



教育部普通高等教育精品教材
普通高等教育“十一五”国家级规划教材

高等院校应用型人才培养规划教材

操作系统

(第3版)

王路群 主编 罗保山 周雯 副主编



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

教育部普通高等教育精品教材
普通高等教育“十一五”国家级规划教材
高等院校应用型人才培养规划教材

操作系统

(第3版)

王路群 主编
罗保山 周 雯 副主编

電子工業出版社

Publishing House of Electronics Industry

内 容 简 介

本书是教育部普通高等教育精品教材、普通高等教育“十一五”国家级规划教材。书中以当前流行操作系统的主要功能为主线,介绍操作系统的概念、接口、进程管理、处理机管理、存储器管理、文件管理、设备管理;以用户比较熟悉的 Linux 和 Windows 7 操作系统为实例,剖析了现代操作系统所采用的最新技术;附录提供习题参考答案,并为任课教师免费提供电子课件。

本书可作为高等学校计算机科学与技术、软件工程、通信与电子信息等相关专业教材和参考书,也可供从事计算机研究、开发、维护和应用的专业人员阅读。

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。

图书在版编目(CIP)数据

操作系统/王路群主编. —3 版. —北京:电子工业出版社, 2013.10

普通高等教育“十一五”国家级规划教材

高等院校应用型人才培养规划教材

ISBN 978-7-121-19393-4

I. ①操… II. ①王… III. ①操作系统—高等职业教育—教材 IV. ①TP316

中国版本图书馆 CIP 数据核字(2012)第 318318 号

策划编辑:吕 迈

责任编辑:刘真平

印 刷:北京京华虎彩印刷有限公司

装 订:北京京华虎彩印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1 092 1/16 印张:18 字数:460.8 千字

印 次:2013 年 10 月第 1 次印刷

定 价:35.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

/ 前 言 /

操作系统是计算机系统的重要组成部分，是整个计算机系统的基础和核心。它负责控制和管理整个计算机系统的资源，组织用户高效、协调地使用这些资源。操作系统也是计算机科学技术及其相关专业的一门重要的必修或选修课程。

本书是教育部普通高等教育精品教材、普通高等教育“十一五”国家级规划教材，既注重对操作系统经典知识的讲解，又紧密结合当代最新的操作系统发展趋势，按照高素质软件人才培养的要求，结合软件发展潮流，全面地向读者展现较新的操作系统的设计原理与基本实现技术，以目前流行的、用户比较熟悉的 Linux 和 Windows 7 操作系统为实例，剖析了现代操作系统所采用的最新技术，以便读者深入了解现代操作系统的全貌，为今后进行大型软件研制与系统开发打下坚实基础。

本书是编者在多年的教学实践和科学研究的基础上，参阅了大量国内外操作系统教材后，几经修改而成的，其特点及主要编写思路如下：

(1) 课程的科学体系与实用性、先进性的关系。按照操作系统的五大功能，以 Linux 和 Windows 7 操作系统为实例，全面、系统地介绍了操作系统的基本概念、基本技术和基本方法。

(2) 本书新增了“操作系统接口”一章，便于读者深入理解、掌握操作系统接口及其分类、联机命令接口、图形化用户界面的操作以及系统调用的实现等相关知识。

(3) 本书将 Linux、Windows 7 的进程与存储管理、设备管理和文件系统两部分单独编成 4 章，有针对性地分析了 Linux 和 Windows 7 的核心技术，使读者能理论联系实际，加深理解。

(4) 根据编者的教学经验，对于比较难以理解的部分，本书均以实例引出，语言深入浅出，使读者能够从简单的实例入手，轻松掌握操作系统的内部工作原理。

(5) 本书配有精选的习题，以帮助读者检验和加深对内容的理解。所有习题均附有参考答案。

本课程先修内容为：计算机基础应用、计算机组成原理和程序设计知识。

本书由王路群担任主编，罗保山、周雯担任副主编，张恒、綦志勇、王彩梅、于继武、王玮、梁晓娅、涂洪涛、李菲、龚丽、张胜洪参加编写，周雯统审全稿。

由于时间仓促，加之编者水平有限，书中不妥或错误之处在所难免，殷切希望广大读者批评指正。同时，恳请读者一旦发现错误，于百忙之中及时与编者联系，以便尽快更正，编者将不胜感激。E-mail: zhouwenwh@126.com。

