

所见的是暂时的，所不见的是永远的



# 计算机的心智 操作系统 之哲学原理

Computer's Mind  
Philosophical Principles of Operating Systems

邹恒明 著



机械工业出版社  
China Machine Press

所见的是暂时的，所不见的是永远的

TP302.1/13

2009

计算机的心智

# 操作系统 之哲学原理

Computer's Mind

Philosophical Principles of Operating Systems

邹恒明 著



机械工业出版社  
China Machine Press

本书集中精力对操作系统的核心内容进行分析,包括操作系统发展的历史背景、进程与线程、内存管理、文件系统、输入与输出、多核环境下的进程调度和操作系统设计。本书用大量生活实例,生动解释了操作系统中的主要难点和模糊点:锁的实现、同步机制的发展轴线、纯粹分段到段页式的演变、多核环境下的进程同步与调度和操作系统设计等内容,而放弃了对操作系统核心以外内容,如安全、多媒体系统、虚拟机技术、光盘技术等论述。本书重点突出、逻辑清晰、内容连贯,便于学生顺利掌握操作系统的核心内容。

本书层次丰富、涵盖操作系统的所有核心内容,适合作为国内高校计算机及相关专业本科生操作系统课程的教材,也是了解计算机操作系统原理不可多得的参考书。

版权所有,侵权必究。

本书法律顾问 北京市展达律师事务所

### 图书在版编目(CIP)数据

计算机的心智:操作系统之哲学原理/邹恒明著. —北京:机械工业出版社,2009.4

ISBN 978-7-111-26642-6

I. 计… II. 邹… III. 操作系统 IV. TP316

中国版本图书馆CIP数据核字(2009)第041558号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:金纯

北京京北印刷有限公司印刷

2009年4月第1版第1次印刷

186mm×240mm·20.25印张

标准书号:ISBN 978-7-111-26642-6

定价:38.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

本社购书热线:(010) 68326294

# 前言 PREFACE



In Pursuit of Absolute Simplicity 求于至简，归于永恒

谨以此书献给夫人蕾蕾，女儿雨洁、雨蓉、雨恒和雨宜。

当你在电脑上玩游戏的时候，当你在电脑上与朋友聊天的时候，当你编写完一个程序需要加载运行的时候，你有没有一种像在观看魔术的感觉？编写好的程序能够编译运行，计算出结果，并显示或打印出来。你有没有觉得它很神秘？

如果想揭开这层神秘的面纱，你就得学习操作系统。

因为操作系统是掌控计算机运行的系统，在学习它的过程中，读者能够了解到程序在计算机上运行的全景，或者说我们所认为的全景。之所以这么说，是因为精确了解程序在计算机上运行的全景是极其困难的（有人认为这根本就是不可能的）。当然，这里的程序指的是有一定规模的程序，而不是那种只有几行代码的小程序。从某种程度上来说，没有人敢肯定自己清楚计算机在任何一个时刻所处的状态。例如，在多流水线计算机上，如果发生中断或异常，我们根本就得不到一个精准的状态。唯一能做的就是推倒重来。



图1 风靡世界的游戏“第二生命”

## 计算机的心智

人有心智吗？我想所有人都会回答：有！人的心智就是人的灵气。这是每一个人的生命之气。就是这个灵气赋予了人丰富的思维、感受和行动能力。

那么计算机有心智吗？这不是一个诡秘或者搞笑的问题。

人们通常认为能够运动的生命都是有灵气的，既然计算机能够完成一些人脑才能够完成的理性任务，它当然也有心智！而这个心智就是操作系统。因为操作系统赋予了计算机以活力。虽然读者有可能尚不明白操作系统是怎么一回事，但也许知道没有操作系统，现代计算机是运转不起来的（这里需排除远古时代的古老计算机）。操作系统作为计算机赖以运转的控制中心，称其为计算机的心智可谓恰如其分。

