

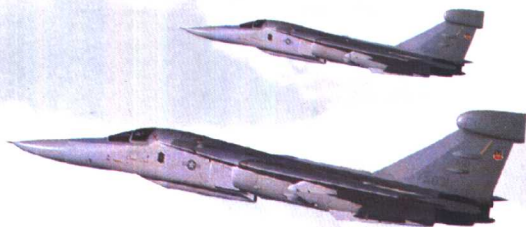
工科课程**提高与应试**丛书

- 涵盖课程重点及难点
- 精设典型题详解及评注
- 选配课程考试模拟及全真试卷

杨麦顺等 编

计算机操作系统

典型题解析及自测试题



西北工业大学出版社

【内容简介】本书是为了配合计算机操作系统原理的教学与研究生入学应试而编写的。全书内容分为两部分。第一部分典型题解析共分六章,每章包括内容提要、典型题解析和习题三个模块;第二部分包括2套自测模拟试题和2套研究生入学试题。附录给出了各章习题、自测题以及研究生入学试题的参考答案。全书突出了题目的典型性和代表性,旨在帮助读者掌握知识、拓宽思路、灵活应用。

本书可作为本科生和其他人员学习计算机操作系统原理的辅助教材,也可作为报考硕士研究生人员的复习参考。

图书在版编目(CIP)数据

计算机操作系统典型题解析及自测试题/杨麦顺,伍卫国,马晓农编. —西安:西北工业大学出版社,2002.4

(工科课程提高与应试丛书)

ISBN 7-5612-1460-X

I. 计… II. ①杨…②伍…③马… III. 操作系统(软件)—研究生—入学考试—解题 IV. TP316-44

中国版本图书馆 CIP 数据核字(2002)第 014129 号

出版发行: 西北工业大学出版社

通信地址: 西安市友谊西路 127 号

邮 编: 710072 电 话: (029)8493844

网 址: <http://www.nwpu.com>

印 刷 者: 陕西宝石兰印务有限责任公司

开 本: 850 mm×1168 mm 1/32

印 张: 6.75

字 数: 168 千字

版 次: 2002 年 5 月 第 1 版 2002 年 5 月 第 1 次印刷

印 数: 1~6 000 册

定 价: 10.00 元

前 言

计算机操作系统是计算机系统的核心软件,它负责管理和控制系统中所有硬、软件资源,以提高系统效率。计算机操作系统课作为计算机专业的主要专业基础课之一,内容涉及大量的概念、方法、原理,具有知识点多、抽象、难于理解、不易掌握等特点。为了帮助学生学好这门课程,以及系统地复习该课程,我们编写了本书。

本书分两部分。第一部分简要说明了操作系统原理课程所涉及的基本概念、作业管理、进程管理、存储器管理、设备管理和文件管理(文件系统)等各部分的重点和难点,通过对若干典型例题的分析和解析,帮助学生学习巩固所学内容,掌握正确的解题思路和方法。书中所选例题、习题大多取自于历年的本科生试题和研究生入学试题,针对学生在考试、作业、答疑中较常出现的问题,突出了题目的典型性和代表性。为了使学生有更多的练习机会,每章后都配有大量的习题。第二部分包括2套自测模拟题和2套研究生入学试题。附录给出了各章习题、自测题以及研究生入学试题的参考答案。

参加本书编写的有杨麦顺、伍卫国、马晓农等同志,在编写过程中西安交通大学陆丽娜教授对本书的总体内容、题目选型等提出了许多宝贵意见和建议,张小绶、楚丽平对本书所给部分习题和自测题进行了解答,在本书出版过程中我们得到了西北工业大学出版社的许多同志的帮助,尤其是编辑们的热心帮助和大力支持。在此对他们一并表示衷心感谢。

由于作者水平有限,时间仓促,书中错误在所难免,恳请广大读者批评指正。

编 者

2002 年 4 月

目 录

第一部分 典型题解析

第一章 操作系统引论	1
一、内容提要	1
二、典型题解析	12
三、习题 1	14
第二章 作业管理	16
一、内容提要	16
二、典型题解析	24
三、习题 2	31
第三章 进程管理	35
一、内容提要	35
二、典型题解析	56
三、习题 3	71
第四章 内存管理	83
一、内容提要	83
二、典型题解析	90
三、习题 4	108
第五章 设备管理	112
一、内容提要	112
二、典型题解析	120
三、习题 5	124
第六章 文件系统	128
一、内容提要	128

