

庞丽萍 郑然

2011 年 1 月于武汉

教学建议

篇	教学章节	教学要求	课时
第一篇 操作系统原理	第 1 章 绪 论	- 了解操作系统在计算机系统的位置 - 掌握多道程序设计技术和分时技术 - 了解操作系统类型, 掌握批量 OS、分时 OS、实时 OS 的特点 - 掌握操作系统的定义、功能、特性	4
	第 2 章 操作系统的运行基础 与结构	- 掌握处理机的态、管态、目态及其区别 - 掌握中断概念、中断类型、中断响应的实质、软件中断处理过程 - 操作系统虚拟机的概念	4
	第 3 章 操作系统的用户界面	- 了解系统初启与系统生成 - 了解应用程序的处理过程, 掌握作业、作业步概念 - 掌握操作系统的用户界面类别 - 掌握系统调用的定义、系统调用的实现过程	4
	第 4 章 进程及进程管理	- 了解顺序程序、并发程序的定义与特点 - 掌握进程定义、进程状态及变迁、进程的组成 - 了解进程控制的功能及基本的进程控制原语, 掌握进程创建、进程撤销、进程等待与唤醒的功能 - 了解锁、上锁原语、开锁原语的功能, 掌握信号灯及 P、V 操作的概念并会使用 - 掌握临界资源、临界区、互斥的定义, 能用信号灯的 P、V 操作实现互斥 - 掌握同步概念, 能用信号灯的 P、V 操作实现进程同步	10
	第 5 章 操作系统的资源管理	- 掌握常用的两种分配策略(先请求先服务、优先调度策略) - 掌握死锁的定义与例子、产生死锁的原因与必要条件、死锁的预防(资源静态分配)、死锁的避免(有序资源分配方法) - 了解作业调度的功能, 掌握常用的作业调度算法; 掌握进程调度的功能、调度方式、常用的进程调度算法 - 掌握地址映射的定义及类型、虚存的概念; 掌握两种分区放置策略; 掌握页式地址变换过程、请调页面机制(扩充页表功能、缺页处理)、淘汰机制(扩充页表功能、常用的页面置换算法) - 掌握设备独立性、设备控制块概念, 掌握缓冲概念, 常用的设备分配方法, I/O 控制功能 - 了解文件及文件系统概念; 掌握文件的逻辑结构及存取方法、文件的物理结构; 掌握文件目录定义与结构; 掌握文件共享的定义及实现方法(当前目录、链接技术); 掌握文件的安全的定义、主要的文件操作命令及打开文件、关闭文件操作命令的功能	16

(续)

篇	教学章节	教学要求	课时
第二篇 Linux 系统 实验	第6章 Linux 系统简介	- 了解 Linux 发展历程、基本特征 - 掌握 Linux 多个不同的发行版本和内核版本	2
	第一篇总课时	第一篇 建议课时	40
	实验一 Linux 系统的使用	- 了解 Linux 系统的安装、熟悉系统的启动过程和桌面环境 - 掌握 Linux 环境下 vi 编辑器的使用方法 - 掌握 Linux 常用的 Shell 命令以及 Shell 编程的基本方法 - 掌握 Linux 系统中编辑、编译、调试、运行一个 C 语言程序的全过程	2+2
	实验二 Linux 文件系统	- 熟悉 Linux 文件系统的文件和目录结构, 掌握 Linux 文件系统的基本特征 - 掌握命令行方式下文件操作命令和程序中文件操作函数的使用方法, 掌握 Linux 文件系统的加载和卸载方法	2+2
	实验三 服务器的配置 与应用	- 掌握 Linux 网络配置方法 - 掌握 Linux 远程控制方法 - 掌握 Web 服务器、FTP 服务器等常用服务器的配置方法	2+4
	实验四 proc 文件系统	- 理解和分析 proc 文件系统, 掌握其特点和使用方法 - 掌握 Linux 终端图形编程方法, 编写基于文本的图形界面 - 掌握 Linux 下图形界面编程工具, 能用 GTK 或 Qt 进行图形界面的开发	2+6
	实验五 进程(线程)控制	- 理解进程概念, 明确进程与程序的区别 - 掌握 Linux 进程创建及撤销的方法, 认识并发执行的实质 - 掌握 Linux 系统下守护进程的创建方法	2+2
	实验六 进程间通信	- 掌握 Linux 系统进程软中断通信的基本原理和实现方法 - 了解管道通信的特点, 掌握管道通信的使用方法 - 了解消息队列通信机制及原理, 掌握消息队列相关系统调用的使用方法及功能	2+6
	实验七 增加新的系统调用	- 加深对系统调用的理解, 掌握增加与调用系统调用的方法 - 掌握内核编译方法	2+4
	实验八 增加新的设备 驱动程序	- 了解 Linux 设备驱动程序的原理 - 学习 Linux 2.6 内核下设备驱动程序的编写方法 - 掌握用模块方式设计和加载驱动程序的方法 - 学会如何通过配置编译内核, 将驱动程序添加进内核	2+6
	实验九 Linux 主存管理	- 掌握 Linux 虚拟主存管理的原理 - 掌握 Linux 中动态主存操作函数的使用方法 - 掌握 Linux 监测主存使用情况的方法	2+4
	实验总课时	第二篇实验部分建议课时	18+36