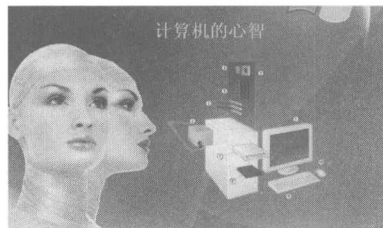


图2 计算机的心智就是操作系统

## 操作系统的奥秘

记得小时候常常念的一首诗是这样的：

从小时候就开始数了。  
数到懂事、数到成熟，  
还没有数清。  
天上的星星为什么数不清呢？  
像记忆和幻想，  
永远背负着固执的迷……

对于许多大学是学计算机及相关专业的同学来说，操作系统就像是天上的星星（如图3所示），隐藏着一个固执的迷，永远学不清楚。不过，操作系统真的难以学清楚吗？



图3 理解操作系统有点类似于数清楚天上的星星

不是的。学不清楚是因为没有看到其背后的奥秘。这个奥秘不是所有人都知道的。即使是研究操作系统的人也不一定意识到它，初学计算机者自然就更加不会注意了。

那么这个奥秘是什么呢？

天上的星星数不清是因为我们试图做的事情是数星星。如果我们换个角度，不去数星星，而是寻找到星星的设计师，让他告诉我们星星的数量，不就数清楚了吗？

这也正是学习操作系统的奥秘。要理解操作系统，就要寻找到操作系统的设计师们，让他们告诉我们操作系统所蕴含的所有秘密。当然，这里的寻找设计师并不是真的找来他们，因为找到所有的设计师是不可能的。这里的设计师指的是一种抽象，一种所有设计师所共有的人生哲学，因为设计师们在设计操作系统时会不自觉地将自己的思维或人生追求构造在操作系统里，从而赋予了操作系统以心智，而操作系统也就在这种心智的指挥下亘古运行着。

## 操作系统之哲学原理

正如前面所述，让设计师告诉我们操作系统秘密是理解操作系统的最好办法。他们所用的载体就是其所遵循的生活哲学，这些生活哲学就是操作系统所遵循的哲学原理。

本书就是试图从这些哲学原理（也就是人类生活哲学）的视角来阐述操作系统，从而揭

开操作系统的神秘面纱，令其不再晦涩难懂。

例如，CPU 管理（进程与线程）、内存管理（虚拟存储）、外存管理（文件系统）、I/O 管理（输入与输出）等操作系统的核心机制不外乎是资源的管理，它们都遵循着一切人类资源管理的基本原则，即如何有效地发掘资源、监控资源、分配资源和回收资源。

除了提供管理的功能外，操作系统还需要保证自己的正常运转，即它必须尽力使自身不发生失效或崩溃，因为这是提供其他一切功能的基础。这与人类把确保自身健康生活作为开发利用资源的前提是一个道理。

如果我们把握了资源的根本属性，即资源管理必然涉及共享和竞争的管理，理解了操作系统必须首先保障自己的正常运转，就会理解操作系统的一切行为。前者指引着操作系统功能的设计与进化，后者则推动着操作系统可靠性地演变。

资源管理也好，保证自身的正确性也好，它们都有着根本的线索。这条根本线索就是人类在长期的生活实践中摸索出来的管理社会和保障自身安全的各种办法。这些办法是随着人类哲学思维的变化而改进的。因此，只要明白了人类的哲学思维，就能明白操作系统所遵循的哲学原理，进而明白整个操作系统的设计与构造。

除了使操作系统易于理解外，从哲学的层面阐述操作系统的原理还有如下好处：

- 操作系统可以变化，但支持其存在的哲学原理是不变的。这样，本书的内容可以在操作系统不断演变的环境下保持有效，而不会像其他书的内容，随着时间的推移而过时。
- 对于很多人来说，操作系统所采取的机制、策略和手段看上去十分枯燥，如果从哲学原理上给它们赋予人性的特点，这些机制、策略和手段便不再枯燥。

通过将人生哲学与操作系统联系起来，从操作系统哲学原理的层次阐述操作系统的核心技术，就能够理解掌握操作系统的精髓。

## 本书内容安排

为清楚地阐述操作系统的哲学原理，也为了使内容显得紧凑，逻辑上一气呵成，本书只选择了操作系统的核心内容进行分析，放弃了对操作系统核心以外内容，如安全、多媒体系统、虚拟机技术、光盘技术等论述。本书集中精力对操作系统发展的历史背景、进程与线程、内存管理、文件系统、输入与输出、多核环境下的进程调度和操作系统设计进行了哲学原理层面的分析与论述。对内容的这种安排有如下好处：

- 可使本书重点突出、逻辑清晰、内容连贯，便于学生顺利掌握操作系统的核心与关键。
- 操作系统的核心内容经过长久的研究与实践，已经变得较为稳定并且形成了公认的标准，讲解起来没有歧义。
- 操作系统的非核心部分由于研究的时间短，工业界参与的程度较低，并无公认的标准，论述起来要么不全面，要么显得凌乱，使刚刚接触操作系统的读者感到迷惑。
- 只要掌握了核心内容的原理，读者便能通过自学掌握操作系统核心以外的知识。

