



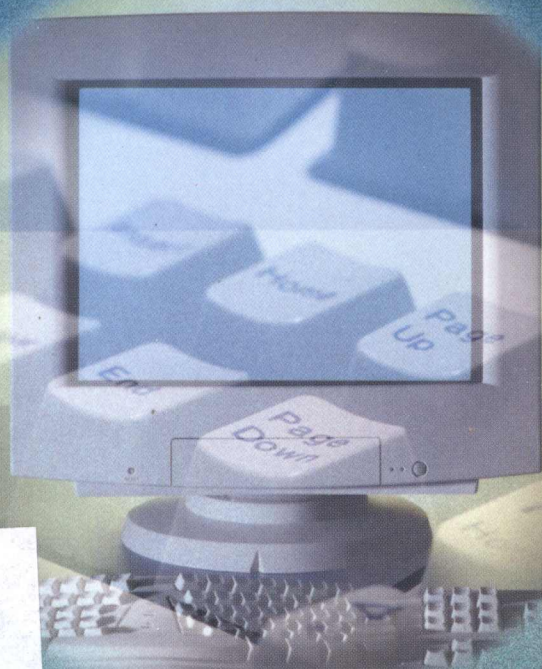
专升本

教育部师范教育司组织编写  
中学教师进修高等师范本科(专科起点)教材

# 计算机操作系统原理

主编 王万森

编者 沈雪明 汪国安 杨 卉 涂相华



高等教育出版社

954

教育部师范教育司组织编写  
中学教师进修高等师范本科(专科起点)教材

# 计算机操作系统原理

主编 王万森

编者 沈雪明 汪国安 杨卉 涂相华

高等教育出版社

## 图书在版编目 (CIP) 数据

计算机操作系统原理/王万森主编. —北京: 高等教育出版社, 2001.7

ISBN 7-04-009361-8

I. 计… II. 王… III. 操作系统(软件)—专升本—教材 IV. TP316

中国版本图书馆 CIP 数据核字 (2001) 第 029025 号

计算机操作系统原理  
王万森 主编

---

出版发行 高等教育出版社

社 址 北京市东城区沙滩后街 55 号

邮政编码 100009

电 话 010—64054588

传 真 010—64014048

网 址 <http://www.hep.edu.cn>

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

经 销 新华书店北京发行所

印 刷 河北新华印刷一厂

开 本 787×960 1/16

版 次 2001 年 7 月第 1 版

印 张 19

印 次 2001 年 7 月第 1 次印刷

字 数 350 000

定 价 16.30 元

---

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

**版权所有 侵权必究**

## 前 言

操作系统是现代计算机系统中必不可少的重要系统软件。随着我国信息技术和信息产业的发展,人们越来越认识到操作系统在我国国民经济和社会信息化进程中的基础性地位。确切地说,如果没有我们国家自己的操作系统,就不会有我们国家真正的信息安全。

操作系统课程是计算机专业的一门专业主干课程,主要研究操作系统的基本原理和实现方法,是计算机专业学生的知识结构中必不可少的一个重要组成部分。

本书是编者在多年操作系统教学的基础上,根据教育部师范教育司 1999 年制定的《中学教师进修高等师范本科(专科起点)教学计划》编写的。全书可分为两大部分:第一部分从第一章到第八章,全面系统地讨论了操作系统的基本原理和实现技术;第二部分为第九章和第十章,分别讨论了 Linux 操作系统和操作系统的最新进展。全书各章的安排分别为:第一章操作系统概述,其主要目的是让读者先对操作系统有一个概括的了解,并初步建立起操作系统的整体概念;第二章用户接口,介绍了操作系统向用户提供服务的各种手段;第三章进程管理,介绍了进程的基本概念、实现方法、控制技术,以及进程互斥、同步、通信的概念与方法;第四章死锁,介绍了死锁的概念和各种死锁对策;第五章处理机管理,介绍了各种调度模型和一些重要的调度算法;第六章存储管理,介绍了实存和虚存管理的有关概念和主要的存储管理方法;第七章设备管理,介绍了 I/O 设备的管理技术和各种磁盘调度算法;第八章文件系统,介绍了文件系统的有关概念,文件的组织、存取、共享、保护方法,以及文件目录、文件存储空间的管理方法;第九章 Linux 操作系统, Linux 是一个具有全部 UNIX 功能的全开放的多用户、多任务操作系统,其源代码可以直接从网上下载,再加上它的微内核结构,作为实例讨论比较合适;第十章操作系统的进一步发展,主要从新一代操作系统和操作系统新技术两个方面来探讨。