二、典型题解析	134
三、习题 6	144

第二部分 自测试题

自测试题 1	147
自测试题 2	150
1999 年西安交通大学研究生入学操作系统试题	152
2001 年西安交通大学研究生入学操作系统试题	155

附录习题、自测题答案

习题 1	157
习题 2	157
习题 3	159
习题 4	175
习题 5	183
习题 6	186
自测题 1	188
自测题 2	192
1999 年西安交通大学研究生入学操作系统试题参考答案 ...	196
2001 年西安交通大学研究生入学操作系统试题参考答案 ...	201
参考文献	207

第一部分 典型题解析

第一章 操作系统引论

一、内容提要

(一) 本章重点

本章的重点是操作系统的基本概念,操作系统的发展史,现代操作系统的特征,操作系统的结构,操作系统的职能,多道程序设计,操作系统的分类等。读者通过对本章的学习应对操作系统的有关概念有一个初步的了解。

1. 操作系统的概念

操作系统是控制和管理计算机系统的硬件和软件资源,合理地组织计算机工作流以及方便用户使用的程序集合。它实际上包含两个方面:一是为用户使用计算机系统提供接口,尽可能地方用户使用,从这个角度出发,可把操作系统看作虚拟机;二是操作系统要管理和控制计算机系统的各种硬软件资源,使其有条不紊的工作,最大限度地提高各种资源的利用率,这样可把它看作计算机系统资源的管理者。

2. 现代操作系统的特点

- ① 可移植性;
- ② 多处理器及可扩展性;
- ③ 分布式计算。

3. 操作系统的功能

从资源管理的角度来看,操作系统通常具备以下 5 个功能:

(1) 处理器管理

CPU 是计算机系统中宝贵的硬件资源之一,因此中央处理器(CPU)管理的主要任务是对处理器进行分配,并对其运行进行有效的控制与管理。

为提高计算机的利用率,操作系统采用了多道程序技术。为了描述多道程序的并发执行,引入了进程的概念。

进程是指程序及数据在计算机上的一次运行。在计算机系统中,进程不仅是独立运行的实体,要独立争夺资源,而且是构成系统的基本细胞,它能确切地反映系统的内在本质和特征,尤其是并发和动态的特征。因此,对于进程的管理和控制是操作系统的基本任务。

通过进程管理来协调多道程序之间的关系,以使 CPU 资源得到最充分的利用。在多道程序环境下,处理器的分配与运行是以进程为基本单位,因此对处理器的管理归结为对进程的管理。进程管理的主要任务就是按照某种原则进行资源分配。进程管理的主要内容包括:

- ① 进程控制;
- ② 进程同步;
- ③ 进程通信;
- ④ 进程调度。

(2) 存储器管理

内存是计算机系统另一宝贵的硬件资源,它的价格较昂贵,因此主机内存容量是有限的。当多道程序共享有限的内存资源时,如何为每道程序分配内存空间,使它们彼此隔离,互不干扰,尤其是当内存不够用时,如何虚拟扩充物理内存,把当前不运行的程序及数据及时调出内存,要运行时再将它从外存调入内存,等等,这些都是存储器管理的任务,它主要的功能包括:

- ① 内存分配;
- ② 内存保护;
- ③ 地址映射;
- ④ 内存扩充。

(3) 设备管理

设备管理是指计算机中除了 CPU 和内存以外的所有输入/输出设备的管理。外部设备的种类繁多,其功能差异很大,但设备管理的首要任务是为这些设备提供驱动程序或控制程序,以使用户不必详细了解设备及接口的技术细节,就可方便地对设备进行操作。另一个任务就是利用中断技术、通道技术和缓冲技术,使外围设备尽可能与 CPU 并行工作,以提高设备的使用效率。它的主要功能包括:

- ① 缓冲管理;
- ② 设备分配;
- ③ 设备处理;
- ④ 虚拟设备管理。

(4) 文件管理

文件是计算机系统中除 CPU、内存、外围设备等硬件资源之外的另一类资源,即软件资源。程序和数据以文件形式存放在外存储器(如磁盘、光盘、磁带)上,需要时再把它们装入内存。文件管理系统的任务是有效地组织、存储、保护文件,以使用户方便、