编 者
2013 年 6 月

目 录

CONTENTS

第 1 章 操作系统概论	1
1.1 操作系统概述	1
1.1.1 操作系统的定义	1
1.1.2 操作系统的特性	3
1.1.3 操作系统的功能	4
1.2 操作系统的结构	8
1.2.1 整体结构	8
1.2.2 层次结构	9
1.2.3 虚拟机结构	10
1.2.4 外核结构	11
1.2.5 客户机/服务器结构	11
1.3 操作系统的形成和发展	12
1.3.1 人工操作阶段	12
1.3.2 单道批处理系统	12
1.3.3 多道批处理系统	14
1.3.4 分时系统	15
1.3.5 实时系统	16
1.3.6 个人机系统	18
1.3.7 网络操作系统	19
1.3.8 分布式操作系统	20
1.3.9 嵌入式操作系统	20
1.4 实用操作系统介绍	21
1.4.1 Windows 7 操作系统	21
1.4.2 Linux 操作系统	23
【本章小结】	25
习题	25
第 2 章 操作系统接口	26
2.1 脱机用户接口	26

2.2	联机用户接口	27
2.2.1	联机命令接口	27
2.2.2	图形化用户界面	30
2.3	系统调用	33
2.3.1	系统调用概述	33
2.3.2	系统调用的实现	35
	【本章小结】	37
	习题	37
第3章	进程管理	38
3.1	进程的引入	38
3.1.1	程序的顺序执行	39
3.1.2	程序的并发执行	39
3.1.3	进程的引入	40
3.2	进程的概念	41
3.2.1	进程	41
3.2.2	进程的状态	43
3.2.3	进程管理	44
3.2.4	进程控制块	46
3.3	进程的控制	47
3.3.1	进程控制机构	47
3.3.2	进程的创建与撤销	48
3.3.3	进程的挂起与恢复	50
3.4	进程的协调	52
3.4.1	进程互斥	52
3.4.2	进程同步	56
3.4.3	信号量和 P、V 操作	57
3.4.4	经典进程的同步	60
3.4.5	管程	64
3.5	进程的通信	66
3.5.1	进程通信的概念	66
3.5.2	进程通信的方式	66
3.5.3	消息传递系统	67
3.6	死锁	69
3.6.1	死锁的概念	69
3.6.2	死锁产生的必要条件	70
3.6.3	死锁的描述	71
3.6.4	死锁的解决方法	72
3.6.5	死锁的预防	72

3.6.6	死锁的避免	73
3.6.7	死锁的检测与解除	76
3.7	线程的概念	78
3.7.1	线程概述	78
3.7.2	多线程模型	79
3.7.3	线程与进程的区别	80
	【本章小结】	81
	习题	82
第4章	处理机管理	84
4.1	分级调度	84
4.1.1	作业的状态及其转换	84
4.1.2	调度的层次	85
4.1.3	作业与进程的关系	86
4.2	作业调度	87
4.2.1	作业调度功能	87
4.2.2	作业调度目标与性能衡量	89
4.3	进程调度	89
4.3.1	进程调度的功能和时机	89
4.3.2	进程调度方式	90
4.3.3	进程调度性能评价	91
4.4	常用调度算法	91
4.4.1	先来先服务法	92
4.4.2	时间片轮转法	92
4.4.3	优先级法	95
4.4.4	其他算法简介	96
4.5	Linux 处理机管理	97
4.5.1	Linux 进程调度机制	97
4.5.2	Linux 进程调度的实现	98
4.5.3	作业级用户接口	100
	【本章小结】	102
	习题	103
第5章	存储器管理	104
5.1	存储器管理概述	104
5.1.1	存储器的层次	104
5.1.2	程序的主要处理阶段	106
5.1.3	存储器管理的功能和任务	106
5.2	存储管理方式	109