说明:

- 1) 建议实验部分教学全部在多媒体机房内完成, 实现“讲-练”结合, 每个实验用 2 学时讲解, 2~6 学时学生自己动手上机实验(‘+’前面为讲授学时, 后面为上机学时)。
- 2) 根据各学校教学要求和计划学时数的不同, 可对教学内容进行取舍。如可选九个实验中的部分实验, 也可将上机学时中的一部分作为课外学时。

目 录

前言 教学建议

第一篇 操作系统原理

第 1 章 绪论	2
1.1 操作系统与计算机系统的关系	2
1.1.1 计算机系统的组成	2
1.1.2 操作系统在计算机系统中的位置	4
1.1.3 操作系统与计算机体系结构的关系	5
1.2 操作系统的发展历程	7
1.2.1 操作系统的发展阶段	7
1.2.2 多道程序设计技术与分时技术	8
1.3 操作系统的基本概念	10
1.3.1 操作系统的定义和特性	10
1.3.2 操作系统的资源管理功能	12
1.4 操作系统的基本类型	14
1.4.1 批量操作系统	14
1.4.2 分时操作系统	14
1.4.3 实时操作系统	15
1.4.4 个人计算机操作系统	16
1.4.5 网络操作系统	16
1.4.6 分布式操作系统	17
1.5 操作系统采用的关键技术	19
1.5.1 操作系统采用的并行处理技术	19
1.5.2 操作系统采用的虚拟技术	20
习题 1	20

第 2 章 操作系统的运行基础与结构	22
2.1 操作系统的运行基础	22
2.1.1 处理机的态	22
2.1.2 特权指令	23
2.1.3 中断概念及类型	23
2.1.4 中断进入	24
2.1.5 软件中断处理过程	26
2.1.6 向量中断	27
2.2 操作系统虚拟机	28
2.3 操作系统的逻辑结构	29
2.3.1 操作系统的四种组织结构	29
2.3.2 Linux 操作系统的逻辑结构	31
2.4 操作系统运行时的调用方式	32
习题 2	33
第 3 章 操作系统的用户界面	34
3.1 操作系统提供的环境	34
3.1.1 操作系统的生成和系统引导	34
3.1.2 应用程序的处理	37
3.2 用户界面	39
3.2.1 用户界面概述	39
3.2.2 操作系统提供的用户界面	39
3.3 系统功能调用	41
3.3.1 系统功能调用的定义	41
3.3.2 系统功能调用的实现	42
3.3.3 应用编程接口	43
习题 3	43
第 4 章 进程及进程管理	44
4.1 进程的引入	44
4.1.1 程序的顺序执行	44
4.1.2 程序的并发执行	45

