



全国高等教育自学考试指定教材 计算机及应用专业(独立本科段)

操作系统

附：操作系统自学考试大纲

课程代码
2326
[2007年版]

组编 / 全国高等教育自学考试指导委员会
主编 / 谭耀铭

本教材附赠网络学习卡

中国人民大学出版社

全国高等教育自学考试指定教材
计算机及应用专业（独立本科段）

操作系统

（2007年版）

（附：操作系统自学考试大纲）

全国高等教育自学考试指导委员会 组编
谭耀铭 主编

中国人民大学出版社

图书在版编目 (CIP) 数据

操作系统 (2007 年版) (附: 操作系统自学考试大纲)
谭耀铭主编; 全国高等教育自学考试指导委员会组编. 2 版
北京: 中国人民大学出版社, 2007
全国高等教育自学考试指定教材. 计算机及应用专业. 独立本科段
ISBN 978-7-300-03235-1

- I. 操…
- II. ①谭…②全…
- III. 操作系统-高等教育-自学考试-教材
- IV. TP316

中国版本图书馆 CIP 数据核字 (2007) 第 058178 号

全国高等教育自学考试指定教材
计算机及应用专业(独立本科段)
操作系统(2007 年版)
(附: 操作系统自学考试大纲)
全国高等教育自学考试指导委员会 组编
谭耀铭 主编

出 版	中国人民大学出版社		
社 址	北京中关村大街 31 号	邮政编码	100080
网 址	http://www.crup.com.cn		
	http://www.ttrnet.com (人大教研网)		
印 刷	北京友谊印刷有限公司		
规 格	184mm × 260mm 16 开本	版 次	1999 年 10 月第 1 版 2007 年 7 月第 2 版
印 张	18.75	印 次	2007 年 7 月第 1 次印刷
字 数	558 000	定 价	28.00 元

版权所有 侵权必究 印装差错 负责调换

组编前言

21世纪是一个变幻难测的世纪，是一个催人奋进的时代。科学技术飞速发展，知识更替日新月异。希望、困惑、机遇、挑战，随时随地都有可能出现在每一个社会成员的生活之中。抓住机遇，寻求发展，迎接挑战，适应变化的制胜法宝就是学习——依靠自己学习，终生学习。

作为我国高等教育组成部分的自学考试，其职责就是在高等教育这个水平上倡导自学、鼓励自学、帮助自学、推动自学，为每一个自学者铺就成才之路。组织编写供读者学习的教材就是履行这个职责的重要环节。毫无疑问，这种教材应当适合自学，应当有利于学习者掌握、了解新知识、新信息，有利于学习者增强创新意识、培养实践能力、形成自学能力，也有利于学习者学以致用，解决实际工作中所遇到的问题。具有如此特点的书，我们虽然沿用了“教材”这个概念，但它与那种仅供教师讲、学生听，教师不讲、学生不懂，以“教”为中心的教科书相比，已经在内容安排、形式体例、行文风格等方面都大不相同了。希望读者对此有所了解，以便从一开始就树立起依靠自己学习的坚定信念，不断探索适合自己的学习方法，充分利用已有的知识基础和实际工作经验，最大限度地发挥自己的潜能，达到学习的目标。

欢迎读者提出意见和建议。

祝每一位读者自学成功。

全国高等教育自学考试指导委员会

2007年5月

编者的话

2005年11月全国高等教育自学考试指导委员会电子电工与信息类专业委员会在上海召开了一次教材研讨会。与会代表聆听了由全国自学考试办公室教材处和命题处的负责人所做的关于修改自学考试教材的必要性和修改原则的报告。代表们进行了分组讨论，并修订了各有关课程的自学考试大纲。

《操作系统》是计算机及应用专业（独立本科段）的一本自学考试教材。该教材的第1版于1999年10月出版，至今已有七年多时间。在这七年的使用中，该教材得到了广大参加自学考试的考生及其他读者的肯定和支持。但七年来计算机科学技术及其应用已有很大发展，因而有必要对原教材进行修改和更新。