安全地访问它们。它的主要功能包括:

- ① 文件存储空间管理;
- ② 文件目录管理;
- ③ 文件存储控制;
- ④ 文件操作。

(5) 用户接口管理

为了方便用户使用,操作系统向用户提供了使用接口,接口(含图形接口和菜单接口)通常以命令、系统调用等形式呈现在用户面前,前者供用户在键盘或屏幕上使用,后者供用户在编程时使用。它的主要功能包括:

- ① 命令接口管理;
- ② 程序接口管理;
- ③ 图形接口管理。

4. 操作系统的产生和发展

任何一种事物都有其产生和发展的历史,操作系统也不例外,不过它是伴随着计算机体系结构的变化而发展的。计算机的发展主要是以元器件的发展作为发展的主要标志,形成了计算机发展的各个不同的阶段,它主要以不断减少成本、体积、功耗、增加计算机的运行速度和存储容量为其追求目标。而操作系统的发展是以不断地提高各种资源利用率和方便用户的使用为追求目标的。一般来说,操作系统的发展依赖于硬件的发展而形成了以下几个发展阶段:

- ① 第一代操作系统(20 世纪 50 年代);
- ② 第二代操作系统(20 世纪 60 年代初);
- ③ 第三代操作系统(20 世纪 60 年代中期至 70 年代中期);
- ④ 操作系统的进一步发展(70 年代中期之后)。

5. 操作系统的特征

(1) 并发性(程序的并发执行)

所谓并发执行是指两个或多个任务程序在同一时间间隔内同时或者交替地执行。由于程序的并发执行,有效地改善了系统资源利用率和提高了系统的吞吐量,使得系统更加复杂,所以操作系统必须具有控制和管理各种并发活动的的能力。

(2) 共享性(资源共享)

由于多道程序的引入使得系统中的多个进程共享系统中的软、硬件资源,即资源的状态不再由一个用户决定,而由多个用户决定。在计算机系统中对资源共享有两种方式:一种是互斥的使用方式,系统的资源可供多个用户使用,但在一段时间内只允许一个用户使用,等它使用完了才交给别的用户使用,这种资源我们称之为临界资源;另一种是同时使用方式,允许多个进程同时访问该类资源,如对磁盘的访问。程序的并发执行和资源共享是操作系统的两大基本特征,且两者又互为存在的条件,相辅相成,即资源共享是以程序的并发执行为先决条件。若系统不允许程序的并发执行,那就不存在资源共享。而系统不能对资源共享实施有效的管理,也必将影响到程序的并发执行。

(3) 不确定性

在多道程序运行中,由于运行环境的影响,程序的运行时间、运行顺序及同一程序或数据的多次运行结果等均具有不确定性。

(4) 虚拟性

虚拟是把一个物理上的客体变为若干个逻辑上的对应物。它体现在操作系统的方方面面:多道程序在单 CPU 的计算机上同时运行的机制使得多道程序好像独占一个 CPU;若干个终端用户分时使用一台计算机,好像每个用户独立地使用一台计算机;虚拟存储器使得主存为 1 MB 的计算机可以运行总容量为 5 MB

以上的程序;通过 SPOOLing 系统,可以把 I/O 设备扩充并改造为数量较多、方便使用的逻辑设备。这些都体现操作系统的虚拟性。

6. 多道程序设计技术

随着中断和通道技术的出现,产生了以存储器为中心的体系结构,输入/输出设备与 CPU 同时运行已成为可能,为提高处理机资源的利用率,软件上出现了多道程序技术。多道程序(multi-programming)是指在内存中同时装入几道作业(任务),主机(对于单 CPU 系统)以交替的方式同时处理多道程序,允许这些作业在系统中交替地运行,当某道作业发出新的资源请求或等待输入/输出传输时,就放弃 CPU,让另一道作业运行,使得处理机被充分的利用。在单处理机系统中,这些程序微观上只能交替地运行,而宏观上,我们可以认为它们是并行的,主机内同时保持和处理若干道已开始运行但尚未结束的程序。

7. 操作系统的分类

对于操作系统可有各种不同的分类方法,通常可按计算机的体系结构、运行环境以及服务对象来进行分类。尽管有各种不同的分类方法,但到目前为止,各种操作系统均属于下列操作系统之一或它们的组合:

- ① 单用户操作系统;
- ② 批处理操作系统;
- ③ 分时操作系统(分时系统);
- ④ 实时操作系统(实时系统);
- ⑤ 网络操作系统;
- ⑥ 分布式操作系统;
- ⑦ 多处理机操作系统(并行操作系统)。

对于以上各种操作系统必须弄清楚它们各自的设计思想、设计目标、主要特点、适用环境以及它们之间的区别。

8. 操作系统的结构

操作系统结构是指其内部的组成结构。早期的操作系统,由于系统规模较小,逻辑关系较简单,设计者往往只注重功能设计和效率,忽视了结构的设计,因操作系统设计不当,引起错误并造成惨重的损失,让人们记忆犹新。因此,人们在总结经验的基础上,认识到系统结构直接影响到系统的性能,从而越来越重视结构的设计,并逐步采用结构程序的设计方法来设计操作系统,使之成为结构清晰、易读易懂、适应性强、可靠性高、易于修改、易于证明其正确性的系统。

采用结构程序设计方法来设计操作系统,可以将操作系统看成一个整体模块,它由若干个模块按一定的结构方式组成。操作系统的结构,到目前为止,大体上有 6 种:模块组合结构、分层结构、管程结构、虚拟机器、客户机服务器结构及面向对象设计结构。

(1) 模块组合结构

模块组合结构也称无序模块结构、模块接口结构。所谓模块组合结构,就是将一个大系统分成若干个相对独立的模块(过程集),这些模块可以独立编制。为使每一模块不太复杂,又可把大模块划分成更小的模块,形成“积木式”的结构方式,并把这些模块按规定的接口(如转子、调用或借助通信区、工作单元等)连接起来。这种结构是到目前为止最常使用的一种组织方式。

(2) 分层结构

所谓分层结构,就是将整个操作系统分解成若干个基本模块,并按照一定的原则,将这些模块排列成若干层,各层之间只有单向依赖关系,即低层为高层服务,高层依赖于低层,各层之间不

能构成循环。

(3) 管程结构

所谓管程,是指共享资源的数据结构以及在其上的能为并发进程所执行的一组操作。一般一个管程由 4 大部分组成:

- ① 管程名 Monitor;
- ② 局限于管程的数据说明;
- ③ 在该数据结构上进行操作的一组过程;
- ④ 对局部数据赋予初值的语句(初启语句)。

所谓类程是指专用资源的数据结构以及在其上规定的全部操作。由于类程是在专用资源上进行操作的一组过程,所以它不存在同步操作。

一个类程由 4 大部分组成:

- ① 类程名 class;
- ② 局限于类程的数据说明;
- ③ 在该数据结构上进行操作的一组过程;
- ④ 初启语句。

管程与类程的主要区别在于管程是管理共享资源,将竞争共享资源的并发进程通过同步操作处理成顺序执行。类程是管理专用资源,类程被进程所调用,被看作进程的延伸,不同的进程调用各自的类程。

管程结构操作系统就是从系统功能和实现相结合的观点出发,从系统中提炼出管程、类程、进程等几种基本成分,将系统分解成由这些基本成分组成的模块,并将这些模块按一定的原则编入各层,核心在最内层,看作是管理 CPU 的专用管程,这种结构称为管程结构(也称为层次管程结构),这种结构具有以下特点:

- ① 从数据结构的角度来看,它是将数据结构及其上操作集中起来的一种抽象的数据类型;
- ② 从资源管理的角度来看,它将系统分成若干模块,用数据

表示抽象的系统资源,并根据共享资源和专用资源在管理上的差别来定义模块的类型和结构,从而引出了管程和类程的概念,它使系统的同步操作相对集中,从而增加了模块的相对独立性。

(4) 虚拟机器

对于一台完全无软件的计算机系统(裸机),即使功能再强,也必定是难于使用的。如果在裸机上覆盖一层 I/O 设备管理软件,用户便可以利用它所提供的 I/O 指令,来进行数据的输入/输出。此时用户所看到的机器,将是一台比裸机功能更强、使用更方便的机器。通常把覆盖了软件的机器称为虚拟机。如果再在第一层软件上覆盖一层文件管理软件,则用户可利用它所提供的文件存取指令,来进行文件的存取。这样,用户所看到的是一台功能更强的虚拟机。假如在此基础上再覆盖一层面向用户的窗口软件,用户便可以在窗口环境下方便地使用计算机,这样便形成一台功能更加强大的虚拟机。