书中每一章的开头部分都给出了该章的学习目标和重点、难点说明,对读者学习和理解该内容起到一定的指导作用;每章附有习题,且对所有习题均给出相应的参考答案,以便于读者自学。

本书由王万森主编,负责全书的大纲制定和统编。参加编写工作的有:王万森(第一、十章),沈雪明(第三、五章),汪国安(第六、七章),杨卉(第八、九章),涂

相华(第二、四章)。

在编写本书之前,编者均曾主讲过不同版本的操作系统教材,这些教材不论对以往的教学,还是对本书的编写,都有很大的帮助。在此,谨向这些教材的作者深表谢意。

彭玉禄审阅了全书,并提出了许多宝贵意见,在此,谨表示衷心感谢。

在本书的编写、出版过程中,得到了高等教育出版社的大力支持,谨向为这本书的出版而辛勤工作的同志们表示诚挚的谢意。

由于作者水平所限,书中疏漏与错误之处在所难免,恳请各位专家和读者批评指正。

编者

2001年3月

# 目 录

<b>第一章 操作系统概述</b> .....	(1)	1.7 操作系统的结构 .....	(23)
1.1 什么是操作系统 .....	(1)	1.7.1 整体式系统 .....	(23)
1.1.1 操作系统的虚拟机 观点 .....	(1)	1.7.2 层次式系统 .....	(23)
1.1.2 操作系统的资源 管理观点 .....	(3)	1.7.3 客户/服务器系统 .....	(24)
1.1.3 操作系统的用户 服务观点 .....	(3)	1.8 中断及其在操作系统 中的作用 .....	(24)
1.2 操作系统的形成过程 .....	(4)	1.8.1 中断是多道程序并发 执行的推动力 .....	(25)
1.2.1 人工操作阶段 .....	(4)	1.8.2 操作系统由中断 驱动 .....	(25)
1.2.2 单道批量处理阶段 .....	(4)	学后自测一 .....	(25)
1.2.3 执行系统阶段 .....	(6)	参考答案一 .....	(26)
1.2.4 多道程序系统阶段 .....	(7)	<b>第二章 用户接口</b> .....	(27)
1.3 操作系统的基本类型 .....	(8)	2.1 用户接口概述 .....	(27)
1.3.1 多道批处理系统 .....	(8)	2.1.1 运行用户程序 的过程 .....	(27)
1.3.2 分时系统 .....	(9)	2.1.2 作业管理概述 .....	(28)
1.3.3 实时系统 .....	(11)	2.2 命令接口 .....	(29)
1.3.4 通用操作系统 .....	(13)	2.2.1 脱机命令接口 .....	(29)
1.3.5 网络操作系统 .....	(13)	2.2.2 联机命令接口 .....	(30)
1.4 操作系统的特征 .....	(14)	2.3 程序接口 .....	(32)
1.5 操作系统的服务 .....	(16)	2.3.1 系统调用概述 .....	(33)
1.5.1 操作系统的公共 服务 .....	(16)	2.3.2 系统调用过程 .....	(34)
1.5.2 操作系统的服务 方式 .....	(17)	2.4 图形接口 .....	(35)
1.6 操作系统的功能 .....	(18)	学后自测二 .....	(36)
1.6.1 用户接口 .....	(18)	参考答案二 .....	(36)
1.6.2 处理机管理 .....	(19)	<b>第三章 进程管理</b> .....	(37)
1.6.3 存储管理 .....	(20)	3.1 进程的基本概念 .....	(37)
1.6.4 设备管理 .....	(21)	3.1.1 进程的引入 .....	(37)
1.6.5 文件管理 .....	(22)	3.1.2 进程的定义 和特征 .....	(42)

