



计 算 机 类 本 科 规 划 教 材

操作系统教程

◆ 屠立忠 徐金宝 主编

◆ 刘晓璐 丁宋涛 王 洋 副主编

◆ 黄陈蓉 主审



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

013066775

TP316-43
90

计算机类本科规划教材

操作系统教程

主 编 屠立忠 徐金宝

副主编 刘晓璐 丁宋涛 王 洋

主 审 黄陈蓉



電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING



北航

C1674612

TP316-43
90

内 容 简 介

操作系统是管理和控制计算机硬件与软件资源的计算机程序,是计算机系统的核心系统软件。操作系统课程是计算机类专业的核心课程和必修课程,操作系统的原理、相关管理技术和调度策略是计算机应用系统开发专业人员必须掌握的专业知识。

本书共分6章,第1章为操作系统概论,第2~5章分别介绍操作系统中处理器管理及并发进程、存储管理、设备管理和文件管理等内容,第6章介绍 Windows 和 Linux 操作系统。为方便读者学习,每章末尾均附有小结和习题。附录部分是验证和重现操作系统基本理论的实验和课程设计项目,供读者选用。

本书面向应用型本科学生,原理和理论叙述简单明了,实例分析联系实际,全书内容易于理解和掌握。本书可作为计算机类应用型本科专业操作系统相关课程的教材或参考书,也可作为从事操作系统原理研究与系统开发的工程技术人员的参考用书。

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

操作系统教程 / 屠立忠, 徐金宝主编. —北京: 电子工业出版社, 2013.8

计算机类本科规划教材

ISBN 978-7-121-20509-5

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

中国版本图书馆 CIP 数据核字 (2013) 第 109756 号

策划编辑: 章海涛

责任编辑: 郝黎明

印 刷: 涿州市京南印刷厂

装 订: 涿州市京南印刷厂

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: 16 字数: 409.6 千字

印 次: 2013 年 8 月第 1 次印刷

定 价: 36.00 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 zlts@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一个重要组成部分,诞生已有 60 多年的历史,发展非常迅猛,新的实用操作系统不断涌现并逐步演变成行业主流。操作系统课程是计算机类专业的重要专业基础课程之一,对计算机行业的从业人员来说,深入学习和掌握操作系统的相关技术是必须完成的任务。操作系统的教科书版本非常多,但适合应用型计算机类专业本科人才培养的教材不多,大部分教材的理论性较强,实践性不足,不利于工程实践能力的培养。

本书是作者在多年从事应用型本科专业教学工作基础上,结合国内外书面和网络资料编写而成的,讲义简明阐述了操作系统的原理和相关技术,并增加了大量工程实例分析,旨在传授理论知识的同时,培养读者的工程应用和工程实践能力。

本教材的建议学时数为 64 学时,其中理论学时 50~54 学时,实验学时 10~14 学时。实际教学时,可对理论教学和实验学时进行适当增减,并根据教学的需要设置课程设计环节。

全书共分 6 章。第 1 章是操作系统概论,介绍操作系统的定义和目标、操作系统的历史和发展、操作系统的特性和基本结构;第 2 章阐述进程、线程概念和处理器管理的基本原理;第 3 章主要阐述存储管理技术,包括实存管理和虚存管理技术,重点分析了分页和分段管理的方法;第 4 章主要介绍设备管理子系统和设备管理技术,重点分析了磁盘驱动调度策略;第 5 章阐述文件管理技术,重点分析了文件的逻辑结构和物理结构,介绍文件的使用方法;第 6 章详细介绍 Windows 和 Linux 两种操作系统中应用的管理技术以及基本的使用技巧。为便于读者自学和课后复习,每章都附有小结和习题。附录 A 为操作系统的基本实验项目和创新实验项目,附录 B 为操作系统课程设计的案例,供教师在实际教学或读者学习过程中选择使用或参考。

本书由屠立忠主编和统稿。徐金宝、刘晓璐、丁宋涛分别参加了第 2 章和第 4 章、第 3 章和第 5 章、第 6 章的编写工作,齐齐哈尔大学的王洋也参加了本书的编写工作。中软国际、东软集团和江苏万和计算机培训中心为本书提供了部分案例,朱硕懋、吴敏、张程远、吕春龙等协助调试了本书中的案例代码,并参与了资料的整理工作,再此也向他们表示感谢。黄陈蓉教授审阅了全部书稿,并提出了许多宝贵意见和建议,在此表示感谢。此次编写工作得到了南京工程学院教务处和计算机工程学院的大力支持。