《操作系统》（2007版）是根据全国高等教育自学考试指导委员会审定后的新大纲重新编写的。编写时按照新大纲的要求，参考了历届自考生和社会助学单位的反馈意见，对原教材的内容作了删改、补充和调整。

全书仍从操作系统实现资源管理的观点出发，阐述操作系统的基本工作原理和实现技术。全书共分9章。第1章介绍操作系统概况。第2章是计算机系统结构简介，阐述了操作系统在计算机系统中的地位、作用，以及与硬件、用户之间的关系。第3章至第6章是从计算机系统的资源管理观点出发，详细阐述了对处理器、主存储器、文件、设备进行管理的基本原理和实现技术。第7章和第8章介绍了进程并发执行时必须解决的三大问题，即进程同步、进程通信和死锁。第9章是有关操作系统的实验指导。

本书还以一个精巧实用的操作系统UNIX为实例，穿插在各章中来阐述操作系统基本原理的实际应用。读者可从UNIX的实现技术中得到启发，既能从中学到对操作系统基本原理的灵活应用，又能加深对基本原理的理解。此外，每章之后都有小结和习题（第9章实验除外），以便于读者掌握要点和进行复习。

本教材在编写过程中力求概念清楚，表述正确，通俗易懂，便于自学。但由于编者水平有限及计算机技术的迅猛发展，书中难免出现不妥之处和错误，恳请读者批评指正，不胜感谢。

本教材经北京大学陈向群教授（主审）、东南大学邢汉承教授、上海交通大学徐良贤教授认真仔细审阅。2006年12月，全国高等教育自学考试指导委员会电子电工与信息类专业委员会在南京大学召开了审稿会。审稿会上三位审阅教授对教材提出了中肯的修改意见。在此向他们表示万分感谢。另外，在本教材的编写过程中还得到了徐永森、徐文、奚红的帮助，在此对他们一并表示由衷的感谢。

本教材既可供计算机及应用专业自学考试的考生使用，也可供普通高等学校计算机专业的学生及各类计算机专业的函大、电大学生使用或作为教学参考书。

编者

2007年5月于南京

目 录

操作系统 (2007 年版)

第 1 章 引论	3
1.1 什么是操作系统	3
1.1.1 计算机系统	3
1.1.2 操作系统	4
1.2 操作系统的形成	4
1.3 操作系统的基本类型	6
1.3.1 批处理操作系统	6
1.3.2 分时操作系统	8
1.3.3 实时操作系统	9
1.4 操作系统的发展	10
1.5 UNIX 操作系统简介	11
1.6 操作系统的功能	12
小结	13
习题	13
第 2 章 计算机系统结构简介	15
2.1 计算机系统结构	15
2.1.1 层次结构	15
2.1.2 系统工作框架	16
2.2 硬件环境	17
2.2.1 CPU 与外设的并行工作	17
2.2.2 存储体系	18
2.2.3 保护措施	19
2.3 操作系统结构	21
2.3.1 设计目标	21
2.3.2 操作系统的层次结构	22
2.3.3 UNIX 系统的结构	23
2.4 操作系统与用户的接口	24
2.4.1 操作控制命令	24
2.4.2 系统调用	25
2.5 UNIX 的用户接口	26