4.2 进程的相关概念.....	48	5.2.5 死锁的避免.....	79
4.2.1 进程的定义.....	48	5.3 处理机管理.....	80
4.2.2 进程的状态与变迁.....	49	5.3.1 处理机的多级调度.....	80
4.2.3 进程描述.....	50	5.3.2 作业调度.....	81
4.2.4 线程的定义与状态.....	52	5.3.3 进程调度的功能.....	84
4.3 进程控制.....	54	5.3.4 调度方式.....	84
4.3.1 进程控制的概念.....	54	5.3.5 进程调度算法.....	85
4.3.2 进程创建与撤销.....	54	5.3.6 调度用的进程状态变迁图.....	86
4.3.3 进程等待与唤醒.....	55	5.4 主存管理.....	87
4.4 进程协作与系统的支持.....	56	5.4.1 主存管理的功能.....	87
4.4.1 进程竞争与合作.....	57	5.4.2 分区存储管理及其存在的 问题.....	91
4.4.2 进程互斥的概念.....	57	5.4.3 分页存储管理的基本概念.....	95
4.4.3 进程同步的概念.....	59	5.4.4 页式地址变换.....	97
4.4.4 锁和上锁、开锁操作.....	60	5.4.5 请求页面的机制.....	98
4.4.5 信号灯和 P、V 操作.....	61	5.4.6 淘汰机制与策略.....	100
4.5 进程互斥与同步的实现.....	62	5.4.7 段页式存储管理.....	104
4.5.1 用锁机制实现进程互斥.....	62	5.5 设备管理.....	107
4.5.2 用信号灯机制实现进程互斥.....	62	5.5.1 设备管理的功能.....	107
4.5.3 进程同步的实现.....	63	5.5.2 设备独立性与设备控制块.....	108
4.5.4 生产者-消费者问题.....	66	5.5.3 缓冲技术.....	110
4.6 进程通信.....	67	5.5.4 设备分配技术.....	112
4.6.1 进程通信的概念.....	67	5.5.5 输入/输出控制.....	114
4.6.2 进程通信方式.....	67	5.6 文件系统.....	118
习题 4.....	69	5.6.1 文件与文件系统.....	118
第 5 章 操作系统的资源管理.....	71	5.6.2 文件的逻辑结构和存取 方法.....	120
5.1 资源管理的机制与策略.....	71	5.6.3 文件的物理结构.....	121
5.1.1 资源管理任务.....	71	5.6.4 文件目录.....	125
5.1.2 虚拟资源.....	72	5.6.5 文件共享与安全.....	127
5.1.3 资源分配机制.....	73	5.6.6 文件操作与文件备份.....	130
5.1.4 资源分配策略.....	74	习题 5.....	132
5.2 死锁及其解决方法.....	76	第 6 章 Linux 系统简介.....	136
5.2.1 死锁的定义与例子.....	76	6.1 Linux 的发展历程.....	136
5.2.2 产生死锁的原因和必要 条件.....	77	6.2 Linux 的基本特性.....	137
5.2.3 解决死锁问题的策略.....	78	6.3 Linux 的发行版本.....	138
5.2.4 死锁的预防.....	79		

6.4 Linux 内核版本	141	(二) 架设 FTP 服务器	178
6.5 学习 Linux 的几点建议	142	(三) 架设 Email 服务器	178
第二篇 Linux 系统实验			
实验一 Linux 系统的使用	144	<三> 实验目的和内容	181
<一> 预备知识	144	(一) 实验目的	181
(一) Linux 系统的安装	144	(二) 实验内容	181
(二) Linux 系统的组成	148	实验四 proc 文件系统	182
(三) Linux 系统的用户界面	148	<一> 预备知识	182
<二> 实验指导	150	(一) proc 文件系统功能	182
(一) 屏幕编辑程序 vi	150	(二) proc 文件描述	183
(二) Shell 脚本编程	152	(三) proc 进程目录结构	184
(三) 编译器 gcc	155	<二> 实验指导	185
<三> 实验目的、内容和范例	156	(一) proc 信息获取	185
(一) 实验目的	156	(二) 终端图形编程 curses 库	186
(二) 实验内容	156	(三) 图形界面编程 GTK 和 Qt 工 具包	188
(三) 实验范例	156	<三> 实验目的、内容和范例	190
实验二 Linux 文件系统	158	(一) 实验目的	190
<一> 预备知识	158	(二) 实验内容	190
(一) Linux 文件系统简介	158	(三) 实验范例	190
(二) Linux 系统的目录结构	159	实验五 进程(线程)控制	192
(三) Linux 文件属性	161	<一> 预备知识	192
<二> 实验指导	163	(一) Linux 进程及其属性	192
(一) 文件系统的生成和加载	163	(二) Linux 进程派生关系	193
(二) 文件系统操作命令	165	(三) Linux 线程	194
(三) 文件操作函数	166	<二> 实验指导	196
<三> 实验目的、内容和范例	169	(一) Linux 进程管理命令	196
(一) 实验目的	169	(二) Linux 进程控制函数	198
(二) 实验内容	169	(三) Linux 守护进程	199
(三) 实验范例	169	<三> 实验目的、内容和范例	200
实验三 服务器的配置与应用	170	(一) 实验目的	200
<一> 预备知识	170	(二) 实验内容	200
(一) Linux 配置文件	170	(三) 实验范例	201
(二) Linux 网络配置	171	实验六 进程间通信	202
(三) Linux 远程控制	174	<一> 预备知识	202
<二> 实验指导	175	(一) Linux 进程间通信	202
(一) 架设 Web 服务器	175	(二) 进程软中断通信	204
		(三) 管道和消息队列	205