在本书的编写过程中,参考了一些有关操作系统的书刊及文献资料,并查阅了大量的网络资料,在此对所有的作者表示感谢。限于水平,书中难免有不足与疏漏之处,恳请广大读者批评指正。

联系邮箱: tulz@njit.edu.cn。

作 者

目 录

第 1 章 操作系统概论	1
1.1 操作系统的概念	1
1.1.1 操作系统的定义和目标	1
1.1.2 操作系统的形成和发展	2
1.1.3 操作系统的主要特性	5
1.2 操作系统的功能	6
1.2.1 处理器管理	6
1.2.2 存储管理	6
1.2.3 设备管理	7
1.2.4 文件管理	8
1.3 操作系统的接口	8
1.3.1 操作接口和操作命令	9
1.3.2 程序接口与系统调用	9
1.4 操作系统的结构	10
1.4.1 整体式结构	10
1.4.2 层次式结构	11
1.4.3 客户/服务器结构	11
1.4.4 虚拟机结构	11
1.5 流行操作系统简介	12
1.5.1 Windows 操作系统	12
1.5.2 UNIX 操作系统	12
1.5.3 Linux 操作系统	13
1.5.4 iOS 操作系统	13
1.5.5 Android 操作系统	14
本章小结	15
习题 1	15
第 2 章 处理器管理及并发进程	17
2.1 多道程序设计	17
2.1.1 程序的顺序执行	17
2.1.2 程序的并发执行	18
2.1.3 多道程序设计	18
2.1.4 并发程序执行的条件	19
2.2 进程	20

2.2.1	进程的定义及其属性	20
2.2.2	进程的状态及其转换	21
2.2.3	进程控制块	23
2.2.4	进程队列	24
2.3	进程的控制	25
2.3.1	操作系统内核	25
2.3.2	原语	26
2.3.3	进程控制原语	26
2.4	进程调度	28
2.4.1	进程调度简介	28
2.4.2	进程调度的算法	28
2.5	线程及其实现	32
2.5.1	为什么要引入多线程概念	33
2.5.2	多线程环境下的进程和线程	33
2.5.3	多线程的优点及其应用	35
2.5.4	多线程实现的三种方式	36
2.5.5	Java 环境下多线程设计举例	37
2.6	并发进程的概念	38
2.6.1	相关进程及其关系	38
2.6.2	与时间有关的错误	39
2.6.3	临界区概念及其管理要求	41
2.6.4	临界区管理的尝试	42
2.6.5	信号量与 PV 操作	46
2.7	进程的互斥和同步	47
2.7.1	进程的互斥	47
2.7.2	进程的同步	50
2.7.3	进程互斥和同步的关系	59
2.8	管程	60
2.8.1	管程的概念	60
2.8.2	管程的特点及其组成	61
2.8.3	汉森 (Brinch Hansen) 方法实现管程	62
2.8.4	汉森方法实现管程的实例	63
2.9	进程通信	66
2.9.1	消息传递概述	66
2.9.2	消息传递的两种方式	67
2.9.3	消息传递应用举例	69
2.9.4	信号量机制、管程和消息传递机制的关系	70
2.10	死锁	70
2.10.1	死锁的定义与产生的原因	70

2.10.2 死锁的防止	72
2.10.3 死锁的避免与银行家算法	73
2.10.4 死锁的检测与解除	77
本章小结	78
习题 2	79
第 3 章 存储管理	81
3.1 存储系统的基本概念	81
3.2 存储管理的基本概念	82
3.2.1 存储管理的功能	82
3.2.2 内存扩充技术	83
3.2.3 存储管理的分类	84
3.3 分区存储管理	85
3.3.1 单一连续区	86
3.3.2 固定分区存储管理	86
3.3.3 可变分区存储管理	88
3.4 简单分页存储管理	94
3.4.1 基本原理	94
3.4.2 地址转换	95
3.4.3 相联存储器和快表	96
3.4.4 页的分配与回收	97
3.4.5 页的共享和保护	98
3.4.6 两级和多级页表	98
3.5 简单分段存储管理	100
3.5.1 基本原理	100
3.5.2 地址转换	101
3.5.3 段的共享	102
3.5.4 分段存储管理的优点	103
3.6 虚拟存储管理	104
3.7 请求分页虚拟存储管理	104
3.7.1 基本原理	105
3.7.2 主存页面分配策略	106
3.7.3 页面调入策略	107
3.7.4 页面置换算法	107
3.7.5 缺页中断率分析	109
3.8 请求分段虚拟存储管理	111
3.8.1 基本原理	111
3.8.2 地址转换	112
3.8.3 段的动态链接	112
3.8.4 段的动态增长	113