5.2.1	连续分配方式	109
5.2.2	离散分配方式	114
5.2.3	虚拟存储系统	114
5.3	覆盖与对换	115
5.3.1	覆盖技术	115
5.3.2	对换技术	116
5.4	分页存储管理技术	116
5.4.1	分页存储管理的基本原理	116
5.4.2	地址映射	118
5.4.3	两级和多级页表	120
5.4.4	页面的共享	121
5.5	分段存储管理技术	121
5.5.1	分段存储管理的基本概念	121
5.5.2	分段存储管理的基本原理	122
5.5.3	段的保护和共享	123
5.5.4	段页式存储管理技术	123
5.6	虚拟存储器	125
5.6.1	虚拟存储器的引入	125
5.6.2	虚拟存储器的实现方法	126
5.6.3	虚拟存储器的特征	126
5.7	请求分页存储管理技术	126
5.7.1	硬件支持及缺页处理	127
5.7.2	页面置换算法	129
	【本章小结】	132
	习题	132
第 6 章	Linux 的进程与存储管理	134
6.1	Linux 的进程管理	134
6.1.1	进程与作业的基本概念	134
6.1.2	进程的管理和控制	135
6.1.3	作业的控制	138
6.2	服务的管理	140
6.2.1	图形界面下的管理	140
6.2.2	ntsysv 命令	141
6.2.3	chkconfig 和 service 命令	142
6.3	Linux 的存储管理	142
6.3.1	Linux 的分段和分页机制	143
6.3.2	Linux 内存管理机制	145
6.4	进程管理项目实训：实现计划任务	148

6.5	存储管理项目实训：磁盘分区	148
	【本章小结】	149
	习题	150
第7章	Windows 的进程与内存管理	151
7.1	Windows 7 的特点及相关的概念	151
7.1.1	Windows 7 体系结构的特点	151
7.1.2	Windows 的管理机制	152
7.2	Windows 进程和线程	154
7.2.1	Windows 进程和线程的定义	154
7.2.2	Windows 进程和线程的关联	155
7.2.3	Windows 进程的结构	156
7.2.4	Windows 线程的结构	156
7.2.5	Windows 进程和线程的创建	156
7.3	Windows 处理器调度机制	157
7.3.1	调度优先级	158
7.3.2	线程状态	158
7.3.3	线程调度机制	159
7.4	Windows 的内存管理	160
7.4.1	内存管理器	161
7.4.2	内存管理的机制	161
7.5	虚拟地址空间	164
7.5.1	虚拟地址空间布局	164
7.5.2	虚拟地址转换	164
7.6	页面调度	165
7.6.1	缺页处理	166
7.6.2	工作集及页面调度策略	166
7.6.3	页框号和物理内存管理	167
	【本章小结】	167
	习题	168
第8章	文件管理	176
8.1	文件与文件系统	176
8.1.1	文件的概念	176
8.1.2	文件的分类	177
8.1.3	文件系统概述	177
8.1.4	文件系统的功能	178
8.2	文件的组织与存储	179
8.2.1	文件的逻辑结构与存取方式	179

8.2.2	文件的物理结构	183
8.3	文件的管理	185
8.3.1	文件控制块	185
8.3.2	文件目录的概念	185
8.3.3	目录文件的结构	186
8.3.4	文件存储空间管理	188
8.4	文件的共享与安全	191
8.4.1	文件共享的实现	191
8.4.2	文件安全的管理	192
	【本章小结】	194
	习题	195
第 9 章	设备管理	196
9.1	设备与设备管理	196
9.1.1	设备的分类	196
9.1.2	设备管理的设计目标	197
9.1.3	设备管理的功能	198
9.1.4	数据传送控制方式	199
9.2	缓冲技术	201
9.2.1	缓冲技术的引入	202
9.2.2	缓冲器的种类	203
9.3	设备分配	206
9.3.1	设备管理的数据结构	206
9.3.2	设备分配的原则、策略及算法	207
9.4	I/O 进程控制	209
9.4.1	控制的引入	209
9.4.2	控制的功能	210
9.4.3	控制的实现	211
9.5	设备驱动程序	211
	【本章小结】	212
	习题	213
第 10 章	Linux 文件系统	214
10.1	Linux 文件系统概述	214
10.2	EXT2 文件系统	216
10.3	虚拟文件系统 (VFS)	221
10.4	创建 Linux 文件系统	225
10.5	挂载和卸载文件系统	226
10.6	检查文件系统错误	226