2.5.1	shell 命令	26
2.5.2	UNIX 系统调用	30
小结	32
习题	33
第3章	处理器管理	35
3.1	多道程序设计	35
3.1.1	什么是多道程序设计	35
3.1.2	为什么要采用多道程序设计	35
3.1.3	采用多道程序设计应注意的问题	38
3.2	进程概述	39
3.2.1	进程的定义	39
3.2.2	为什么要引入进程	40
3.2.3	进程的属性	41
3.3	进程队列	42
3.3.1	进程控制块	42
3.3.2	进程的创建和撤销	43
3.3.3	进程队列的链接	44
3.4	UNIX 系统中的进程	46
3.4.1	UNIX 进程的特点	46
3.4.2	UNIX 进程的组成	46
3.4.3	UNIX 进程的状态	51
3.4.4	UNIX 进程的创建和终止	53
3.4.5	UNIX 进程的换进换出	56
3.4.6	UNIX 进程的睡眠与唤醒	57
3.5	中断技术	58
3.5.1	中断和中断类型	58
3.5.2	中断响应	59
3.5.3	中断事件的处理	61
3.5.4	中断优先级和中断屏蔽	62
3.6	UNIX 系统的中断技术	63
3.6.1	中断事件和异常情况	63
3.6.2	处理器状态字	64
3.6.3	中断处理	64
3.7	处理器调度	66
3.7.1	处理器的两级调度	66
3.7.2	批处理作业的调度算法	67
3.7.3	进程调度算法	72
3.7.4	UNIX 系统的进程调度算法	75
小结	77

习题	79
第4章 存储管理	82
4.1 概述	82
4.1.1 信息的二级存储	82
4.1.2 存储管理的功能	82
4.2 重定位	84
4.2.1 绝对地址和相对地址	84
4.2.2 重定位的方式	84
4.3 单用户连续存储管理	86
4.3.1 存储空间的分配	86
4.3.2 覆盖技术	87
4.3.3 对换技术	88
4.4 固定分区存储管理	88
4.4.1 基本原理	88
4.4.2 主存空间的分配与回收	88
4.4.3 地址转换和存储保护	89
4.5 可变分区存储管理	90
4.5.1 主存空间的分配与回收	91
4.5.2 地址转换和存储保护	96
4.5.3 移动技术	96
4.6 页式存储管理	98
4.6.1 页式存储管理的基本原理	98
4.6.2 页式主存空间的分配与回收	99
4.6.3 页表和地址转换	100
4.7 虚拟存储管理	103
4.7.1 什么是虚拟存储器	103
4.7.2 虚拟存储器的工作原理	103
4.7.3 页式虚拟存储器的实现	103
4.7.4 多级页表	108
4.8 UNIX系统的页式虚拟存储管理	110
4.8.1 UNIX的虚拟地址结构	110
4.8.2 UNIX的页表和地址转换	110
4.8.3 UNIX的页面调度	112
小结	112
习题	114
第5章 文件管理	116
5.1 文件和文件系统	116
5.1.1 文件	116
5.1.2 文件系统的组成部分	117

5.2	文件的存储介质	117
5.3	文件的存取方式	119
5.4	文件目录	119
5.4.1	一级目录结构	120
5.4.2	二级目录结构	120
5.4.3	树形目录结构	121
5.4.4	文件目录的管理	123
5.5	文件的组织	123
5.5.1	文件的逻辑结构	123
5.5.2	文件的存储结构	124
5.5.3	记录的成组与分解	131
5.6	磁盘存储空间的管理	133
5.6.1	位示图法	134
5.6.2	空闲块表法	135
5.6.3	空闲块链法	135
5.7	基本文件操作及其使用	136
5.7.1	基本文件操作	136
5.7.2	文件操作的使用	138
5.8	文件的安全性	139
5.8.1	文件的保护	139
5.8.2	文件的保密	141
5.9	UNIX 系统的文件管理	142
5.9.1	UNIX 的文件和文件系统	142
5.9.2	UNIX 的文件结构	143
5.9.3	UNIX 的文件目录	144
5.9.4	UNIX 的索引节点	145
5.9.5	UNIX 的打开文件表	148
5.9.6	UNIX 的文件操作	150
5.9.7	UNIX 的文件存储空间管理	153
	小结	155
	习题	156
第 6 章	设备管理	158
6.1	设备管理的功能	158
6.2	外围设备的分类	159
6.3	独占设备的管理	160
6.3.1	设备的绝对号与相对号	160
6.3.2	独占设备的分配	160
6.4	磁盘的驱动调度	163
6.4.1	访问磁盘的操作时间	163