3.9	请求段页式虚拟存储管理	113
	本章小结	113
	习题 3	113
第 4 章	设备管理	116
4.1	设备管理概述	116
4.1.1	设备管理的任务与目标	116
4.1.2	设备管理的功能	117
4.1.3	外部设备的分类	117
4.2	设备 I/O 控制方式	119
4.2.1	程序查询方式	119
4.2.2	中断方式	120
4.2.3	DMA (直接内存存取) 方式	120
4.2.4	I/O 通道控制方式	121
4.3	设备 I/O 软件原理	123
4.3.1	I/O 软件的目标	124
4.3.2	I/O 中断处理程序	124
4.3.3	设备驱动程序	125
4.3.4	与设备无关的 I/O 软件	125
4.3.5	用户级的 I/O 软件	125
4.4	缓冲技术	126
4.5	外围的设备分配、回收与启动	128
4.5.1	设备类相对号和绝对号	128
4.5.2	外部设备的分配和回收	129
4.6	磁盘驱动调度	130
4.6.1	磁盘结构	131
4.6.2	磁盘调度	132
4.6.3	磁盘移臂调度	133
4.6.4	磁盘的旋转调度	134
4.7	虚拟设备	136
4.7.1	脱机工作方式	136
4.7.2	Spooling 技术	137
4.7.3	虚拟设备	138
	本章小结	138
	习题 4	139
第 5 章	文件管理	141
5.1	文件系统	141
5.1.1	文件和文件系统	141
5.1.2	文件的分类	142

5.1.3	文件系统的功能	143
5.2	文件目录	143
5.2.1	目录内容	144
5.2.2	目录结构	144
5.3	文件结构与存取方法	147
5.3.1	文件的逻辑结构及存取方法	147
5.3.2	文件的物理结构及存取方法	148
5.3.3	存储空间管理	151
5.4	文件的使用	152
5.4.1	文件访问	152
5.4.2	文件控制	154
5.4.3	目录管理	155
5.4.4	文件的共享	155
5.5	安全性和保护	156
5.5.1	文件的访问权限	156
5.5.2	文件的存取控制	156
5.5.3	文件的完整性	158
	本章小结	158
	习题 5	158
第 6 章	Windows 和 Linux 操作系统	160
6.1	WINDOWS 2000/XP 操作系统	160
6.1.1	Windows 2000/XP 简介	160
6.1.2	Windows 操作系统总体架构	161
6.1.3	用户模式和内核模式	165
6.1.4	Windows 操作系统和应用程序	167
6.2	LINUX 操作系统	171
6.2.1	Linux 简介	171
6.2.2	Linux Shell	172
6.2.3	vi 文本编辑器	174
6.2.4	gcc 编译器和 gdb 调试	177
	本章小结	181
	习题 6	182
附录 A	操作系统参考实验项目	183
第一部分	基本实验部分	183
实验 1	进程管理	183
实验 2	进程调度	184
实验 3	存储管理	185
实验 4	独占设备的静态分配模拟	185

实验 5 文件管理	186
第二部分 创新实验部分 (Linux)	187
实验 6 more 命令实现	187
实验 7 文件列表命令实现	191
实验 8 文件系统路径命令实现	197
实验 9 操作系统引导程序的实现	203
实验 10 操作系统内核加载程序的实现	210
附录 B 操作系统课程设计参考案例	217
案例 1 多线程编程解决进程间同步和互斥问题的实现	217
案例 2 固定分区/可变分区管理算法的模拟	220
案例 3 页面置换算法的模拟	221
案例 4 银行家算法的模拟	228
案例 5 移动臂调度算法的模拟	233
案例 6 一个简单文件管理器的实现	239
参考文献	245