本书覆盖全国硕士研究生入学统一考试计算机学科专业基础综合考试大纲中操作系统全部内容。

本书一共分为7篇22章。7篇分别是基础原理篇、进程原理篇、内存原理篇、文件原理篇、I/O原理篇、多核原理篇和操作系统设计原理篇。本书的内容结构如图4所示。

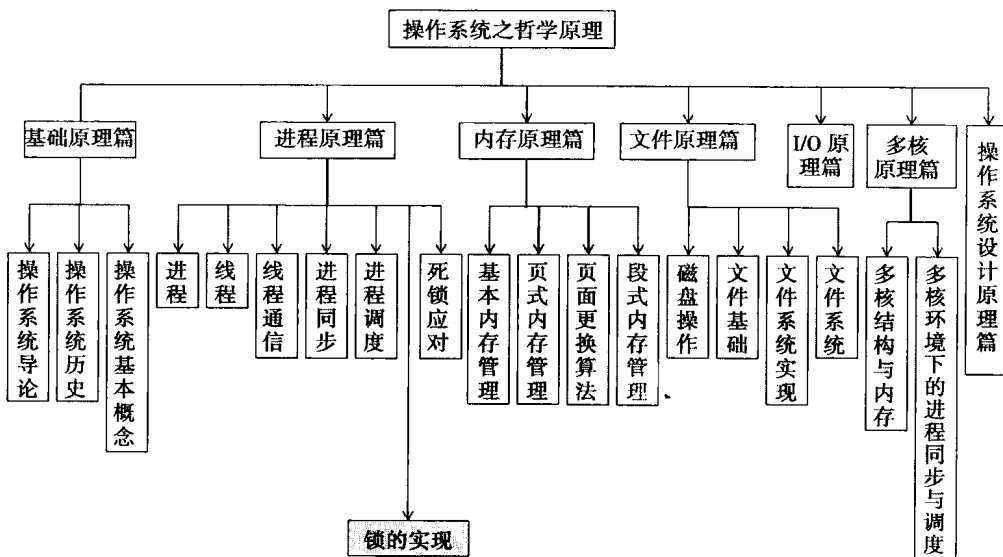


图4 本书内容结构

## 基础原理篇

该篇包含第1章至第3章的内容。第1章的内容包括智者的挑战、人造学科、程序是如何运行的、什么是操作系统、魔术与管理、用户程序与操作系统、操作系统范畴和为什么学习操作系统。第2章探讨操作系统演变的主要过程：从单一控制终端单一操作员，到批处理、多道批处理、分时操作系统、实时操作系统、现代操作系统；对商业操作系统演变的过程进行分析，然后探讨操作系统分类和操作系统的未来发展趋势。第3章简要回顾计算机硬件基本知识，探讨什么是“抽象”，讲解用户态与内核态，阐述操作系统结构、系统调用、操作系统的壳等知识。

## 进程原理篇

该篇对操作系统最为核心的概念“进程”进行讲解，包括第4章至第10章的内容。第4章阐述的内容包括进程出现的逻辑必然性、多道编程的效率、进程的创建和消亡、进程的状态及其转换、进程与地址空间、进程管理和进程模型的缺陷。第5章讲解的内容包括线程、线程管理、线程的用户态、内核态和混合态实现、现代操作系统的线程实现模型、多线程之间的关系、线程主要考虑的问题。第6章的内容包括为什么要通信、管道、记名管道、套接字、信号、信号量、共享内存、消息队列等。第7章的内容包括为什么同步、同步的目的、锁原语的进化、睡觉与叫醒原语、信号量、管程、消息传递和栅栏。第8章讲解的内容包括调度的目标、先来先服务、时间片轮转、短任务优先、优先级调度、混合调度、实时调度等算法，并对优先级倒挂和线程的不确定性进行讨论。第9章讲述如何使用中断启用和禁止、测试与设置来实现锁原语。第10

章对死锁的产生、发展、防止与避免进行讲解，并讨论死锁、活锁和饥饿的关系。

## 内存原理篇

这篇对操作系统的另外一个重要构成部分“内存管理”进行阐述，包括第11章至第14章的内容。第11章讲述内存管理的环境、内存管理的目标、虚拟内存、操作系统在内存中的位置、单道编程的内存管理、固定加载地址、多道编程的内存管理、固定分区、非固定分区、交换、地址翻译、闲置空间管理等内容。第12章的内容包括基址极限的问题、分页管理、页表、页面翻译过程、分页管理系统的优缺点、多级页表、地址翻译速度、锁住页面、内存抖动和页面尺寸设计。第13章对页面更换算法的来龙去脉、欲达到的目的、各种具体的页面更换算法进行细致讲解。第14章的内容包括分段管理系统、分段的优缺点、段号与寻址位数，并对否定之否定在内存管理模式发展过程中的作用进行讨论。

## 文件原理篇

这篇对操作系统的第三个核心构件“文件系统”进行讲解，包括第15章至第18章的内容。第15章讲述的内容包括磁盘结构、磁盘访问速度、磁盘的操作系统界面、磁盘访问过程和磁盘调度。第16章讲述为什么需要文件系统、什么是文件系统、文件系统的目的、文件的基本知识、文件的存储结构、文件类型、文件访问、文件属性、文件操作、文件夹、相对与绝对路径、共享与链接、内存映射的文件等内容。第17章的内容包括文件系统分布、文件的实现、文件夹的实现、共享文件的实现、磁盘空间的管理等。第18章的内容包括文件安全性能（文件访问控制、访问控制表、能力表）、文件可靠性能（持久性、一致性、日志、交易、随影、一致性检查）和文件系统的效率性能（提前读取、减少磁臂移动距离、日志结构的文件系统LFS）。