由此可知,人们在计算机系统上覆盖一层软件后,系统功能便能增强一级。按此观点,操作系统可以包含若干层软件,在裸机上覆盖上操作系统,便可得到一台功能显著增强,使用极其方便的多层(台)虚拟机器。

(5) 客户机/服务器结构

采用客户机/服务器(Client/Server, C/S)结构构造一个操作系统的基本思想是:把操作系统分为若干进程,其中每个单一进程实现单独的一套服务(功能)。例如,在一个操作系统中设置主存服务、进程生成服务、处理机调度服务、网络服务、文件管理服务以及显示服务等。每一种服务对应一个服务器,每个服务都运行在“用户态”,循环检查是否有客户请求该服务器提供的某种服务。客户可以是一个应用程序,也可以是另一个操作系统的成分。客户机通过发送一条消息给服务器请求一项服务,运行在“核心态”的操作系统内核把消息传给服务器,由服务器执行具体

操作,其结果经由内核用消息返回给用户(客户)。

采用 C/S 结构构造的操作系统,其组成部件小且自成一个独立的子系统。由于每个服务器是以独立的“用户态”进程方式运行的,在这种情况下,即使单个服务器出现故障,也不会引起操作系统其他部分崩溃。此外,在多机系统或网络系统中,不同的服务器可以在不同的处理机或不同的节点上运行,从而使得这样构造的操作系统更适合于分布式计算环境。

采用 C/S 结构构造的操作系统的特点是:

- ① 简化了基本操作系统;
- ② 提高了系统的可靠性;
- ③ 适合于分布式计算环境。

(6) 面向对象设计结构

Bertrand Meyer 在他的《面向对象软件的构造》(Object-oriented Software Construction)一书中,把操作系统描绘成像一些“没有顶”的程序。正像其他大的软件系统一样,要找出驱动一个操作系统的单个主程序是困难的。因此,面向对象的方法不是试图去设计这样一个自顶向下的系统,而是开始把目标集中到软件要完成其工作所必须处理的数据上。对于一个操作系统来说,这类数据采用系统资源的形式、文件、进程、内存块等等。

围绕数据设计一个系统的主要目标是生成一个容易修改的软件。面向对象的软件使修改减小到最小,它所用的一种方法是通过把数据的物理表示隐藏在对象中,对象是一个数据结构,它的物理格式被隐藏在类型定义后面。它包含一组形式特性(称作属性),并且通过一组服务来对其进行操作。

以对象为基础构造一个操作系统具有以下十分明显的优点:

- ① 操作系统访问和操纵其资源是一致的。操作系统生成、删除和引用一个事件对象与它生成、删除和引用一个进程对象都是用相同的方法,即使用对象句柄。而且由于每个资源是一个对

象,跟踪资源的使用是通过监控对象的生成和使用来简单地实现的。

② 由于所有的对象是采用同样的方式被保护,因此安全措施得到简化。当某人试图访问一个对象,它的安全系统就介入并批准该操作,不管该对象是进程,或是一段被共享的内存,还是通信端口。

③ 在两个或多个进程之间,对象为共享资源提供了方便和一致性的范例。

④ 句柄是用来处理所有类型的对象。当两个进程各自打开一个对象的句柄时,这两个进程就共享这个对象。操作系统可以通过跟踪一个对象有多少个句柄被打开,来决定该对象是否仍在使用中。当它不再使用时,操作系统可以删掉该对象。

(二) 本章难点

本章的难点是对多道程序设计技术的理解。

多道程序设计是操作系统的理论基础,利用处理机和外设并行工作来提高系统的效率,减少处理机的等待时间,虽然从微观上看是串行的,但从宏观上却是并行的,这种并行是“伪并行”。

多道程序是几道程序共享系统资源,使得处理机、内存和外设资源更加充分地利用,但多道程序的引入,必然给操作系统的理解、设计和调试带来了复杂的因素。

多道程序设计技术,得到了计算机硬件的中断系统和通道技术的支持。现代操作系统一般都采用基于多道程序设计的技术。

实现多道程序设计,必须解决以下几个问题:

- (1) 存储保护和地址重定位;
- (2) 处理机管理和调度;
- (3) 资源的管理和分配。

多道程序设计具有多道、宏观上并行以及微观上串行的