第1章 操作系统概论

现代计算机系统都是由硬件和软件组成的。操作系统是配置在计算机硬件上的第一层软件，是对硬件的扩充，用于控制硬件的工作、管理计算机系统的各种资源，并为系统中各种程序的运行提供服务。本章首先介绍操作系统的概念、定义、目标和发展历史，然后简单介绍操作系统的功能以及操作系统提供的服务和接口，分析操作系统的基本结构，本章最后介绍几种流行操作系统的基本情况。

1.1 操作系统的概念

一个完整的计算机系统由硬件资源和软件资源组成，其中硬件资源包括中央处理器、存储器 and 外部设备等机械装置和电子部件，它们共同构成了计算机系统运行软件和实现各种功能的物质基础；软件资源包括系统软件、支撑软件和应用软件。

操作系统是所有软件中最基础和最核心的部分，是控制和管理计算机系统各种硬件和软件资源，有效地组织多道程序运行的系统软件，是计算机用户和计算机硬件之间的中介程序，它为用户执行程序提供更方便、更有效的环境。

1.1.1 操作系统的定义和目标

操作系统作为控制应用程序执行的系统软件，已经存在很多年，其功能和内涵也在不断丰富和扩充，所以至今仍无法给出一个严格和统一的定义。但比较公认的定义是：管理系统资源、控制程序执行、改善人机界面、提供各种服务，合理组织计算机工作流程和为用户方便而有效地使用计算机提供良好运行环境的最基本的系统软件。

任何一种计算机系统，均需配备操作系统，有的系统还同时配备了两种或两种以上的操作系统，因而操作系统是现代计算机系统不可分割的重要组成部分，它为人们营造各种以计算机为核心的应用环境奠定了坚实的基础。人们使用操作系统，最直截了当的目标是让用户更加有效和方便地使用计算机，同时也希望能充分发挥计算机硬件系统的最大效用，提高工作效率。操作系统的主要目标可归结为以下几个。

1. 方便使用

操作系统通过对外提供各种接口，尽可能简化用户操作，提高计算机系统的易用性。例如，用户可以直接输入命令或单击屏幕上显示的菜单，操作程序的运行和计算机的使用；而计算机软件开发人员可以在程序中利用系统调用直接对磁盘的文件或外部设备上检测数据进行读/写操作。

2. 扩充功能

操作系统通过适当的管理机制和提供新的服务来扩大机器的功能。例如，操作系统可以采用虚拟机技术为用户提供不同的运行模拟环境和平台，采用虚拟存储管理技术为用户提供比实际内

存大得多的运行存储空间, 采用 Spooling 技术将独占设备模拟成共享设备等。

3. 管理资源

操作系统应配置管理计算机系统中所有的软硬件资源的机制。如操作系统可以按照用户和程序的要求分配各种软硬件资源, 然后在用户和程序不再使用时回收这些资源, 以供下次重新分配。

4. 提高效率

操作系统应合理组织计算机的工作流程, 改善系统性能并提高系统效率。如采用多道程序技术实现多进程的并发执行, 提高处理器等系统资源的使用效率; 提供多线程技术降低多进程并发执行时系统频繁切换所产生的管理开销。

5. 开放环境

操作系统应遵循国际和行业标准来设计, 构筑一个开放的环境, 以便使用者共享应用软件和资源。国际和行业标准包括系统平台标准、通信标准和用户接口标准等, 遵循这些标准可以解决各种应用的运行兼容性问题; 支持应用程序在不同的平台上的可移植性。

1.1.2 操作系统的形成和发展

自 1946 年诞生第一台计算机至今, 计算机经历了 60 多年的发展时期, 操作系统伴随计算机硬件的发展及应用的日益广泛而发展。最初的计算机系统上没有操作系统, 软件的概念也不明确。随着处理器集成技术、中断技术和通道技术等硬件技术的不断发展, 促进了软件概念的形成, 从而也推动了操作系统的形成和发展。而操作系统等软件的发展反过来也促进了硬件的发展。粗略地说, 操作系统的发展是由人工操作阶段过渡到早期批处理阶段而具有其雏形, 而后发展到多道程序系统时才逐步完善的。

1. 人工操作阶段

早期的计算机运算速度慢, 可用资源少, 系统只支持机器语言或汇编语言, 因而没有操作系统, 由单个用户独占计算机。程序员通过卡片或纸带将程序和数据输入计算机, 运行结果显示在屏幕上, 或者穿孔于卡片或纸带上。人工操作方式的特点如下。