## I/O 原理篇

这篇对计算机与外界进行沟通的机制“输入与输出”进行讲解。本篇仅有一章（第19章），讨论的内容包括输入输出的重要性和目的、I/O硬件的哲学原理、物理I/O模式（专有通道I/O、内存映射的I/O、复合I/O、DMA）、输入输出软件之哲学原理、软件I/O模式（可编程I/O、中断驱动I/O、DMA）、I/O软件分层和设备驱动程序等。

## 多核原理篇

这篇对新出现的多核技术进行讲解。重点讨论多核环境给操作系统带来的影响。全篇分为多核结构和多核操作系统两章。第20章讲解的内容包括多核处理器结构（超线程结构、多核结构、多核超线程结构）、多核内存结构（UMA、NUMA、COMA、NORMA）、对称多核处理器计算机的启动过程、多处理器之间的通信和SMP缓存一致性等。第21章的内容包括多核进程同步、多核环境下的软件同步原语、旋锁及其实现、队列旋锁、多核环境下的进程调度、多核环境下的能耗管理和多核系统性能。



## 操作系统设计原理篇

这篇从高屋建瓴的角度对操作系统设计的十条哲学原理进行阐述。显然，操作系统的设计原则有很多，本篇选取的只是这诸多原则里面非常重要的十条。第22章将从操作系统和人类社会两个层面对这十条原理进行论述与比较，以使读者更加清楚地明白操作系统就是人类社会在计算机里面的反映。操作系统的其他设计原则读者可自行发现。

### 本书的特点

相对于国内外其它操作系统教材，本书的独特性体现在四个方面：逻辑导向，通过逻辑推理将核心原理演绎出来；联系生活，用人所熟知的生活实例来揭示奥秘；抽象提升，从哲学高度进行阐述以将各种原理串成有机整体；知识整合，引入相关编译和计算机组成的知识来加深读者对细腻之处的把握。这些特点赋予了本书风格上更加清新、内容上更加丰富、逻辑上更加严谨、叙述上更加幽默、解说上更加深刻、和层次上更加优美的引人入胜的效果。

读者在阅读学习完本书后，将达到如下目标：

- 了解操作系统在计算机软硬件整个体系中的中心主导作用。
- 掌握操作系统的基本概念、原理、技术和实现机制。
- 理解操作系统原理背后的人文背景与历史动机。
- 运用操作系统知识来分析和解决问题。
- 掌握操作系统设计的原理，为以后设计操作系统打下基础。

这里需要提醒的是，本书阐述的是操作系统的原理，它不依赖于任何具体的实现，而是凌驾于所有具体商业操作系统的进程实现之上。即本书所阐述的思想和原理对所有操作系统都适用。但具体商业操作系统在应用这些原理时可以有很灵活的方式。事实上，具体的商业操作系统在应用这些原理时确实采取了不同的方式，有的更为精密，也有的偷工减料。另外，由于我们注重的是原理，对个体机制实现时采取的数据结构通常不作琐细的论述，而是点到为止。这是因为数据结构必须以真正的操作系统为蓝本进行讲解，而真正的商用操作系统使用的数据结构通常非常复杂，对此进行繁琐的讲解将把学生弄得晕头转向，而不利于对操作系统核心原理的把握。

当然了，如果要达到能够设计开发真正商业操作系统的境界，读者还需要进行“操作系统工程”或“操作系统实现”的学习。而这种工程或实现的课程通常以具体的操作系统为对象进行讲述。这些具体的操作系统可以是 Windows、UNIX、Linux，当然也可以是其他一些非主流商业操作系统。如果能够将本书阐述的原理与操作系统工程或实现相结合，将取得更好的效果。

最后，本作者感谢下列人士为本书审阅书稿：上海交通大学的陈凌峰、张漳、顾夏中、徐燕和美国密歇根大学的鞠晓恩，其中陈凌峰和张漳将本人的讲课做了原始记录。

现在就让我们一起来揭示秘密，数清操作系统里的星星吧。

# 目 录 CONTENTS

## 前言

## 第一篇 基础原理篇

第1章 操作系统导论	2
引子: 智者的挑战	2
1.1 人造学科	3
1.2 程序是如何运行的	5
1.3 什么是操作系统	7
1.4 魔幻与管理	8
1.5 用户程序与操作系统	9
1.6 操作系统的范畴	11
1.7 为什么学习操作系统	12
思考题	13
第2章 操作系统历史	14
引子: 操作系统进化的推动因素	14
2.1 第一阶段: 状态机操作系统 (1940年以前)	15
2.2 第二阶段: 单一操作员、单一控制 端操作系统(20世纪40年代)	16
2.3 第三阶段: 批处理操作系统 (20世纪50年代)	16
2.4 第四代: 多道批处理操作系统 (20世纪60年代)	18
2.5 第五代之一: 分时操作系统 (20世纪70年代)	19
2.6 第五代之二: 实时操作系统	20
2.7 第六代: 现代操作系统 (1980年以后)	21
2.8 操作系统的演变过程	22