10.7 使用交换空间	227
10.8 设置开机自动挂载文件系统	227
10.9 文件系统项目实训：挂载与卸载 Linux 文件系统	228
10.10 设备管理项目实训：Linux 设备管理	228
【本章小结】	232
习题	232
第 11 章 Windows 的设备管理和文件系统	233
11.1 Windows I/O 系统的结构	233
11.1.1 设计目标	233
11.1.2 设备管理服务	234
11.2 设备驱动程序和 I/O 处理	235
11.2.1 设备驱动类型和结构	235
11.2.2 Windows 的 I/O 处理	236
11.3 Windows 的文件系统	238
11.3.1 Windows 磁盘管理	239
11.3.2 Windows 文件系统格式	239
11.3.3 Windows 文件系统驱动	241
11.4 NTFS 文件系统	242
11.4.1 NTFS 的特点	242
11.4.2 NTFS 的磁盘结构	245
11.4.3 NTFS 的文件系统恢复	246
【本章小结】	247
习题	248
附录 A 习题答案	254
第 1 章习题答案	254
第 2 章习题答案	255
第 3 章习题答案	257
第 4 章习题答案	264
第 5 章习题答案	266
第 6 章习题答案	269
第 8 章习题答案	270
第 9 章习题答案	272
第 10 章习题答案	274

操作系统概论

【本章学习目标】

- 了解：操作系统的发展历程、操作系统的设计结构、常见操作系统的特点。
- 理解：操作系统的形成和基本类型（批处理操作系统、分时操作系统、实时操作系统、网络操作系统和分布式操作系统、嵌入式操作系统）。
- 掌握：操作系统的定义和五大主要功能（处理机管理、存储器管理、设备管理、文件管理、用户接口管理）；批处理系统、分时系统和实时系统的特点。

【本章学习要点】

应深入理解、掌握什么是操作系统，它是如何形成的，主要具有什么功能，有哪些类型；与其他软件相比，操作系统有怎样的特殊性，有什么基本特征。同时可以结合上机操作，考虑操作系统的功能是如何实现的等问题。

1.1 操作系统概述

计算机是由 CPU、内存、磁盘、显卡、声卡等许多设备组成的，而且这些设备的厂商品种繁多，而且不同厂商生产的同种设备虽然完成同种功能，但是具体细节却千差万别。

为了正确地管理和使用这些设备来实现具体的应用，程序员就得了解和掌握各种设备的工作原理，可以想象这是非常复杂、困难的工作。

多年的研究和发展得出了解决方法：在硬件的基础上加载一层软件来管理整个系统。这个软件通过设备驱动程序来与计算机硬件打交道，通过一系列的功能模块将整个计算机硬件系统抽象成为一个公共、统一、开放的接口，从而使得程序员不必再陷入各种硬件系统的具体细节！这一层软件就是操作系统。

1.1.1 操作系统的定义

计算机系统可以粗分为四个部分：硬件、操作系统、应用程序和用户（见图 1.1）。

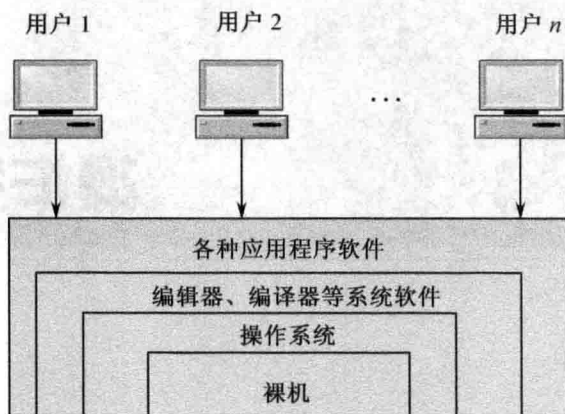


图 1.1 计算机系统的层次关系

硬件，如中央处理单元（Central Processing Unit, CPU）、内存（Memory）、输入/输出设备（Input/Output Device, I/O Device），提供基本的计算资源。

应用程序，如电子制表软件、编译器、网络浏览器，规定了按何种方式使用这些资源来解决用户的计算机问题。操作系统控制和协调各用户的应用程序对硬件的使用。

由图 1.1 可看出，计算机的硬件和软件之间是一种层次结构的关系。裸机在最里层，它的外面是操作系统，各种应用软件运行在操作系统之上，它们以操作系统为支撑环境，同时又向用户提供完成其作业所需的各种服务。

综上所述，可以将操作系统定义为：

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，合理地组织计算机的工作流程，控制程序的执行并向用户提供各种服务功能，使得用户能够灵活、方便、有效地使用计算机，使整个计算机系统能高效地运行。它是计算机与用户之间的接口。