(1) 用户独占系统。用户使用计算机时独占全部机器资源, 计算机资源的利用率和计算机运行效率极其低下。

(2) 人工介入多。程序员全程介入计算过程的输入、运算和输出等阶段, 自动化程度较低, 出错的机会较高。计算机的使用者通常是计算机专业技术人员, 非专业人员难以操作。

(3) 计算时间长。整个计算过程中, 数据的输入、程序的执行和结果的输出均是联机进行的, 每个环节还必须进行校对, 因而计算时间很长, 浪费了大量的人力。

人工操作方式的计算过程如图 1-1 所示。

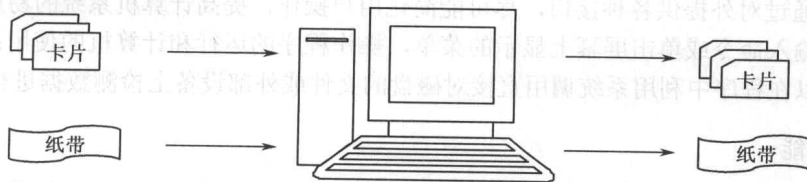


图 1-1 人工操作方式的计算过程

2. 批处理系统阶段

早期批处理系统借助于作业控制语言 (Job Control Language, JCL) 对人工操作方式进行了变革。用户可以通过脱机方式控制和应用计算机, 通过作业控制卡来描述对作业的加工和控制步骤, 并把作业控制卡连同程序、数据一起提交给操作员, 操作员收集到一批作业后一起把它们放到卡片机上输入计算机; 计算机则运行一个驻留内存的执行程序, 以对作业进行自动控制和成批处理。显然, 这种系统能实现作业到作业的自动转换, 缩短作业的准备和创建时间, 减少人工操作和人工干预, 提高了计算机的使用效率。

早期批处理系统中, 作业的输入和输出均是联机实现的, I/O 设备和 CPU 是串行工作的, CPU 利用率较低。为了解决这个问题, 在批处理系统中引进了脱机 I/O 技术, 方法是除主机外另设一台辅机, 辅机的主要功能是与 I/O 设备打交道, 需要进行输入操作时, 输入设备上的作业通过辅机记录到磁带上 (脱机输入); 主机可以把磁带上的作业读入内存执行, 作业计算完成后, 主机将结果记录到磁带上; 接下来, 辅机可以读出磁带上的结果, 控制打印机输出结果。可以看出, 主机和辅机是可以并行工作的, 程序的处理和数据的输入输出速度明显提高, 这种技术就是假脱机 I/O 技术, 假脱机技术显然使批处理系统效率大大提高。

为了进一步提高批处理系统的效率, 计算机主机中逐渐加载一些管理程序, 这些管理程序的功能包括自动控制和处理作业流, 设备驱动和输入输出控制, 程序加载和装配, 以及简单的文件管理等。这些管理程序丰富了输入输出设备类型, 并对进入系统的程序和数据进行了有效的管理, 从而缩短了作业的准备和创建时间, 充分发挥批处理系统的性能。这些管理程序就形成了操作系统的雏形。