<二> 实验指导.....	208	(三) Linux 设备驱动	225
(一) 软中断通信函数	208	<二> 实验指导	226
(二) 管道通信的使用	210	(一) 设备驱动函数	226
(三) 消息队列的应用	211	(二) 模块方式动态加载	227
<三> 实验目的、内容和范例	213	(三) 编译时静态加载	228
(一) 实验目的	213	<三> 实验目的、内容和范例	229
(二) 实验内容	213	(一) 实验目的	229
(三) 实验范例	213	(二) 实验内容	229
实验七 增加新的系统调用	216	(三) 实验范例	230
<一> 预备知识	216	实验九 Linux 主存管理	235
(一) Linux 内核结构	216	<一> 预备知识	235
(二) Linux 内核源代码	218	(一) Linux 主存管理简介	235
(三) Linux 系统调用	219	(二) Linux 主存分配	236
<二> 实验指导	220	(三) Linux 主存回收	238
(一) 系统调用的添加	220	<二> 实验指导	239
(二) Linux 内核编译	221	(一) 主存信息的监视	239
<三> 实验目的和内容	222	(二) 主存的申请和释放	241
(一) 实验目的	222	<三> 实验目的、内容和范例	242
(二) 实验内容	222	(一) 实验目的	242
实验八 增加新的设备驱动程序	223	(二) 实验内容	242
<一> 预备知识	223	(三) 实验范例	242
(一) Linux 设备分类	223	参考文献	243
(二) Linux 设备文件	224		

第一篇

操作系统原理

操作系统是计算机系统的核心软件，在计算机系统中占有重要的地位。操作系统管理计算机系统的各类资源，控制用户程序的执行，为用户提供使用计算机的手段和方法。现代操作系统支持多用户、多任务的同时执行，为此，操作系统实现了多用户、多任务对系统资源的有效共享；解决了多用户、多任务同时执行时产生的问题。本篇清晰、概要地阐述了现代操作系统的概念及其实现技术。

第1章 绪 论

第2章 操作系统的运行基础与结构

第3章 操作系统的用户界面

第4章 进程及进程管理

第5章 操作系统的资源管理

第6章 Linux 系统简介

第 1 章

绪 论

随着计算机技术的快速发展,计算机在科学研究、人类活动以及社会生活的各个领域得到了广泛的应用,从办公自动化到数据、图像处理,从工业控制到科学计算,从非面授教育到电子商务……计算机的应用已是无处不在,无处不有!那么,人们如何与计算机交互?计算机如何根据用户的意愿去工作?它又是如何去管理繁杂的数据和各类资源?所有这些问题都是由配置在计算机上的一个重要系统软件——操作系统来解决的。现代的计算机几乎没有不配置操作系统的。那么,操作系统是什么?能做些什么?其功能是如何实现的?它在计算机系统中处于什么位置?有什么特点?这些问题都是本章要讨论的内容。