3.1.3 进程的基本状态 及其变迁 .....	(43)	3.8.3 实现管程的三个 关键问题 .....	(76)
3.2 进程的实现 .....	(45)	学后自测三 .....	(77)
3.2.1 进程的结构描述 .....	(45)	参考答案三 .....	(80)
3.2.2 进程控制块的 结构 .....	(46)	<b>第四章 死锁及其对策</b> .....	(89)
3.2.3 进程控制块的 组织形式 .....	(47)	4.1 死锁的基本概念 .....	(89)
3.3 进程控制 .....	(48)	4.1.1 资源 .....	(90)
3.3.1 进程控制机构 .....	(48)	4.1.2 死锁的定义 .....	(90)
3.3.2 进程控制原语 .....	(49)	4.1.3 产生死锁的原因 .....	(91)
3.4 进程的互斥与同步 .....	(53)	4.2 死锁原理及对策 .....	(92)
3.4.1 进程互斥 .....	(53)	4.2.1 死锁原理及产生 死锁的必要条件 .....	(92)
3.4.2 进程同步 .....	(55)	4.2.2 死锁的描述 .....	(93)
3.4.3 利用信号量机制解决 进程互斥、同步及前 趋图问题 .....	(56)	4.2.3 解决死锁的方法 .....	(96)
3.5 利用信号量机制解决经典 进程同步问题 .....	(61)	4.3 鸵鸟算法 .....	(96)
3.5.1 生产者—消费者 问题 .....	(61)	4.4 死锁的检测和恢复 .....	(97)
3.5.2 哲学家进餐问题 .....	(64)	4.4.1 利用资源分配图 描述系统状态 .....	(97)
3.5.3 读者—写者问题 .....	(66)	4.4.2 死锁检测中的数据 结构 .....	(100)
3.6 进程通信 .....	(67)	4.4.3 死锁检测算法 .....	(100)
3.6.1 进程通信的类型 .....	(67)	4.4.4 死锁的恢复 .....	(101)
3.6.2 消息传递系统 .....	(68)	4.5 死锁预防 .....	(102)
3.6.3 消息缓冲队列通信 机制——直接通信 方式的实现 .....	(70)	4.5.1 打破“不剥夺” 条件 .....	(102)
3.7 线程的概念 .....	(72)	4.5.2 打破“部分分配” 条件 .....	(103)
3.7.1 线程的定义和 属性 .....	(73)	4.5.3 打破“环路等待” 条件 .....	(103)
3.7.2 线程与进程的 比较 .....	(73)	4.6 死锁避免 .....	(104)
3.8 管程的概念 .....	(74)	4.6.1 系统状态的 安全性 .....	(104)
3.8.1 管程的引入 .....	(74)	4.6.2 银行家算法 .....	(105)
3.8.2 管程的定义 .....	(75)	4.6.3 银行家算法 举例 .....	(107)
		学后自测四 .....	(109)
		参考答案四 .....	(110)

<b>第五章 处理机管理</b> .....	(112)	存储管理 .....	(150)
5.1 调度的类型和模型 .....	(112)	6.2.4 分区的存储保护 .....	(152)
5.1.1 作业调度 .....	(112)	6.3 虚拟存储系统的基本	
5.1.2 进程调度 .....	(114)	概念 .....	(152)
5.1.3 中级调度 .....	(115)	6.3.1 何谓虚拟存储器 .....	(152)
5.1.4 调度队列模型 .....	(115)	6.3.2 实现虚拟存储器的	
5.2 调度算法的选择和		基本原理 .....	(153)
性能评价 .....	(118)	6.4 分页存储管理 .....	(154)
5.2.1 调度算法选择时		6.4.1 分页存储管理的	
考虑的因素 .....	(118)	实现原理 .....	(154)
5.2.2 调度算法性能的		6.4.2 简单分页存储	
评价 .....	(120)	管理 .....	(155)
5.3 调度算法 .....	(121)	6.4.3 请求分页存储	
5.3.1 先来先服务调度		管理 .....	(158)
算法 .....	(121)	6.4.4 请求分页存储管理	
5.3.2 短作业(进程)优先		的页面置换算法 .....	(160)
调度算法 .....	(123)	6.4.5 抖动和工作集	
5.3.3 优先级调度算法 .....	(125)	模型 .....	(164)
5.3.4 时间片轮转调度		6.4.6 页面保护 .....	(167)
算法 .....	(127)	6.4.7 页式存储管理的	
5.3.5 多级反馈队列调度		优缺点 .....	(167)
算法 .....	(128)	6.5 请求分段存储管理 .....	(168)
5.3.6 实时调度算法 .....	(131)	6.5.1 段式存储管理的	
学后自测五 .....	(133)	基本思想 .....	(168)
参考答案五 .....	(135)	6.5.2 段式存储管理的	
<b>第六章 存储管理</b> .....	(139)	实现原理 .....	(168)
6.1 存储管理的基本概念 .....	(139)	6.5.3 段的共享与保护 .....	(172)
6.1.1 存储器的层次 .....	(139)	6.5.4 分段与分页的区别	
6.1.2 地址重定位 .....	(140)	与联系 .....	(173)
6.1.3 存储管理的目的		6.5.5 段式存储管理的	
和功能 .....	(143)	优缺点 .....	(174)
6.1.4 存储分配方式 .....	(144)	6.6 段页式存储管理 .....	(174)
6.2 实存储管理技术 .....	(145)	6.6.1 段页式存储管理的	
6.2.1 单一连续区管理 .....	(145)	基本思想 .....	(174)
6.2.2 固定式和可变式		6.6.2 段页式存储管理的	
分区存储管理 .....	(145)	地址变换过程 .....	(175)
6.2.3 可重定位分区		6.6.3 段页式存储管理的	