6.4.2	移臂调度	164
6.4.3	旋转调度	166
6.4.4	信息的优化分布	168
6.5	输入/输出操作的实现	169
6.5.1	通道结构和通道程序	169
6.5.2	外围设备的启动	172
6.5.3	I/O 中断事件的处理	174
6.6	缓冲技术	175
6.6.1	单缓冲技术	175
6.6.2	双缓冲技术	176
6.6.3	缓冲池技术	176
6.7	虚拟设备	177
6.7.1	为什么要提供虚拟设备	177
6.7.2	虚拟设备的实现	178
6.8	UNIX 的设备管理	181
6.8.1	UNIX 的设备和设备文件	181
6.8.2	UNIX 的块设备缓冲技术	182
	小结	185
	习题	186
第 7 章	进程同步与进程通信	189
7.1	进程的顺序性与并发性	189
7.1.1	进程的顺序性	189
7.1.2	进程的并发性	189
7.2	与时间有关的错误	190
7.3	临界区与 PV 操作	192
7.3.1	临界区	192
7.3.2	PV 操作	193
7.3.3	用 PV 操作管理临界区	194
7.4	进程的互斥	195
7.4.1	用 PV 操作实现进程的互斥	196
7.4.2	读者/写者问题	198
7.5	进程的同步	201
7.5.1	协作	201
7.5.2	用 PV 操作实现进程的同步	202
7.6	进程通信	214
7.6.1	通信机制	214
7.6.2	间接通信	215
7.6.3	用进程通信实现进程同步	218
7.7	UNIX 中的进程同步与进程通信	218

7.7.1	UNIX 中的进程同步	218
7.7.2	UNIX 中的进程通信	219
7.8	线程概念	221
7.8.1	什么是线程	221
7.8.2	进程与线程	222
小结	223
习题	225
第 8 章	死锁	228
8.1	死锁的形成	228
8.2	死锁的特征	232
8.2.1	死锁的必要条件	232
8.2.2	资源分配图	232
8.3	死锁的防止	234
8.3.1	互斥条件	235
8.3.2	占有并等待条件	235
8.3.3	不可抢夺条件	236
8.3.4	循环等待条件	236
8.4	死锁的避免	238
8.4.1	安全状态	238
8.4.2	银行家算法	239
8.5	死锁的检测	242
8.5.1	死锁的检测方法	242
8.5.2	死锁的解除	245
小结	246
习题	247
第 9 章	实验	249
9.1	模拟设计进程调度	249
9.1.1	实验要求	249
9.1.2	模拟设计	249
9.1.3	流程图	250
9.2	模拟设计页面调度	251
9.2.1	实验要求	251
9.2.2	模拟设计	251
9.2.3	流程图	253
9.3	模拟设计文件操作	253
9.3.1	实验要求	253
9.3.2	模拟设计	254
9.3.3	流程图	254

参考文献..... 262

附：操作系统自学考试大纲

出版前言..... 265
I. 课程性质与设置目的..... 266
II. 课程内容与考核目标..... 267
III. 有关说明与实施要求..... 280
附录 题型举例..... 283

后记..... 286

操作系统

(2007 年版)

第1章 引 论

1.1 什么是操作系统

众所周知，操作系统是现代计算机系统中一种不可缺少的系统软件。它在计算机用户与计算机硬件之间起着中介作用。但是，操作系统到底是什么？操作系统要做哪些工作？是每一个初学者必然提出的问题。下面将根据操作系统在计算机系统中的地位和作用阐述操作系统的概念。

1.1.1 计算机系统

计算机系统是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信息的系统。计算机系统由硬件（子）系统和软件（子）系统组成。硬件系统是计算机系统赖以工作的实体。软件系统保证计算机系统按用户指定的要求协调地工作。

硬件系统主要由中央处理器（CPU）、主存储器、辅助存储器（磁盘、磁带、光盘等）以及各种输入/输出设备（键盘、显示器、打印机等）组成。软件系统是指计算机系统内的程序及其文档。这些硬件和软件的组成部分都被看做计算机系统的资源。因此，计算机系统的资源包括两大类：硬件资源和软件资源。图1—1所示是一个计算机系统的抽象视图。