1.1 操作系统与计算机系统的关系

1.1.1 计算机系统的组成

计算机系统是一个整体概念,不论是大型机、小型机还是微型机,都是由计算机硬件和软件两大部分组成的。

1. 硬件结构

硬件指的是组成计算机的任何机械的、磁性的、电子的装置或部件,硬件也称为硬设备,它由 CPU (Central Processing Unit, 中央处理器)、存储器、输入/输出 (I/O) 设备等组成。

CPU 是一种能够解释指令、执行指令并控制操作顺序的硬设备。它由算术逻辑运算单元 (ALU) 和控制单元构成。ALU 包含一个能完成算术和逻辑操作的功能单元、一组通用寄存器和状态寄存器。通用寄存器为功能单元提供操作数,并能接收、保存操作的结果。状态寄存器保存着 CPU 运行过程中的当前状态。现代 CPU 一般包含 32~64 个通用寄存器,每个寄存器能够保存一个 32 位 (bit) 的数值。控制单元负责从主存储器提取指令、分析其类型,并产生信号通知计算机其他部分执行指令所指定的操作。控制单元包含一个程序计数器 (program counter, PC) 和一个指令寄存器 (instruction register, IR)。程序计数器指示下一步应该执行的指令,而指令寄存器包含当前指令的拷贝。习惯上,也将 CPU 称为处理机 (processor, 能执行指令序列的硬部件)。

主存储器 (main memory), 又称主存, 是组成计算机的一个重要部件, 其作用是存放指令和数据, 并能由 CPU 直接随机存取。主存接口由存储地址寄存器 (memory address register, MAR)、

存储数据寄存器（memory data register, MDR）以及命令寄存器（command register, CR）三个寄存器组成。主存的单元数目和每个单元的位数取决于当时的电子制造技术以及硬件设计的考虑。现代计算机为了提高性能，兼顾合理的造价，往往采用多级存储体系。多级存储体系由存储容量小、存取速度高的高速缓冲存储器，存储容量和存取速度适中的主存，存储容量大但存取速度较慢的辅存组成。

I/O 设备负责信息的传输，将数据从外部传输到计算机内，或将主存中的内容传输到计算机的外部世界。I/O 设备分为存储设备（如磁盘或磁带）、字符设备（如终端显示器、鼠标）和通信设备（如连接调制解调器的串行端口或网络接口）。每个设备通过设备控制器与计算机的地址总线 and 数据总线相连。控制器提供一组物理部件，可以通过 CPU 指令操纵它们以完成 I/O 操作。

上述的这些部件一般采用总线结构组织在一起，典型的单处理机系统结构如图 1-1 所示。由这些硬部件构成的机器称为裸机，它是计算机系统的物质基础和最基本的硬件环境。

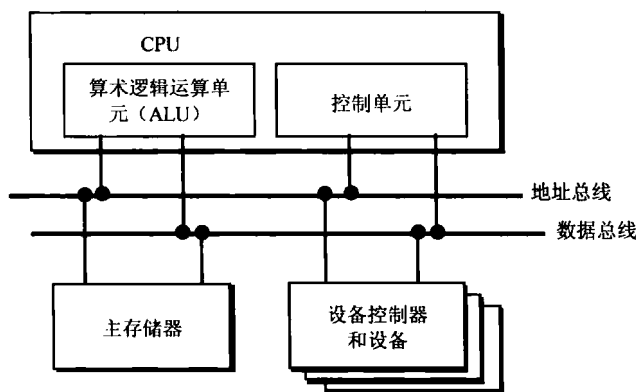


图 1-1 单处理机系统结构

计算机的硬件结构是由著名数学家冯·诺依曼（Von Neumann）于 20 世纪 40 年代提出来的。冯·诺依曼总结了手工计算的规律以及前人研究计算机的经验，提出并实现了称为“存储程序式计算机”的计算机，又称为冯·诺依曼计算机体系结构。这台计算机实现了计算的自动化，其最主要的特征是计算方案或称计算机程序必须存放在机器内。从 20 世纪 40 年代至今，计算机体系结构不断地发展变化，但对于单 CPU 的计算机而言，其结构仍然是存储程序式计算机的体系结构。