批处理系统的计算过程如图 1-2 所示。

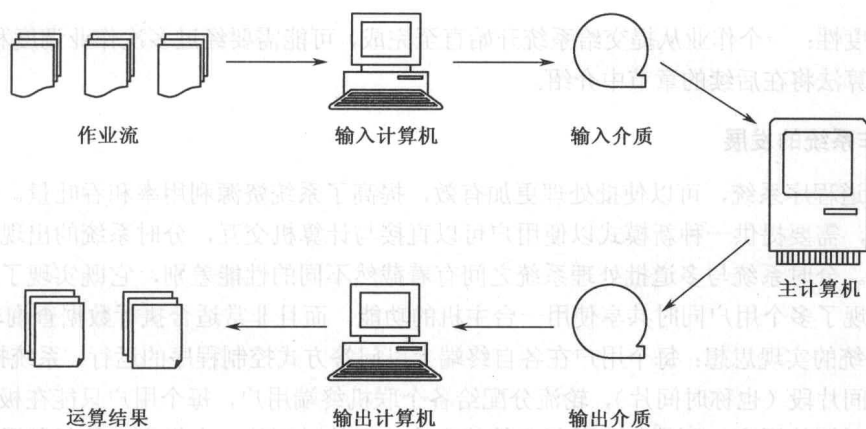


图 1-2 批处理系统的计算过程

3. 多道程序系统阶段

20 世纪 60 年代初, 中断和通道等两项技术取得了突破, 它们的结合为实现 CPU 和 I/O 设备的并行工作提供了硬件基础。多道程序系统是指在主存中存放多道用户的作业, 这些作业可以共享系统资源并交替计算。从宏观上说, 这些作业都处在运行状态而尚未完成, 因而这些作业可以并发执行; 而从微观来说, 这些作业又是串行执行的, 因为任一时刻只有一个作业在使用 CPU 运算。

严格地说, 早期的多道程序系统仍旧属于批处理系统。引入多道程序设计技术的根本目的是

提高 CPU 的利用率,充分发挥 CPU 和 I/O 设备的并行性,现代计算机系统一般都采用了多道程序设计技术。程序之间、设备之间、设备和 CPU 之间均可以并行工作。

多道程序设计系统如图 1-3 所示。

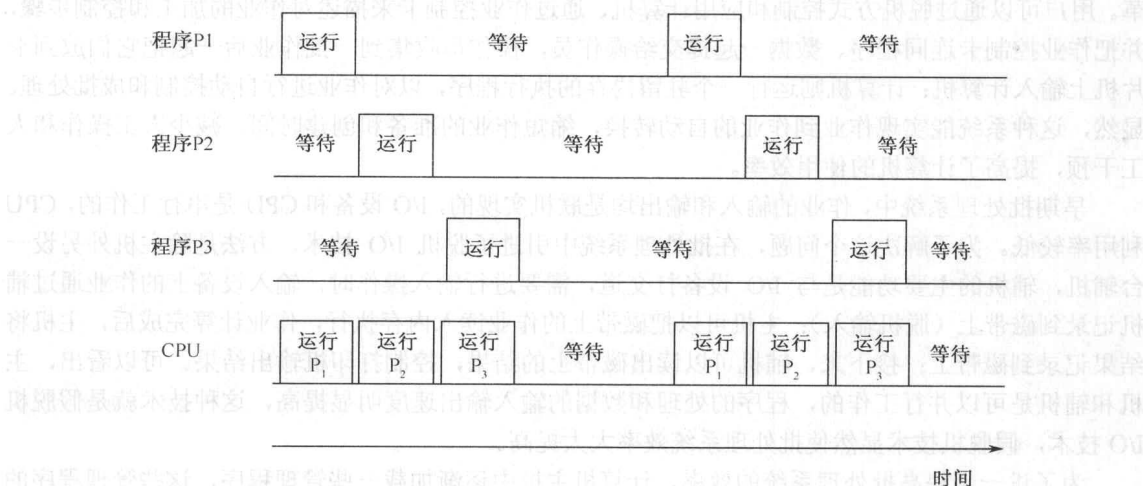


图 1-3 多道程序设计系统

多道程序系统具备多道、宏观并行和微观串行等特点,另外还有一些显著特征。

(1) 无序性:多个作业完成的先后顺序与它们进入主存的顺序之间并无严格的对应关系。如先进入主存的程序不一定能保证首先完成,甚至可能最后完成,而后进入主存的程序也有可能先完成。

(2) 调度性:一个作业从提交给系统开始直至完成,可能需要经过多次作业调度和进程调度。具体的调度算法将在后续的章节中介绍。

4. 操作系统的发展

通过多道程序系统,可以使批处理更加有效,提高了系统资源利用率和吞吐量。但是,对许多作业来说,需要提供一种新模式以便用户可以直接与计算机交互,分时系统的出现解决了人机交互的问题。分时系统与多道批处理系统之间有着截然不同的性能差别,它既实现了人机交互的功能,还实现了多个用户同时共享使用一台主机的功能,而且非常适合执行数据查询功能。

分时系统的实现思想:每个用户在各自终端上以问答方式控制程序的运行,系统把 CPU 的时间划分成时间片段(也称时间片),轮流分配给各个联机终端用户,每个用户只能在极短的时间内执行,如果时间片用完,则挂起当前任务等待下次分配的时间片。人机交互的任务通常是发出简短命令的小任务,所用的时间片不会太大,因而每个终端用户的每次请求基本上都能获得系统较为快速的响应,感觉上是独占了这台计算机。可以看出,分时系统是多道程序系统的一个变种,CPU 被若干个交互式的用户通过联机终端多路复用了。分时操作系统如图 1-4 所示。