2.9 操作系统的未来发展趋势	25
思考题	26
第3章 操作系统基本概念	27
引子: “差不多”精神	27
3.1 计算机硬件基本知识	28
3.2 抽象	31
3.3 内核态和用户态	31
3.4 操作系统结构	33
3.5 进程、内存和文件	35
3.6 系统调用	36
3.7 壳	37
思考题	39

## 第二篇 进程原理篇

第4章 进程	42
引子	42
4.1 进程概论	43
4.2 进程模型	44
4.3 多道编程的效率	44
4.4 进程的产生与消失	46
4.5 进程的层次结构	47
4.6 进程的状态	47
4.7 进程创立	49
4.8 进程与地址空间	49
4.9 进程管理	50
4.10 进程的缺陷	52
思考题	52
第5章 线程	53
引子	53
5.1 进程的分身术——线程	54
5.2 线程管理	55

5.3	线程的实现方式	56	8.8	其他调度算法	102
5.4	现代操作系统的线程实现模型	60	8.9	实时调度算法	103
5.5	多线程的关系	60	8.10	高级议题：调度异常之优先级 倒挂	104
5.6	讨论：从用户态进入内核态	61		思考题	106
5.7	讨论：线程的困惑——确定性 与非确定性	62	第9章	锁的实现	107
	思考题	63		引子：锁的实现哲学	107
第6章	线程通信	64	9.1	以中断启用与禁止来实现锁	108
	引子	64	9.2	以测试与设置指令来实现锁	109
6.1	为什么要通信	64	9.3	以非繁忙等待、中断启用与禁止 来实现锁	111
6.2	线程对白：管道、记名管道、套接字	65	9.4	以最少繁忙等待、测试与设置 来实现锁	113
6.3	线程电报：信号	69	9.5	中断禁止、测试与设置	116
6.4	线程旗语：信号量	70		思考题	117
6.5	线程拥抱：共享内存	71	第10章	死锁应对	118
6.6	信件发送：消息队列	71		引子	118
6.7	其他通信机制	72	10.1	发生死锁的原因	118
	思考题	72	10.2	死锁的描述	120
第7章	进程同步	73	10.3	死锁的4个必要条件	121
	引子	73	10.4	哲学家就餐问题	122
7.1	为什么要同步	73	10.5	死锁的应对	122
7.2	线程同步的目的	75	10.6	消除死锁的必要条件	128
7.3	锁的进化	75	10.7	银行家算法：冒险的代价	130
7.4	睡觉与叫醒：生产者与消费者问题	81	10.8	哲学家就餐问题之解	132
7.5	信号量	84	10.9	讨论：死锁的思考——综合 治理	134
7.6	锁、睡觉与叫醒、信号量	87	10.10	讨论：死锁、活锁与饥饿	134
7.7	管程	88		思考题	135
7.8	消息传递	91			
7.9	栅栏	93			
	思考题	93			
第8章	进程调度	96			
	引子	96			
8.1	调度的目标	97	第三篇	内存原理篇	
8.2	处理器调度的总体目标	97	第11章	基本内存管理	138
8.3	先来先服务调度算法	98		引子	138
8.4	时间片轮转	98	11.1	内存管理的环境	139
8.5	短任务优先	99	11.2	内存管理的目标	140
8.6	优先级调度	101	11.3	虚拟内存的概念	140
8.7	混合调度算法	102	11.4	操作系统在内存的位置	142
			11.5	单道编程的内存管理	143