冯·诺依曼计算机是人类历史上第一次实现自动计算的计算机，它的影响是十分深远的。它具有逻辑判断能力和自动连续运算能力。它的特点是集中顺序过程控制。计算是过程性的，完全模拟人的手工操作过程，即首先取原始数据，执行一个操作，将中间结果保存起来，再取一个数，与中间结果一起又执行一个操作，如此计算下去，直到计算完成。由系统中的程序计数器体现其顺序性（在单 CPU 的计算机系统中只有一个程序计数器），它根据程序规定的顺序依次执行每一个操作。集中控制是指机器各部件的工作由 CPU 集中管理和指挥。冯·诺依曼计算机的计算模型是顺序过程计算模型。

2. 软件层次

没有配置软件的计算机称为裸机，裸机上没有任何一种可以协助用户解决问题的手段，只提供最低级的机器语言。裸机这种工作环境用户是不能接受的。为了对硬件的性能加以扩充和完善，方便用户使用计算机，在裸机外添加了能实现各种功能的软件程序。例如，为了方便用户描述自己的算题任务提供了程序设计语言以及相应的翻译程序（如 C 程序设计语言及 C 编译程序）；为了方便、有效地解决各类问题，提供了各种服务性程序和实用程序，如系统程序库、编辑程序、连接装配程

序；为了维护系统正常工作，可配置查错程序、诊断程序；还有数据库管理系统、各种应用系统等。这些系统程序或应用程序以及有关的数据形成了软件。另外，在程序研制过程中会形成大量文档和资料，它们也属于软件范畴。所以，软件是程序、数据以及有关的文档资料的集合。软件一般可以分为以下几类。

系统软件：操作系统、编译系统、数据库系统以及与计算机密切相关的程序。

应用软件：应用程序、软件包（如数据统计软件包、运筹计算软件包等）。

工具软件：各种诊断程序、检查程序、引导程序。

依上所述，计算机系统的软件层次由图 1-2 说明。

在所有软件中有一个很重要的软件称为操作系统，它管理系统中的各种软、硬件资源，控制用户和应用程序的工作流程，将系统各部件有机地融合成一个整体，使计算机真正体现了系统的完整性和可利用性。

在计算机系统中，硬件是计算机系统的物质基础，没有硬件就不能执行指令和实施最原始、最简单的操作，软件也就失去了效用；而若只有硬件，没有配置相应的软件，计算机也不能发挥它的潜在能力，这些硬件资源也就没有活力。因此，硬件和软件这二者是互相依赖、互相促进的。只有软件和硬件有机地结合在一起的系统，才能称得上是一个计算机系统。

1.1.2 操作系统在计算机系统的位置

计算机系统是由硬件和软件两大部分组成的一个完整的系统。由所有的硬件组成的裸机处于系统的最底层，裸机的外面是软件部分。软件是相当丰富的，它们的功能和使用特性也不同。其中，应用软件是在系统软件的支持下完成各项工作的，所以它应在系统软件的外层。系统软件中操作系统处于核心地位，它负责整个系统的管理和控制，是其他系统软件和应用软件运行的基础。所以，在所有软件中操作系统处于最内层并直接与裸机相接，与计算机硬件的关系最为密切。

图 1-3 更清楚地说明操作系统在计算机系统的位置。在计算机系统中，操作系统位于裸机之上、所有软件最内层这样一个位置上，它与计算机硬件和其他软件有着密切的联系。从操作系统在计算机系统所处的位置来分析操作系统应具备的功能及其实现技术，能对操作系统有更深入的理解。操作系统与各层的关系表现在两个方面：一是操作系统对各层的管理和控制；二是各层对操作系统的影响和制约。