分时系统具有同时性、独立性、及时性和交互性等特征,得到了极为广泛的应用。

虽然多道程序系统和分时操作系统获得了较高的资源利用率和快速的响应时间,但它们难以满足实时控制和实时信息处理领域的需要。于是出现了实时操作系统,目前有三种典型的实时系统,过程控制系统、信息查询系统和事务处理系统。过程控制系统主要应用在现场实时数据采集、计算处理,进而控制相关执行机构的场合,如卫星测控系统、火炮自动控制系统等。信息查询系

统应用在必须做出极快回答和响应的实时信息处理场合，如情报检索系统。事务处理系统不仅要
对终端用户及时做出响应，而且要对系统中的文件和数据进行频繁的更新，如银行业务处理系统、
电子商务系统等。

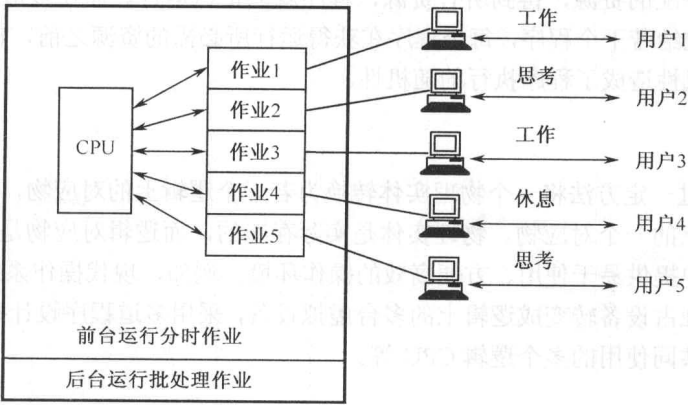


图 1-4 分时操作系统

实时系统是指能及时响应外部事件的请求，并在规定的较短时间内完成对该事件的处理，并
控制所有实时任务协调一致地运行的操作系统。

实时系统具有多路性、独立性、及时性、交互性和可靠性等特征，与分时系统相比，及时性
的特点更为明显。

1.1.3 操作系统的主要特性

前面介绍的几种操作系统都各自具有自己的特征，如批处理系统可以实现多个作业的成批处
理，简化用户使用难度，分时系统具有允许人机交互处理的特性，实时系统具有实时特性。但现
代操作系统应该具备并发性、共享性、异步性和虚拟性四种最基本特性，其中并发性是最重要的
特性。

1. 并发性

并发性是指两个或两个以上的事件在同一时间段内发生。操作系统是一个并发系统，操作系
统的并发性体现在计算机系统中同时存在若干个运行着的程序，这些程序交替执行。并发性有效
改善了包括 CPU 和 I/O 设备等系统资源的利用率，但也会产生一系列的系统管理问题，比如程序
和程序之间如何切换，程序切换时如何保证程序和数据互不干扰，这都要求系统提供控制和管理
程序的并发执行的机制和策略。

2. 共享性

共享性是指计算机系统的资源可被多个并发执行的程序共同使用。通常计算机系统的资源都
会按照进入系统的程序的个性要求进行分配，为了提高资源利用率，这种分配策略最好是动态的，
也就是程序需要使用某类资源时，它可以向系统提出申请，系统根据资源使用情况予以分配；而
当程序使用完毕后，应及时释放和归还资源，以便系统将资源分配给其他程序。

3. 异步性

异步性是指多道程序环境中, 程序的执行顺序和速度始终是动态变化和随机的。由于程序运行完全依赖于系统分配的资源, 得到所有资源, 程序就能正常运行。而在多道程序环境中, 系统有限的资源必须分配给若干个程序, 每个程序在获得运行所必需的资源之前, 它只能等待。因而, 系统资源分配的随机性造成了程序执行的随机性。

4. 虚拟性

虚拟性是指通过一定方法将一个物理实体转换为若干个逻辑上的对应物, 或者将物理上的多个实体转换为逻辑上的一个对应物。物理实体是实际存在的, 而逻辑对应物是虚拟的。采用虚拟技术的目的是为用户提供易于使用、方便高效的操作环境。例如, 现代操作系统采用 Spooling 技术把打印机这样的独占设备转变成逻辑上的多台虚拟设备, 采用多道程序设计技术将一个 CPU 虚拟成可供多个程序共同使用的多个逻辑 CPU 等。