优缺点 .....	(177)	8.1.3 文件系统及其	
学后自测六 .....	(177)	功能 .....	(208)
参考答案六 .....	(179)	8.1.4 文件操作 .....	(209)
<b>第七章 I/O 系统及设备管理</b> .....	(180)	8.2 文件的组织和存取方法 ..	(209)
7.1 设备管理概述 .....	(180)	8.2.1 文件的逻辑组织 ..	(209)
7.1.1 I/O 设备的类型 ..	(180)	8.2.2 文件的物理组织 ..	(210)
7.1.2 设备管理的任务		8.2.3 文件的存取方法 ..	(214)
和功能 .....	(181)	8.3 文件目录管理 .....	(216)
7.2 I/O 控制方式 .....	(182)	8.3.1 目录的概念 .....	(216)
7.2.1 程序直接控制		8.3.2 目录的结构 .....	(217)
方式 .....	(182)	8.3.3 文件路径名 .....	(219)
7.2.2 程序中断 I/O 控制		8.4 文件存储空间的管理 .....	(219)
方式 .....	(183)	8.4.1 空白文件目录 .....	(220)
7.2.3 DMA 控制方式 ..	(183)	8.4.2 位示图 .....	(220)
7.2.4 I/O 通道控制		8.4.3 空闲块链 .....	(221)
方式 .....	(184)	8.5 文件的共享 .....	(223)
7.3 外设的启动和 I/O 中断		8.5.1 绕道法 .....	(223)
处理 .....	(186)	8.5.2 链访法 .....	(223)
7.3.1 I/O 设备的启动 ..	(187)	8.5.3 基本文件目录和符	
7.3.2 I/O 中断事件的		号文件目录结构 ..	(224)
处理 .....	(188)	8.6 文件的保护机制 .....	(225)
7.4 虚拟设备技术 .....	(189)	8.6.1 文件存取控制	
7.5 I/O 设备的分配 .....	(191)	矩阵 .....	(226)
7.5.1 设备分配中的数据		8.6.2 文件存取控制表 ..	(226)
结构 .....	(191)	8.6.3 用户存取权限表 ..	(227)
7.5.2 设备分配的原则和		8.6.4 口令核对法 .....	(227)
机制 .....	(193)	8.6.5 密码技术 .....	(227)
7.5.3 设备分配的实施 ..	(196)	8.7 文件系统的安全性 .....	(228)
7.6 驱动调度 .....	(198)	8.8 文件的主要操作 .....	(228)
7.6.1 移臂调度 .....	(199)	8.8.1 文件创建和删除 ..	(228)
7.6.2 旋转调度 .....	(202)	8.8.2 文件打开和关闭 ..	(229)
学后自测七 .....	(203)	8.8.3 文件的读和写 .....	(230)
参考答案七 .....	(204)	8.8.4 文件的链接和	
<b>第八章 文件系统</b> .....	(206)	解除链接 .....	(231)
8.1 文件系统概述 .....	(206)	学后自测八 .....	(232)
8.1.1 文件和文件分类 ..	(207)	参考答案八 .....	(232)
8.1.2 文件命名 .....	(208)	<b>第九章 Linux 操作系统</b> .....	(234)