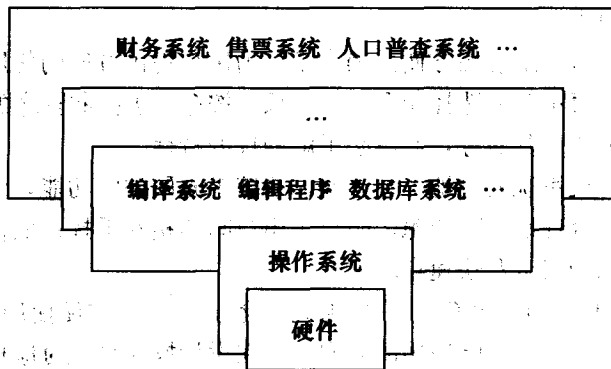


图 1—1 计算机系统的抽象视图

从操作系统的角度来观察，各种不同层次的用户（人、应用程序、其他计算机系统）要利用计算机系统解决的问题可能不相同，但是，任何一个程序在执行前必须获得主存空间后才能被装入主存，程序的执行要依靠处理器，程序执行中还要使用各种设备完成信息的输入和输出，或者调用共享文件和公共子程序等。也就是说，任何程序的执行都要占用计算机系统的资源。然而，各用户对资源的请求和使用可能会有冲突。例如，当一个用户正在用打印

机输出信息时，另一个用户也要求使用该打印机，如果对用户的这种资源请求不加限制，则将会引起混乱。因此，计算机系统必须具有控制和协调资源分配的能力。各用户还希望计算机系统能按各自的要求控制程序的执行。于是，计算机系统必须具有识别用户要求的能力，也必须具有对不同用户进行不同控制执行的能力。

1.1.2 操作系统

操作系统是计算机系统中一个不可缺少的组成部分。不同的人从不同的角度看到的操作系统不尽相同。例如，操作使用者认为操作系统是一组命令的集合，它接收输入的命令，并按要求完成指定的功能；程序设计人员认为操作系统是一组功能调用程序的集合，它为程序员编制程序提供了方便。因而，至今对操作系统尚未有一个严格的定义。

但从总体出发，一般认为，操作系统（Operating System，缩写为 OS）是一种管理计算机系统资源、控制程序执行、改善人机界面和为其他软件提供支持的系统软件。

操作系统是一种资源管理程序。为什么要对计算机系统的资源进行管理呢？首先，计算机系统有限的资源与众多的资源请求者之间存在矛盾。因此，必须采取一些办法为用户分配需要的资源，如 CPU 时间、存储空间、输入/输出设备等，使相互竞争者能合理地共享资源，以保证计算机系统的效率和公平。其次，计算机系统的各种资源其性质均不相同。因此，必须研究它们的“个性”，总结它们的“共性”，寻求合适的使用方法和策略，以提高计算机系统的可靠性和安全性。

操作系统也是一种控制其他程序执行的控制程序。操作系统识别用户的命令和要求，按照用户的意图控制程序的执行。同时，操作系统也必须既要为用户使用计算机系统提供方便，又要核对用户要求的合法性，以防止用户错误和不合理地使用计算机系统。

因此，操作系统作为一种系统程序，其目的是提供一个供其他程序执行的良好环境。故它有两个主要的设计原则：

第一，能使得计算机系统使用方便。操作系统为用户提供方便的使用接口，用户按需要输入命令或从提供的“菜单”中选择命令，操作系统按命令去控制程序的执行；用户也可以调用操作系统的功能模块为其服务，而不必了解硬件的特性。

第二，能使得计算机系统高效地工作。操作系统扩充硬件的功能，使硬件的功能发挥得更好；操作系统使用户合理地共享资源，防止各用户间的相互干扰；操作系统以文件形式管理软件资源，保证信息的安全和快速存取。

这两个原则，“方便”和“效率”，有时会发生矛盾。为求方便使用，可能要牺牲效率。同样，为了保证效率，可能会影响使用的方便性。在设计操作系统时应根据计算机系统的功能和服务对象，权衡方便性和高效性，作出决策。