操作系统是一个大型的软件系统，其功能复杂，体系庞大。从不同的角度看，结果也不同，正是“横看成岭侧成峰”，下面我们从系统和用户两个角度来分析一下引入操作系统的目的。

1. 用户视角

计算机的用户观点因所使用接口的不同而异。绝大多数计算机用户坐在一台这样的 PC 前，PC 由显示器、键盘、鼠标和主机组成。这类系统设计是为了让单个用户单独使用其资源，其目的是优化用户所进行的工作（或游戏）。对于这种情况，操作系统的设计目的是为了使用方便，性能是次要的，而且不在乎资源使用率——如何共享硬件和软件资源。

近来，各种手持计算机开始成为时尚。绝大多数这些设备为单个用户所独立使用。有的也通过有线或无线（更为常见）与网络相连。由于受电源、速度和接口所限，它们只能执行相对较少的远程操作。绝大多数这类操作系统的设计目的是为了个人使用，当然如何在有限的电池容量中发挥最大的效用也很重要。

有的计算机几乎没有或根本没有用户观点。例如，在家电和汽车中所使用的嵌入式计

算机可能只有键盘,只能打开和关闭指示灯来显示状态,而且这些设备及其操作系统通常设计成无须用户干预就能自行运行。

2. 系统视角

从计算机的角度来看,操作系统是与硬件最为密切的程序。计算机系统可能有许多资源,用来解决 CPU 时间、内存空间、文件存储空间、I/O 设备等问题。操作系统管理这些资源。面对许多甚至冲突的资源请求,操作系统必须决定如何为各个程序和用户分配资源,以便计算机系统能有效而公平地运行。因此,操作系统是计算机资源的管理者。

1.1.2 操作系统的特性

操作系统种类繁多,但却有一些共同特征,这些特征也是操作系统这一软件层次与应用软件的区别所在。现代操作系统都具有并发、共享、虚拟和异步特性,其中并发性是操作系统最重要的特征,其他三个特性均基于并发性而存在。

1. 并发性

并行性和并发性是既相似又有区别的两个概念,并行性是指两个或多个事件在同一时刻发生;而并发性是指两个或多个事件在同一时间间隔内发生。

在多道程序环境下,并发性是指在一段时间内,宏观上有多个程序在同时运行,但在单处理机系统中,每一时刻却仅能有一道程序执行,故微观上这些程序只能是分时地交替执行。倘若在计算机系统中有多个处理机,则这些可以并发执行的程序便可被分配到多个处理机上,实现并行执行,即利用每个处理机来处理一个可并发执行的程序,这样,多个程序便可同时执行。

操作系统必须具备控制和管理各种并发活动的的能力。

2. 共享性

在操作系统环境下,所谓共享是指系统中的资源可供内存中多个并发执行的进程(线程)共同使用。由于资源属性的不同,进程对资源共享的方式也不同,目前主要有以下两种资源共享方式。

(1) 互斥共享方式:系统中的某些资源,如打印机、磁带机,虽然它们可以提供给多个进程(线程)使用,但规定在一段时间内只允许一个进程(线程)访问该资源。为此,当一个进程 A 要访问某资源时,必须先提出请求,如果此时该资源空闲,系统便可将之分配给请求进程 A 使用,此后若再有其他进程也要访问该资源(只要 A 未用完),则必须等待。仅当 A 进程访问完并释放该资源后,才允许另一进程对该资源进行访问。我们把这种资源共享方式称为互斥式共享,而把在一段时间内只允许一个进程访问的资源称为临界资源或独占资源。

(2) 同时访问方式:系统中还有另一类资源,允许在一段时间内由多个进程“同时”对它们进行访问。这里所谓的“同时”往往是宏观上的,而在微观上,这些进程可能是交替地对该资源进行访问。典型的可供多个进程“同时”访问的资源是磁盘设备,一些用重入码编写的文件,也可以被“同时”共享,即若干个用户同时访问该文件。

并发和共享是操作系统的两个最基本的特征，它们又是互为存在的条件。一方面，资源共享是以程序（进程）的并发执行为条件的，若系统不允许程序并发执行，自然不存在资源共享问题；另一方面，若系统不能对资源共享实施有效管理，协调好各进程对共享资源的访问，也必然影响到程序并发执行的程度，甚至根本无法并发执行。