---

9.1 Linux 简介 .....	(234)	9.6.1 Linux 的 shell .....	(263)
9.1.1 Linux 特性 .....	(234)	9.6.2 X Windows .....	(267)
9.1.2 Linux 系统结构及 核心功能 .....	(236)	学后自测九 .....	(267)
9.2 Linux 的进程管理 .....	(236)	参考答案九 .....	(268)
9.2.1 Linux 进程概述 ..	(236)	<b>第十章 操作系统的进一步发展</b> ..	(269)
9.2.2 Linux 进程调度 ..	(239)	10.1 计算机系统发展对操作 系统的影响 .....	(269)
9.2.3 Linux 进程控制 ..	(240)	10.2 新一代操作系统介绍 .....	(271)
9.2.4 Linux 进程通信 ..	(242)	10.2.1 微机操作系统 ..	(271)
9.3 Linux 的存储管理 .....	(248)	10.2.2 网络操作系统 ..	(274)
9.3.1 Linux 的分段和 分页机制 .....	(249)	10.2.3 分布式操作 系统 .....	(278)
9.3.2 Linux 内存管理 机制 .....	(252)	10.3 操作系统新技术 .....	(281)
9.4 Linux 的文件系统管理 .....	(255)	10.3.1 微内核操作系统 技术 .....	(281)
9.4.1 逻辑文件系统一般 存储结构 .....	(255)	10.3.2 面向对象操作 系统技术 .....	(284)
9.4.2 Linux 的虚拟文件 系统 .....	(255)	学后自测十 .....	(288)
9.5 Linux 的设备管理 .....	(261)	参考答案十 .....	(288)
9.6 Linux 的用户接口 .....	(263)	<b>参考文献</b> .....	(289)

# 第一章 操作系统概述

## 学习目标

- 了解操作系统的形成过程,建立起操作系统的整体概念;
- 熟悉操作系统的基本类型和服务方式;
- 掌握操作系统的定义、特征和功能。

## 学习重点、难点

- 操作系统的各种观点;
- 操作系统的基本类型和特征;
- 操作系统的服务和功能。

今天,任何一个计算机系统都需要配置相应的操作系统。无论是巨型计算机还是微型计算机,也无论是个人计算机系统还是计算机网络系统,如果离开了操作系统,都将无法运行。操作系统(Operating System)是现代计算机系统的重要组成部分,其他所有软件,例如,汇编程序、编译程序、数据库管理系统等系统软件以及各种应用软件,都将依赖于它的支持,并取得它的服务。

学习和掌握操作系统的基本原理和方法,是计算机科学技术发展的需要,它可以帮助我们从宏观上理解计算机系统的动态工作过程,为计算机研究和应用提供必要的理论和技术基础。

为使读者能尽快地对操作系统的整体概貌有一个初步了解,本章先从不同角度对操作系统进行较粗略的全面概述。

## 1.1 什么是操作系统

每一个使用过操作系统的人都得到过操作系统的服务与帮助。然而,要给操作系统下一个精确定义却并非易事。几十年来,人们从不同角度对操作系统有过许多不同的解释,但至今仍无一个统一的定义。以下是从诸多不同解释中选出的三种最有代表性的观点。

### 1.1.1 操作系统的虚拟机观点

这种观点认为,操作系统是添加在硬件上的第一层软件,是对硬件功能的首

次扩充和直接延伸。

众所周知,现代计算机系统由软件和硬件两大部分所组成。软件部分又可大致分为系统软件和应用软件两大类。系统软件用来管理计算机本身及应用软件;应用软件用来完成用户所要求的实际任务。硬件部分是指未配置任何软件的计算机(简称裸机)。

裸机又可分为物理器件、微程序(Microprogram)与机器语言三个层次。在这三个层次中,把物理器件作为硬件无可非议,但把微程序与机器语言划为硬件只是一种习惯。微程序通常被放在只读存储器中,是一层很原始的软件,用来直接控制物理器件,并为上一层软件提供一个更清晰的接口。从本质上讲,微程序是一个机器语言的解释器,它把机器指令再分解为一系列的操作步骤。例如,执行一条 ADD 指令,微程序首先需要确定运算数据所在的位置,然后取数,相加,最后存放运算结果。由微程序解释执行的这一套指令称为机器语言。机器语言实际上并不是硬件机器的组成部分,只不过由于硬件制造厂商通常把它放入机器手册的硬件部分进行描述,致使人们认为它是机器硬件的一部分。

如果计算机系统中没有配置操作系统,让用户直接使用裸机,那么就必须要对中央处理机、主存储器、时钟、终端、磁盘和其他输入/输出设备的管理细节全部交给用户去完成,这将使程序设计和计算机应用变得非常复杂和困难。正是为了避免这种实现上的复杂性给程序设计和计算机应用所带来的极大麻烦,人们才不断在裸机上添加软件,延伸裸机功能,构造基于裸机,但又比裸机更易于用户理解和编程的虚拟计算机,简称虚拟机(Virtual Machine)。这一层添加在裸机上的软件就是操作系统。可见操作系统是对硬件系统功能的首次扩充。