1.2 操作系统的功能

操作系统的主要任务是为多道程序的运行提供良好的运行环境, 以保证程序能有条不紊地高效运行, 并能最大限度地提高系统中各种资源的利用率和方便用户的使用。为了实现上述目标, 操作系统应具有处理器管理、存储管理、设备管理和文件管理等方面的功能。另外, 为了方便用户使用操作系统, 还须向用户提供良好的用户界面和用户接口。

1.2.1 处理器管理

处理器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着关键的影响。处理器也是计算机系统最重要的资源, 其运算速度往往要比其他硬件设备的工作速度快得多, 其他设备的运行也常常需要处理器的介入。用户程序进入内存后, 只有获得处理器, 才能真正得以运行。因此, 对处理器的有效管理, 提高处理器的利用率是操作系统最重要的管理任务。

处理器管理的主要工作内容如下。

- (1) 统计系统中每个作业程序的状态, 以便将处理器分配给相关候选程序。
- (2) 指定处理器调度策略, 也是挑选待分配候选程序的必须遵循的原则。
- (3) 实施处理器的分配, 以便让获得处理器的程序真正投入运行。

为了顺利完成工作任务, 操作系统通过分别对作业、进程和线程进行相应的低级、中级和高级调度, 实现处理机的管理和调度。

1.2.2 存储管理

存储器是计算机系统中除了处理器以外的另一种重要资源, 主存储器 (也称为内存) 是处理器和外部设备共享和快速访问程序和数据的部件, 程序只有加载到主存储器后, 才有可能获得执行。存储管理的主要任务是为多道程序提供良好的环境, 方便用户使用存储器, 并提高存储器的利用率以及从逻辑上扩充内存。现代操作系统中, 存储器管理应具有内存分配和回收、内存保护、地址映射、主存共享和内存扩充等功能。

1. 内存分配和回收

内存分配和回收的主要任务是为每道程序分配内存空间，回收程序运行结束后释放的空间，提高存储器的利用率。内存分配的策略有静态和动态两种：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，在整个运行过程中不再接受新的请求，也不允许作业在内存中重新定位；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空间，也允许作业在内存中重新定位。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程序都只在自己的内存空间中运行，彼此互不干扰，既不允许用户程序访问系统程序和数据，也不允许用户程序转移到非共享的其他程序中运行。操作系统通过设置界限寄存器和越界检查机制保证执行程序的上界和下界。

3. 地址映射

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，这就导致地址空间内的逻辑地址和内存空间中的物理地址不一致。地址映射的任务是把用逻辑地址编程的应用程序装入主存，并将逻辑地址转换成主存物理地址，此功能应在硬件的支持下完成。

4. 内存共享

内存共享让主存中的多个应用程序实现存储共享，提高存储资源的利用率。多个应用程序共同访问同一段代码或者数据，可以通过内存共享技术将相关内存地址空间加载到应用程序的地址空间，而不必重复加载。

5. 内存扩充

内存扩充的任务不是扩大物理内存的容量，而是借助虚拟存储技术从逻辑上去扩充内存容量，使用户感觉到内存容量比实际物理内存大得多。虚拟存储技术的基本思想来自程序运行的局部性特点，在辅助存储器的配合下，采取部分装入，用时调入，不用时置换到辅助存储器的机制。

1.2.3 设备管理

设备管理用于管理计算机系统的所有的外部设备，而设备管理的主要任务是：完成用户进程提出的 I/O 请求；为用户进程分配其所需的 I/O 设备；提高 CPU 和 I/O 设备的利用率；提高 I/O 速度；方便用户使用 I/O 设备。设备管理的主要功能有缓冲区管理、设备分配、设备驱动和虚拟设备等功能。

1. 缓冲区管理

设置缓冲区的目的是匹配 CPU 的高速特性和 I/O 设备的相对低速特性，最常见的缓冲区机制有单缓冲、双缓冲和公用缓冲池等。

2. 设备分配

设备分配通常采用独享、共享和虚拟分配三种技术，以满足不同用户程序对外部设备不同的输入/输出要求。