3. 虚拟性

操作系统中的所谓“虚拟”，是指通过某种技术把一个物理实体变为若干个逻辑上的对应物。物理实体（前者）是实的，即实际存在的；而后者是虚的，是用户感觉上的东西。相应地，用于实现虚拟的技术，称为虚拟技术。在操作系统中利用了多种虚拟技术，分别用来实现虚拟处理机、虚拟内存、虚拟外部设备和虚拟信道等。

在虚拟处理机技术中，是通过多道程序设计技术，让多道程序并发执行的方法，来分时使用一台处理机的。此时，虽然只有一台处理机，但它能同时为多个用户服务，使每个终端用户都认为是有一个 CPU 在专门为他服务。亦即，利用多道程序设计技术，把一台物理上的 CPU 虚拟为多台逻辑上的 CPU，也称为虚拟处理机，我们把用户所感觉到的 CPU 称为虚拟处理器。

类似地，可以通过虚拟存储器技术，将一台机器的物理存储器变为虚拟存储器，以便从逻辑上来扩充存储器的容量。此时，虽然物理内存的容量可能不大（如 32MB），但它可以运行比它大得多的用户程序（如 128MB）。这使用户所感觉到的内存容量比实际内存容量大得多，认为该机器的内存至少也有 128MB。当然这时用户所感觉到的内存容量是虚的。我们把用户所感觉到的存储器称为虚拟存储器。

我们还可以通过虚拟设备技术，将一台物理 I/O 设备虚拟为多台逻辑上的 I/O 设备，并允许每个用户占用一台逻辑上的 I/O 设备，这样便可使原来仅允许在一段时间内由一个用户访问的设备（临界资源），变为在一段时间内允许多个用户同时访问的共享设备。例如，原来的打印机属于临界资源，而通过虚拟设备技术，可以把它变为多台逻辑上的打印机，供多个用户“同时”打印。此外，也可以把一条物理信道虚拟为多条逻辑信道（虚信道）。在操作系统中，虚拟的实现主要是通过分时使用的方法。显然，如果 n 是某物理设备所对应的虚拟的逻辑设备数，则虚拟设备的平均速度必然是物理设备速度的 $1/n$ 。

4. 异步性

异步性是指操作系统中的各个程序的推进次序无法预知。异步性的产生是由现代操作系统的并发性引起的。在并发执行的多个进程之间，何时能够获得所需资源，何时等待哪些进程释放资源，以及当前占有资源的进程何时释放资源等因素都是不确定的，因此用户无法预知各个进程的执行时间。异步性是现代操作系统的一个重要特征。

1.1.3 操作系统的功能

1. 处理机管理功能

CPU 是整个计算机系统的核心硬件资源。它的性能和使用情况对整个计算机系统的性能有关键的影响。CPU 是较为昂贵的资源，它的速度一般比其他硬件设备的工作速

度要快得多,其他设备的正常运行往往也离不开 CPU。因此,有效地管理 CPU,充分利用 CPU 资源也是操作系统最重要的管理任务。

在多道程序的环境中,CPU 分配的主要对象是进程(或线程),操作系统通过选择一个合适的进程占有 CPU 来实现对 CPU 的管理,因此,对 CPU 的管理归根结底就是对进程的管理。操作系统有关进程方面的管理任务很多,主要有作业和进程调度、进程控制、进程通信等。

1) 作业和进程调度

一个作业通常要经过两级调度才得以在 CPU 上执行。首先是作业调度,它从后备队列中按照一定的算法,选中一批作业放入内存,并为它们分配必需的资源,在将它们调入内存后,为这些作业建立相应的进程,使它们都成为可能获得处理机的就绪进程,然后进程调度按一定的算法从就绪进程中选出一个合适进程,使之在 CPU 上运行。

2) 进程控制

进程控制的主要功能是为作业创建进程,撤销已结束的进程,以及控制进程在运行过程中的状态转换。

在传统的多道程序环境下,要使作业运行,必须先为它创建一个或几个进程,并为之分配必要的资源。当进程运行结束时,立即撤销该进程,以便能及时回收该进程所占用的各类资源。在现代操作系统中,进程控制还应具有一个进程创建若干个线程的功能和撤销(终止)已完成任务的线程的功能。