除操作系统以外的其他系统软件,如命令解释程序、编译程序、编辑程序等都位于操作系统之上。尽管它们也都属于系统软件,但与操作系统不同。操作系统是在核心态(Kernel Mode),或称管态(Supervisor Mode)下运行的一部分软件,硬件保护它免受用户程序的破坏;而解释程序、编译程序、编辑程序等则都在用户态(User Mode)下运行,它们不受硬件的保护。例如,用户可以自主修改一个编译程序,但无权自编一个磁盘中断处理程序。因为磁盘中断处理程序是操作系统的一部分,是受硬件保护的,硬件阻止用户对它进行修改。至于应用软件,则处在所有系统软件之上。计算机系统的这种抽象结构如图 1-1 所示。

按照这种观点,其他系统软件和应用软件全部都建立在操作系统的基础之上,操作系统实际上是向它们隐藏了硬件和硬件操作细节的真相,把它们对裸机的使用转化成了对虚拟机的使用。

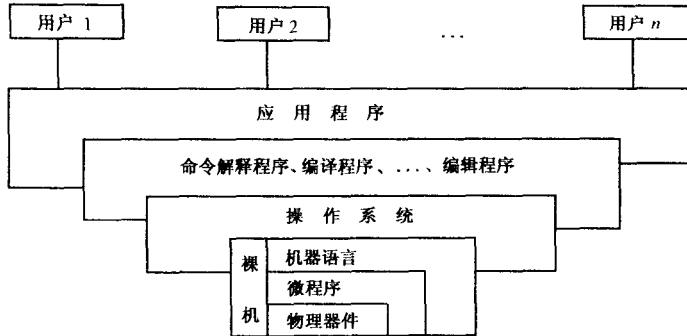


图 1-1 计算机系统的抽象层次结构

### 1.1.2 操作系统的资源管理观点

这种观点认为,操作系统是管理计算机系统资源的软件,它负责控制和管理计算机系统的全部资源,确保这些资源能被高效合理地使用,确保系统能够有条不紊地运行。

现代计算机系统的资源大致可分为硬件资源和软件资源两大类。硬件资源是计算机硬件系统的总和,它包括中央处理机、主存储器、输入/输出设备、辅助存储设备等。软件资源是系统中各种程序和数据的总和,这些程序和文件均以文件的形式保存在计算机系统中。从资源管理观点出发,根据操作系统所管理的资源的类型,可把整个操作系统分为处理机管理、存储器管理、设备管理和文件管理四大部分。其中,处理机管理负责 CPU 的运行和分配;存储器管理负责主存储器的分配、回收、保护与扩充;设备管理负责输入/输出设备的分配、回收与控制;文件管理负责文件存储空间和文件信息的管理。所有这些管理,构成了操作系统的资源管理功能。

### 1.1.3 操作系统的用户服务观点

这种观点认为,操作系统是一个为用户服务的大型程序。从用户角度看,当计算机配置了操作系统之后,用户就不再直接使用计算机硬件,而是利用操作系统所提供的命令和服务去使用计算机。或者说,用户在操作系统的帮助下能方便、快捷、安全、可靠地操纵计算机硬件和运行自己的程序。这就把操作系统看成了用户与计算机之间的一个接口。这一接口为用户提供了两种使用操作系统的方式:命令方式和系统调用方式。所谓命令方式,是指操作系统提供了一组联

机命令,用户可通过键盘或鼠标使用这些命令,并直接操纵计算机系统。所谓系统调用方式,是指操作系统提供了一组系统调用,用户可在自己的程序中通过调用相应的系统来操纵计算机。

综合上述各种观点,可以给出操作系统的解释:操作系统是控制和管理计算机硬件和软件资源,合理组织计算机工作流程以及方便用户的程序集合。

## 1.2 操作系统的形成过程

操作系统同其他任何事物一样,也有一个产生与发展的过程,下面对操作系统的形成过程做一个简单回顾。

### 1.2.1 人工操作阶段

在 20 世纪 50 年代以前的第一代计算机中是没有配置操作系统的,计算机只是由控制台控制的一个庞大的物理机器。当时,人们使用计算机的过程可大致描述为:首先由程序员将其编好的程序从纸带或卡片机上装入内存,然后再通过控制台上的按钮或开关启动程序执行,最后当程序运行完毕时,取下纸带和运算结果,开始下一个用户程序;依次重复上述过程。这种人工操作方式存在以下几个主要问题:

(1) 资源独占。当一个用户开始操作后,计算机中的全部资源都归该用户所有,直到程序运行结束或出现错误时才会把资源转让给下一个用户。

(2) 串行工作。人的手工操作与计算机的运行以及 CPU 与 I/O 设备之间都是按串行方式工作的。

(3) 人工干预。计算机是在人的直接联机干预下进行工作的。

可见,人工操作方式严重降低了计算机资源的利用率,形成了所谓的人工操作方式与机器利用率的矛盾,简称人一机矛盾。对早期的计算机来说,这种矛盾尚不突出,原因是那时的计算机本身所拥有的资源数量较少,计算速度也较慢,一般程序所需的计算时间要比其他操作时间(如装卸带、卡,I/O 等)占用的时间长。但是,随着 CPU 速度的提高、存储容量的增大和外部设备的增多,人一机矛盾就变得越来越尖锐,使得这种方式到了非改不可的地步。

### 1.2.2 单道批量处理阶段

为了解决上述问题,人们首先想到的是如何摆脱从一个用户程序过渡到另一个用户程序时的手工干预,使其转换能自动进行。这就产生了由计算机对一批用户程序进行自动处理的所谓批量处理技术。在批量处理方式中,一个用户

程序及其所需要的数据和操作命令的总和被称为一个作业。批量处理技术出现于 20 世纪 50 年代末期的第二代计算机中,它又可分为早期批量处理和脱机批量处理两个阶段。

### 1. 早期批量处理

早期批量处理方式是把若干个用户作业集中起来组成一批作业,并在内存中放置一个监督程序,由监督程序来负责实现对这批作业的处理和从一个作业过渡到另外一个作业的自动转换。

在这种处理方式下,先由操作员把一批用户作业的卡片叠放到读卡机上,然后由监督程序开始对该批作业进行自动处理。监督程序的处理过程是:先把读卡机上的这批作业全部输入到磁带上,然后按照某种策略从该批作业中选择一个作业调入内存,对其进行汇编或编译,并把汇编或编译结果装入内存,启动执行,运行结束后输出其计算结果。当第一个作业全部完成后,监督程序会自动选择下一个作业运行。重复上述过程,直到该批作业全部完成为止。这样,在监督程序处理第一批作业的同时,操作人员可以将第二批作业的卡片叠放到读卡机上,当监督程序处理完第一批作业后,便可自动地从读卡机上输入和处理第二批作业。这样,监督程序就可以不停地对一批批、一个个作业进行处理,从而实现了作业之间的自动转换,解决了人工操作阶段所存在的人工干预问题和人工操作与计算机之间的串行工作问题。

虽然这种处理方式提高了系统的处理能力,但作业的输出/输入和 CPU 的计算仍然是串行的。也就是说,作业信息由卡片送到磁带,再由磁带调入内存,以及计算结果在打印机上的输出,都是由 CPU 来处理的。这种 CPU 和 I/O 设备之间的串行工作方式大大降低了程序的执行速度。

### 2. 脱机批量处理

为解决早期批量处理所存在的问题,人们在批量处理中引入了脱机输入/输出技术,形成了脱机批量处理方式。这种处理方式的处理过程如图 1-2 所示,它在早期批量处理的基础上增加了一台功能较差的处理机。原来的那台处理机称为主处理机(简称主机),它专门负责计算工作,不直接与输入/输出设备打交道。新增加的这台处理机称为卫星机,专门负责输入/输出工作。

在图 1-2 中,卫星机负责把输入设备上的作业输入到输入磁带上和把输出磁带上的计算结果从输出设备上输出;主机直接从输入磁带上调入作业执行,并把计算结果送到输出磁带上。这样做使作业的输入和输出工作完全与主机脱离,并且卫星机的工作与主机的工作是同时进行的。因此,这种批处理方式解决了 CPU 与 I/O 设备之间的串行工作问题。

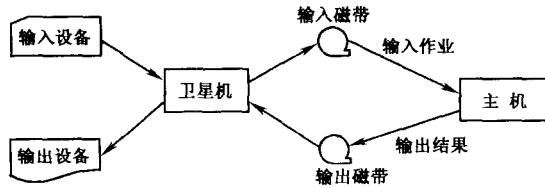


图 1-2 脱机批量处理

虽然批处理方式提高了系统的处理能力,但却带来了必须解决的保护问题。例如,监督程序、系统程序和用户程序之间是通过相互调用的方式来进行转移的,当目标程序企图执行一条非法指令时,整个系统就会停顿下来;另外,若程序陷入死循环,则整个系统也无法进行下去。更为严重的是它无法防止用户程序破坏监督程序的问题,潜伏着搞乱系统的危险。