1.2 操作系统的形成

早期的计算机体积巨大，速度低，设备少，编制的程序也比较简单。那时，计算机上不配置操作系统，程序员往往直接使用机器指令编写程序。程序的装入、调试以及控制程序的

运行都是通过控制台上的开关来实现的。在这期间，整个计算机都被一个程序员所占有。因而，不需要专门的操作员，程序员身兼两职，既是操作员，也是程序员。当程序运行时通过控制台上的开关和显示灯来管理他的程序。如果发现错误，程序员暂停程序的运行，测试主存储器寄存器的内容，然后直接从控制台上排除程序错误。

随着计算机的发展，硬件增加了读卡机、行式打印机和磁带机。为了使程序设计变得容易，产生了协助用户使用计算机的软件——原始汇编系统。该系统提供了“汇编语言”和“汇编语言解释程序”。汇编语言是最早的程序设计语言，它的每个语句都与一条机器指令相对应，但用符号记忆码替换了指令中的二进制数字码。程序按符号形式的汇编语言书写要比用机器指令书写来得容易。用汇编语言编写的程序称为源程序，它不能直接在机器上执行，只有通过汇编语言解释程序把源程序转换成用机器指令序列表示的目标程序后才能在计算机上运行。因此，在这样的计算机系统中，首先要把汇编语言解释程序和源程序都记录在特制的卡片上，然后再装入和执行。这种特制的卡片以穿孔的形式记录信息，每张卡片可以记录80个字符，有孔的位置对应二进制“1”，无孔的位置对应二进制“0”。一个程序或一组数据往往要用一叠卡片。当把程序和数据记录到卡片上后，就可按如下步骤进行操作：

- (1) 启动读卡机，把汇编语言解释程序装入计算机；
- (2) 执行汇编语言解释程序，汇编语言解释程序从读卡机上读入源程序，并将其转换成目标程序，把产生的目标程序记录到卡片上或磁带上；
- (3) 启动读卡机或磁带机把目标程序装入计算机；
- (4) 运行目标程序，目标程序从读卡机上读入需处理的数据，经处理后把产生的结果从打印机上打印出来。

这种工作方式比早期的工作方式前进了一步，但本质上还是手工操作方式。用户需要不断地更换读卡机上的卡片，这就增加了出错机会。

为了启动读卡机、磁带机、打印机等设备进行工作，必须配置控制设备工作的程序。由于每种设备都有自己的特点，所以，对每一种设备都需要编写专门的例行子程序，这种例行子程序被称为**设备驱动程序**，供用户需要时调用。这些设备驱动程序可以看成是最原始的操作系统。

20世纪50年代末至60年代初，计算机进入第二代。不仅计算速度有了很大提高，而且存储容量大幅度增长。这给软件的发展奠定了基础，先后出现了FORTRAN、ALGOL和COBOL等程序设计和相应的编译程序。程序员用这些程序设计语言能很容易地编写程序。但是，如果对计算机的操作仍停留在手工操作方式上，那么仍要不断地装卸卡片或磁带，不仅花费时间，而且操作复杂，容易错乱。于是迫切需要一种能对计算机硬件和软件进行管理和调度的程序。当时在美国的IBM360系列计算机系统和英国的1900系列计算机系统上都配置了这种程序，称为**管理程序**。

有了管理程序后，程序员不必亲自上机操作，而可由专业化的操作员代劳。操作员只要从控制台输入命令，然后就由管理程序来识别和执行。这样，不仅操作速度快，而且当计算机在执行过程中发生错误或意外时，管理程序将输出信息向操作员报告。管理程序不仅协助操作员操纵计算机，而且还管理计算机的部分资源。例如，当设备发生故障时，管理程序将负责处理而不必用户费心。管理程序还对文件进行管理，用户可以按文件名而不是物理地址存取信息，这不仅方便灵活，而且安全可靠。这种管理程序可以被看做初级的操作系统。