3) 进程通信

在多道程序环境下,为了加速应用程序的运行,应在系统中建立多个进程,并且再为一个进程建立若干个线程,由这些进程(线程)相互合作去完成一个共同的任务。而在这些进程(线程)之间,又往往需要交换信息。例如,有三个相互合作的进程,它们是输入进程、计算进程和打印进程。输入进程负责将所输入的数据传送给计算进程;计算进程利用输入数据进行计算,并把计算结果传送给打印进程;最后,由打印进程把计算结果打印出来。进程通信的任务就是用来实现在相互合作的进程之间的信息交换。

当相互合作的进程(线程)处于同一计算机系统时,通常在它们之前是采用直接通信方式,即由源进程利用发送命令直接将消息(message)挂到目标进程的消息队列上,以后由目标进程利用接收命令从其消息队列中取出消息。当相互合作的进程处于不同的系统中时,常采用间接通信方式,即由源进程利用发送命令将消息送入中间实体——邮箱(相应通信系统称为电子邮件系统),以后由目标进程从中取走消息。

2. 存储器管理功能

存储器可以说是一种最重要的系统资源,存储器对作业的重要程度就如同土地对于人类。一个作业要在 CPU 上运行,它的代码和数据就要全部或部分地驻在内存中。操作系统也要占据相当大的内存空间。在多道程序系统中,并发运行的程序都要占有自己的内存空间,因此内存空间总是一种紧张的系统资源。存储管理的任务主要有:

1) 内存分配与回收

操作系统在实现内存分配时,可采取静态和动态两种方式:

(1) 在静态分配方式中, 每个作业的内存空间是在作业装入时确定的; 在作业装入后的整个运行期间, 不允许该作业再申请新的内存空间, 也不允许作业在内存中“移动”。

(2) 在动态分配方式中, 每个作业所要求的基本内存空间, 也是在装入时确定的, 但允许作业在运行过程中, 继续申请新的附加内存空间, 以适应程序和数据的动态增涨, 也允许作业在内存中“移动”。

为了实现内存分配, 在内存分配的机制中应具有这样的结构和功能:

- ① 内存分配数据结构, 该结构用于记录内存空间的使用情况, 作为内存分配的依据;
- ② 内存分配功能, 系统按照一定的内存分配算法, 为用户程序分配内存空间;
- ③ 内存回收功能, 系统对于用户不再需要的内存, 通过用户的释放请求, 去完成系统的回收功能。

2) 内存保护

内存保护的主要任务, 是确保每道用户程序都只在自己的内存空间内运行, 彼此互不干扰。

为了确保每道程序都只在自己的内存区中运行, 必须设置内存保护机制。一种比较简单的内存保护机制, 是设置两个界限寄存器, 分别用于存放正在执行程序的上界和下界。系统须对每条指令所要访问的地址进行检查, 如果发生越界, 便发出越界中断请求, 以停止该程序的执行。如果这种检查完全用软件实现, 则每执行一条指令, 便须增加若干条指令去进行越界检查, 这将显著降低程序的运行速度。因此, 越界检查都由硬件实现。当然, 对发生越界后的处理, 还须与软件配合来完成。

3) 地址映射

一个应用程序(源程序)经编译后, 通常会形成若干个目标程序; 这些目标程序再经过链接便形成了可装入程序。这些程序的地址都是从“0”开始的, 程序中的其他地址都是相对于起始地址计算的; 由这些地址所形成的地址范围称为“地址空间”, 其中的地址称为“逻辑地址”或“相对地址”。此外, 由内存中的一系列单元所限定的地址范围称为“内存空间”, 其中的地址称为“物理地址”。

在多道程序环境下, 每道程序不可能都从“0”地址开始装入(内存), 这就致使地址空间内的逻辑地址和内存空间中的物理地址不一致。使程序能正确运行, 存储器管理必须提供地址映射功能, 以将地址空间中的逻辑地址转换为内存空间中与之对应的物理地址。该功能应在硬件的支持下完成。

4) 内存扩充

存储器管理中的内存扩充任务, 并非是去扩大物理内存的容量, 而是借助于虚拟存储技术, 从逻辑上去扩充内存容量, 使用户所感觉到的内存容量比实际内存容量大得多; 或者是让更多的用户程序能并发运行。这样, 既满足了用户的需要, 改善了系统的性能, 又基本上不增加硬件投资。为了能在逻辑上扩充内存, 系统必须具有内存扩充机制, 用于实现下述各功能:

- ① 请求调入功能。
- ② 置换功能。