11.6	多道编程的内存管理	144				
11.7	闲置空间管理	150				
	思考题	152				
<b>第12章</b>	<b>页式内存管理</b>	153	<b>第四篇</b>	<b>文件原理篇</b>		
	引子	153				
12.1	基址极限管理模式的问题	154	<b>第15章</b>	<b>磁盘操作</b>	194	
12.2	分页内存管理	156		引子	194	
12.3	分页系统的优缺点	160		15.1	磁盘组织与管理	195
12.4	翻译速度	161		15.2	磁盘的结构	195
12.5	缺页中断处理	163		15.3	盘面的结构	196
12.6	锁住页面	163		15.4	磁盘驱动器的访问速度	197
12.7	页面尺寸	164		15.5	操作系统界面	198
12.8	内存抖动	165		15.6	磁盘调度算法	199
	思考题	167			思考题	202
<b>第13章</b>	<b>页面更换算法</b>	168	<b>第16章</b>	<b>文件基础</b>	203	
	引子	168		引子	203	
13.1	页面需要更换	169		16.1	为什么需要文件系统	203
13.2	页面更换的目标	169		16.2	什么是文件系统	204
13.3	随机更换算法	170		16.3	文件系统的目标	205
13.4	先进先出算法	170		16.4	文件的基本知识	205
13.5	第二次机会算法	171		16.5	从用户角度看文件系统	205
13.6	时钟算法	172		16.6	地址独立的实现机制：文件夹	212
13.7	最优更换算法	172		16.7	文件系统调用	214
13.8	NRU 算法	173		16.8	内存映射的文件访问	215
13.9	LRU 算法	174			思考题	216
13.10	工作集算法	179	<b>第17章</b>	<b>文件系统实现</b>	217	
13.11	工作集时钟算法	181		引子	217	
13.12	页面替换策略	181		17.1	文件系统的布局	218
	思考题	182		17.2	文件的实现	219
<b>第14章</b>	<b>段式内存管理</b>	184		17.3	目录实现：地址独立的实现	226
	引子	184		17.4	闲置空间管理	231
14.1	分页系统的缺点	185			思考题	232
14.2	分段管理系统	186	<b>第18章</b>	<b>文件系统</b>	233	
14.3	分段的优缺点	188		引子	233	
14.4	段页式内存管理	189		18.1	文件系统访问控制	234
14.5	段号是否占用寻址字位	190		18.2	主动控制：访问控制表	235
14.6	讨论：否定之否定的嵌套——纯粹 分段与逻辑分段、分页与段页	191		18.3	能力表	236
	思考题	192		18.4	访问控制的实施	238
				18.5	文件系统性能	239
				18.6	提高系统性能的方法	245
				18.7	文件系统设计分析： 日志结构的文件系统	248

18.8 海量数据文件系统 .....	250	21.7 旋锁 .....	286
思考题 .....	251	21.8 其他同步原语 .....	289
<b>第五篇 I/O 原理篇</b>		21.9 多核环境下的进程调度 .....	289
第 19 章 输入输出 .....	254	21.10 多核环境下的能耗管理 .....	292
引子 .....	254	21.11 讨论: 多核系统的性能 .....	293
19.1 什么是输入输出 .....	255	思考题 .....	295
19.2 输入输出的目的 .....	256	<b>第七篇 操作系统设计原理篇</b>	
19.3 输入输出硬件 .....	256	第 22 章 操作系统设计之原理 .....	298
19.4 输入输出软件 .....	262	引子 .....	298
19.5 I/O 软件分层 .....	266	22.1 操作系统设计的追求 .....	300
思考题 .....	269	22.2 操作系统设计的第 1 条哲学原理: 层次架构 .....	300
<b>第六篇 多核原理篇</b>		22.3 操作系统设计的第 2 条哲学原理: 没有对错 .....	301
第 20 章 多核结构与内存 .....	272	22.4 操作系统设计的第 3 条哲学原理: 懒人哲学 .....	302
引子 .....	272	22.5 操作系统设计的第 4 条哲学原理: 让困于人 .....	303
20.1 以量取胜 .....	273	22.6 操作系统设计的第 5 条哲学原理: 留有余地 .....	304
20.2 多核基本概念 .....	273	22.7 操作系统设计的第 6 条哲学原理: 子虚乌有——海市蜃楼之美 .....	305
20.3 多核的内存结构 .....	277	22.8 操作系统设计的第 7 条哲学原理: 时空转换——沧海桑田之变 .....	305
20.4 对称多处理器计算机的启动 过程 .....	279	22.9 操作系统设计的第 8 条哲学原理: 策机分离与权利分离 .....	305
20.5 多处理器之间的通信 .....	279	22.10 操作系统设计的第 9 条哲学原理: 简单为美——求于至简、 归于永恒 .....	306
20.6 SMP 缓存一致性 .....	281	22.11 操作系统设计的第 10 条哲学原理: 适可而止 .....	306
20.7 多处理器、超线程和多核的 比较 .....	281	思考题 .....	307
思考题 .....	282	结语 .....	308
第 21 章 多核环境下的进程同步 与调度 .....	283	参考文献 .....	310
引子 .....	283		
21.1 多核环境下操作系统的修正 .....	284		
21.2 多核环境下的进程同步与调度 .....	284		
21.3 多核进程同步 .....	284		
21.4 硬件原子操作 .....	285		
21.5 总线锁 .....	285		
21.6 多核环境下的软件同步原语 .....	286		

# PART ONE

## 第一篇 基础原理篇

对于任何一门课程来说，首要探讨的问题就是这门课的主题到底是什么？对于刚接触操作系统的入门者来说，自然想到的问题当然也会是操作系统到底是什么东西。回答这个问题是本篇的职责。此外，操作系统作为计算机的核心控制系统，它在计算机运行过程中扮演什么角色？它的来历是什么？它有一些什么基本概念？我们应该如何看待操作系统？它是如何参与到程序的执行过程中的？这些也是学习操作系统需要了解的基本问题。