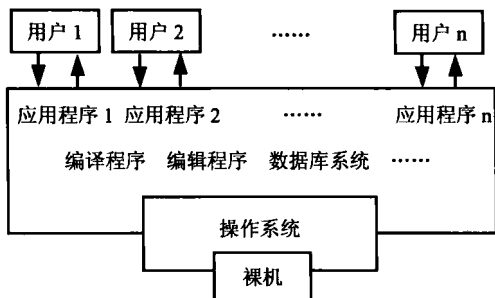


图 1-2 计算机系统的软件层次

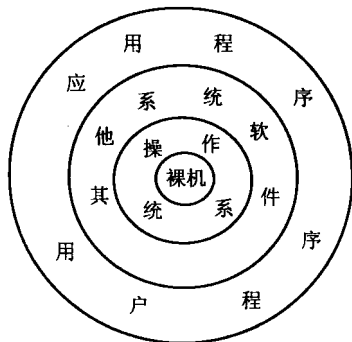


图 1-3 计算机系统的组成

1. 操作系统对各层的管理和控制

(1) 操作系统直接与硬件交互

操作系统是在裸机上扩充的第一层软件，负责所有硬部件的分配、控制工作。如控制 CPU 的工作，访问存储器，进行磁盘、打印机、显示器、键盘等设备的驱动和中断处理工作。

(2) 操作系统与用户和应用程序交互

操作系统为上层软件提供运行基础,它与其他系统软件 and 应用程序是一种管理与被管理、调用与被调用的关系。所有软件都在操作系统的管理和控制下工作。另外,操作系统为其他软件 and 用户提供运行环境,即提供方便、简单的用户界面。

2. 各层对操作系统的影响和制约

(1) 计算机硬件结构对操作系统实现技术的影响和制约

操作系统在裸机上运行,计算机的硬件结构对操作系统的实现技术有着重大的影响和制约。单 CPU 计算机的特点是集中顺序过程控制,其计算模型是顺序过程计算模型。而现代操作系统大多数是多用户、多任务操作系统,是一个并行计算模型,这就是一对矛盾。

大家熟知的 Windows 系统就是一个多任务操作系统,用户在 Windows 系统中可以开很多窗口,一个窗口就是一个任务,系统支持多个任务同时执行。例如,用户可以在编辑一个图像时,同时听着音乐,还可下载一个文件。那么单 CPU 计算机如何运行多任务呢?这是一件十分困难的事,需要许多技术来支持。为此,操作系统提出并实现了以下各章节要讨论的内容,如多道程序设计技术、分时技术,进程概念、进程控制及同步,资源分配与调度的机制和策略,使得单 CPU 的计算机能支持多任务同时执行,只是操作系统的实现技术会比较复杂。这就是计算机的硬件结构对操作系统的实现技术的影响和制约。

(2) 用户和应用程序的需求对操作系统实现技术的影响和制约

用户和上层软件运行在操作系统提供的环境上,对操作系统会提出各种要求,操作系统必须满足不同的应用需求,为此需要设计不同类型的操作系统。例如,多用户需要公平的响应,直接与计算机“会话”,这就需要分时操作系统。若要进行过程控制或实时信息处理,应用程序需要实时响应,这时操作系统必须提供实时和具有可预测能力的服务,即提供实时操作系统。另外,不论具有哪种需求的用户都希望计算机使用起来方便、快捷,所以,操作系统必须提供良好的、直观的用户界面。

计算机配置操作系统的目的就是要提高计算机的使用效率,方便用户的使用,这也就是操作系统的宗旨。在操作系统发展的初期,由于硬件价格昂贵,将提高计算机的使用效率放在了第一位。随着计算机硬件技术、微电子技术的快速发展,计算机应用的普及和应用水平的日益提高,方便用户的使用、提高服务质量(QoS)越来越重要。在操作系统的功能实现上必须考虑这一因素和变化。

1.1.3 操作系统与计算机体系结构的关系

计算机系统的硬件基础是冯·诺依曼体系结构,操作系统是运行在裸机上的第一层软件,它必然受到冯·诺依曼型计算机的结构特点的制约和影响。

冯·诺依曼型计算机是一个顺序计算模型,其根本特点是集中顺序过程控制,操作系统既然是计算机系统中的一个重要组成部分,当然也不能不反映这一特点。从操作系统的发展历程来看,可以分为早期的单用户操作系统和现代的多用户、多任务操作系统两大阶段。