### 1.2.3 执行系统阶段

从 20 世纪 50 年代末到 20 世纪 60 年代初,硬件方面获得了两个重要进展:一是通道的引入;二是中断的出现。所谓通道,实质上是一个功能单一、结构简单的 I/O 处理机,它独立于 CPU,并直接控制外部设备与内存进行数据传输,可代替上述卫星机的工作。所谓中断,开始是作为外部设备向中央处理机的“汇报”手段提出来的,即在输入/输出结束,或在硬件发生某种故障时,由相应硬件向 CPU 发出一个信号,使 CPU 停止正在执行的操作,而转去执行为处理该信号而设置的程序,中断处理完毕后 CPU 再回到原来的断点继续执行。而现代计算机系统中断概念已被大大扩展。

为了获得 CPU 和外部 I/O 设备在执行时间上的重叠,就必须提供中断处理程序和 I/O 控制程序,这样就把原来的监督程序扩大到了执行系统。可见,执行系统的程序可包括三大类:I/O 控制程序、中断处理程序和管理程序。这时,执行系统的程序已比较庞大,若把他们全部放到内存,会大大减少用户程序的可用空间。为此,可让那些所有程序都要用到的中断处理程序和 I/O 控制程序常驻内存,而其他部分放在外存。常驻内存的那部分程序称为执行程序。

执行程序与前述监督程序有着明显差别,它与其他程序的关系不是相互调用关系,而是一种控制关系,执行程序对其他程序拥有控制权。用户程序的输入/输出通过委托执行程序来实现,系统对错误的输入/输出要求提供自动检查,受托程序完成之后,再用中断信号通知执行程序,以保证系统的安全。此外,用户程序发生死循环也可以通过时钟中断进行检测处理;非法操作也可以通过非法



操作中断得到及时处理。

执行系统虽然较好地解决了 CPU 和 I/O 设备之间的操作并行问题,但内存中仍仅能存放一个程序,并且当这个程序因等待 I/O 而不能继续执行时,CPU 必须处于空闲状态。

#### 1.2.4 多道程序系统阶段

为解决执行系统所存在的问题,提高 CPU 的利用率,又引入了多道程序技术。多道程序技术的主要思想是,在内存中同时放入若干道用户作业或者说若干道用户程序,并允许他们交替执行,共享系统中的各种硬、软件资源。当一道程序因 I/O 请求而暂停执行时,CPU 便转去执行另一道程序。这种允许多道程序同时运行的系统称为多道程序系统。多道程序系统不仅使 CPU 得到了充分利用,同时还改善了 I/O 设备和内存的利用率。

从宏观上看,多道程序系统中的若干道用户程序是同时在系统中运行的,而从微观上看,这些程序则是在轮流使用 CPU,只是由于 CPU 的运行速度很快,人们感觉不到而已。多道程序的出现标志着操作系统的形成,最早出现的多道程序系统是多道批处理系统,之后又出现了分时系统、实时系统等。

虽然多道程序系统有效地提高了系统资源利用率,但实现多道程序则需要妥善解决下述一系列问题:

(1) 内存的分配和保护。在内存空间中同时驻留了多道程序,应为每道程序分配自己的内存空间,使它们既不因相互重叠而丢失信息,又不因某道程序出现异常而破坏其他程序。

(2) 处理机的管理和分配。多道程序共同使用一个中央处理机,这就必将引起各道程序对中央处理机的争夺,系统要协调它们之间的关系,既能使那些紧急的程序优先获得处理机,又能使各道程序都有得到处理机的机会。

(3) I/O 设备的管理和分配。计算机系统中的 I/O 设备数量一般都少于多道程序所需要的设备总量,这就必将引起各道程序对 I/O 设备的争夺。对此,系统应该能够进行协调,并能为各道程序分配相应的 I/O 设备。

(4) 文件存储空间的组织和管理。通常,在辅助存储器上以文件形式存放着大量的有用信息,为提高文件存储空间的利用率,加速对信息的检索速度,系统应对它们进行组织和管理。同时,为方便用户使用文件,系统还应该提供存储和检索文件信息的手段。

在多道系统中,解决上述问题的一组程序的集合构成了操作系统。