本书的基础原理篇就是针对上述问题而写成。它对这些问题进行解答和讨论，并为我们接下来介绍操作系统的核心功能部件打下基础和铺垫。本篇包含第1章至第3章内容。第1章的内容包括智者的挑战、人造与神造、程序是如何运行的、什么是操作系统、魔术与管理、用户程序与操作系统、操作系统范畴和为什么学习操作系统。第2章探讨操作系统演变的主要过程：从单一控制终端单一操作员，到批处理、多道批处理、分时操作系统、实时操作系统、现代操作系统；对商业操作系统演变的过程进行分析，然后探讨操作系统分类和操作系统的未来发展趋势。第3章简要回顾计算机硬件基本知识，探讨什么是“抽象”，讲解用户态与内核态，阐述操作系统结构、系统调用、操作系统的壳等知识。

本篇最为重要的核心思想是操作系统在计算机运行过程中扮演的角色：魔术师和管理者。魔术师将丑陋变得美好，将没有变为有，将少变为多；而管理者则对所有计算机资源进行管理以达到公平和效率的“双料”境界。对操作系统这两个角色的理解将非常有助于对进程、线程、虚拟内存、文件系统和输入输出系统的掌握。



化丑为美：魔术师是操作系统扮演的一个根本角色

# 第 1 章 操作系统导论

## 引子：智者的挑战

西方有一个著名的故事，名曰：智者的挑战。相传很久以前，有座村子里住着一位智者。同村有个年轻人学到一些知识后就想来挑战这位智者。于是，年轻人想到了一个方法。他来到智者面前，将双手放在背后，问这个智者：

“我刚刚从树上抓了只鸟，现在在我手上。你能告诉我，这只鸟是活的还是死的呢？”

这是一个诡诈的问题。因为，如果回答“是活的”，则这个年轻人只需要在背后将鸟掐死，然后给智者看这只死鸟；如果回答“是死的”，则这个年轻人只需要将鸟放飞即可。这样，无论智者如何回答，年轻人都可以让智者答错，然后可以大大地嘲笑一番智者的水平。

作为智者，当然一眼看出了年轻人的诡计。但是又不能不回答这个问题。因为不回答问题等于承认答不上来，当然也就不是什么智者了。但如何回答呢？

智者的回答简洁、精妙，甚至妙不可言：“As you will”。

乍看上去，这个答案没有什么神奇之处。但如果读者的英语水平很高，就可以看出其中的奥妙。will 这个词在英语中的意思是意愿或意志。因此这个答案的意思是：这只鸟的死活与年轻人的意志保持一致；年轻人的意志是让鸟活，这只鸟就是活的；年轻人的意志是让鸟死，则这只鸟就是死的。

如果到此止步，则这个答案存在着巨大的漏洞：年轻人可以将鸟掐死，但坚持说自己的意志是让鸟活着，即自己不小心将鸟掐死了，而自己的意志却是想让鸟活着。这样的话，智者的回答就错了。

妙不可言的是，will 在英语中还有一层意思：将要，即将要发生的事或即将要采取的行动。就是说，这只鸟的死活与年轻人将要采取的行动保持一致：年轻人将要放飞这只鸟，这只鸟就是活的；年轻人将要掐死这只鸟，这只鸟就是死的。

因此，智者的回答将人的意志与行为全部包括进来。这样，即使年轻人声称他的意愿和行

为并不一致，智者的回答也正确无误。

好，我们知道了智者的回答。但这与操作系统课程有什么关系吗？

有！很多人都觉得操作系统枯燥、乏味，甚至令人厌烦。更有人说懂不懂操作系统没有关系。不是很多人在学习操作系统之前就已经写过程序了吗？有的人甚至已经写过很大很复杂的程序了。可见，不懂操作系统并不妨碍我们学习使用计算机。

如果读者这样想，我劝你再想一想。你虽然写过程序，可你知道程序到底是如何在计算机上运行的吗？如果不知道，你怎么敢肯定你的程序总是会运行正确呢？你怎么敢说你写的程序最大限度地利用了系统的能力了呢？一个人觉得操作系统没用，那是因为他不知道怎么用，或者他没有用操作系统的意愿。说明白一点，你如果认为操作系统没有用，那是因为你的编程和程序开发处在一个低级的水平上。如果你掌握了操作系统，你的编程水平将显著提高。

换句话说，操作系统有没有用，我的回答是“*As you will*”。你如果有意愿或者有行动使用操作系统，操作系统就是有用的；如果你没有意愿或行动，则操作系统就是没有用的。当然了，我希望读者在看完这本书后能够领悟到操作系统的巨大用途。万一在读完本书后，读者还是困惑或者觉得没用，我唯一能说的也是“*As you will*”。当然，我希望这种情况发生的概率不大。