早期的单用户操作系统只允许一个用户使用计算机,该用户独占计算机系统的各种资源,整个系统为其程序运行提供服务。在用户程序的处理过程中,除了 CPU 和外部设备有可能进行并行操作外,其余部分都是顺序操作。这种单用户操作系统简单明了,容易实现。但在这样的系统中,昂贵的计算机硬部件并没有得到充分利用,计算机的性能特别是资源利用率大大低于可能达到的程度。

在硬件和软件技术快速发展的今天,如何解决计算机的顺序计算模型和操作系统的并行计算模型这一对矛盾?这有两个途径:一是操作系统为适应硬件基础而采取的实现多任务并发执行的技术;二是计算机体系结构的变化以适应并行处理的需要。



1. 现代操作系统采用的技术

为了提高资源利用率,现代操作系统将单处理机系统改造成了逻辑上的多处理机系统,具有了并行处理能力。让多个用户共用一个计算机系统,这就必须解决多个用户的算题任务共享计算机系统资源的问题,也需要解决系统如何控制和协调多个任务的同时执行的问题。为此,在单 CPU 的计算机上配置的操作系统实现了一系列新的软件技术。

为支持多任务并发执行,操作系统引入进程的概念,实现对进程的描述、进程控制、进程调度的功能;协调多任务之间的关系,实现进程同步、互斥和进程通信的功能。为解决多用户、多任务共享系统资源的问题,操作系统实现了系统资源(CPU、存储器、各类外部设备及信息资源(文件))的分配机制、策略和方法。

为了解决计算机的顺序计算模型和操作系统的并行计算模型这一对矛盾,单处理机上配置的操作系统实现技术变得非常复杂、不易理解,最终使操作系统成为一个庞然大物,且效果并不一定很理想。

2. 具有并行处理能力的计算机体系结构

在单 CPU 计算机上配置的操作系统越来越复杂的情况下,人们不断研究与并行计算模型一致的计算机体系结构,出现了多处理机系统、消息传递型多计算机和计算机网络等具有并行处理能力的计算机体系结构。

(1) 多处理机系统

多处理机系统是由多个处理机、公共主存、外部设备等部件组成的系统。所有处理机共享一个公共主存,共享 I/O 通道、控制器和外部设备,多个处理机可以同时并行处理多个任务。它的特点是通过共享存储器实现多个处理机之间的互相通信,由于高度的资源共享,被称为紧耦合系统。多处理机系统存在瓶颈、可扩展性差的问题。这样的系统对并行处理的任务有一定的局限性,任务之间的数据共享应尽量少。

(2) 消息传递型多计算机

消息传递型多计算机由两台以上的计算机组成,每台计算机有自己的控制部件、本地存储器(处理机/存储器对)或 I/O 设备,按 MIMD(多指令流多数据流)模式执行程序,采用消息通信机制实现通信。消息传递型多计算机又可称为大规模并行处理机 MPP(Massively Parallel Processor)。消息传递型多计算机的各结点由定制网络连接,定制网络的结构可以是网格、环、超立方体、带环立方体结构等。

消息传递型多计算机的结构特点是:1)多个处理机/存储器对;2)分布式存储,无共享资源;3)消息传递网络,由硬件直连,传递速率高;4)可扩展性好。这种结构的并行计算机应用比较广泛。

(3) 计算机网络

计算机网络是通过通信线将独立自主的计算机互连而成的集合体。互连是指两台计算机之间彼此交换信息,可以通过导线、激光、微波、卫星等方式进行互连。独立自主指的是网络中每一台计算机都是自主的,不接受其他计算机的命令,没有主从关系。

计算机网络的特点是:1)具有多个处理部件;2)无公共主存;3)有消息通信机制。组成计算机网络的各计算机的位置可以分布在不同的地方,甚至可以相距很远。这一点与消息传递型多计算机不同。消息传递型多计算机的各结点互相分离但距离不大。当前,计算机网络的应用十分广泛,它可用于需要分布处理的大量应用中。

目前在市场上销售的计算机,大部分仍然采用冯·诺依曼式计算机结构,预计将来也仍然是如此。因此,我们一方面要学好当前计算机上配置的操作系统,另一方面也要关心计算机系统结构发展的新趋势。从计算机体系结构的角度出发去分析操作系统,就容易理解操作系统的功能和实