## 1.1 人造学科

要想学好操作系统，具有恰当的思维模式是十分必要的。这个思维模式就是本书所强调的“哲学”：一种思维方式或一种生活方式。我们以一个问题来说明这一点。这个问题是：什么是计算机的根本特征？

对于这个问题，相信很多人会说计算机就是个计算机器，或者是用来进行大规模计算的机器，或者是用于数据处理的机器，或者是具备某些其他具体功能的机器。这些回答当然没有错，问题是这些答案并不能帮助我们更好地学习理解计算机。就像我们问“张三这个人的根本特征是什么”，而回答是“张三有175cm高”一样。这种答案虽然是正确的，但意义不大，因为我们无法从答案中推导出一系列有用的结论。

那这个问题该如何回答呢？这就要看我们对事物的观察程度。如果我们仔细看看身边的事物，就会发现所有的东西可以划分为两类：一类是本来就存在于自然中，人类所做的只不过是发现；另一类是本来并不存在，人类所做的是发明。第一类事物我们称之为神造事物或者自然存在的事物，第二类事物当然就是人造事物。从这个思维模式上看，计算机毫无疑问就是人造事物，这正是我们所需要的答案，即计算机的根本特征是“人造”。

引申一下可知，计算机学科就是一个人造学科。那么知道计算机学科是人造学科对我们学习计算机有什么帮助呢？有，太多了。下面我们来看看人造和神造有什么区别。

人造学科四个特点：

- 不精确、具有相对性。
- 从对人类活动的观察导出。
- 依赖于人的主观判断力。
- 通常符合人的直觉。



第一个特点就是所谓的“没有对错”。在人造的学科里，没有什么绝对的对或者错，而只有所谓的“好”或者“坏”，“有意义”或者“没意义”。例如，如果本书在讨论计算机时某个方面的论述与你见到的计算机不一样，这不说明本书错了。就算世界上没有本书所论述的计算机存在，也不说明本书错了。我们只需要按照本书的论述再造一台计算机即可。但是，本书论述的计算机与你知道的计算机之间可以进行好和坏的比较。

第二个特点说的是人造学科是从什么得到灵感的，那就是“对人类活动的观察”。这样，读者只要对人类生活仔细观察，就可以很容易地理解计算机里面的许多原理。例如，在操作系统中，广泛使用的栈和队列就是从生活中观察到的现象进行抽象所获得，如图 1-1 所示。

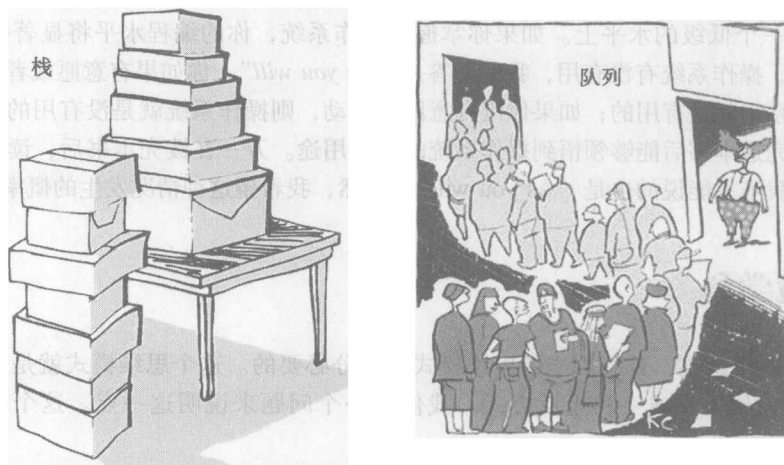


图 1-1 从观察人类活动而获得的栈和队列结构

第三个特点说的是在人造学科里，人的主观能动性起着关键的作用。不同的人观察同样的现象，得出的结论或抽象出的东西可能不一样，甚至完全相反。这样，多数人所认同的抽象就将成为人造学科里的标准，即存在少数服从多数的原则。

第四个特点说的是人造学科里面的许多原理与人的直觉直接呼应，即如果我们按照人的直觉去理解这些原理，就会十分直截了当。例如，操作系统里面的同步机制与人类男女谈恋爱时所用的约会机制十分相似。对于一个谈过恋爱或与别人约会过的人来说，如果将自己谈恋爱的直觉用在操作系统进程的学习上，就会发现进程同步是个十分容易理解的概念。

相对于人造学科，神造学科刚好具有相反四个属性：

- 精确、绝对。
- 从对自然存在的观察导出。
- 不依赖于人的主观判断力。
- 通常违反人的直觉。

第一个特点说的是神造的事物具有精确、绝对的属性。对于这种学科，存在正确与错误之分，我们提出的观点要么正确，要么错误，不存在中间状态。例如，纯数学领域的各种运算，如  $2^2$  的结果应该是 4。如果运算的结果不是 4，则属于运算错误。