

高等学校计算机专业核心课

名师精品·系列教材

计算机 操作系统 习题与考研真题解析

姜华 赵阳 王艳秋 主编

汤小丹 主审

ANALYSIS OF EXERCISES AND
POSTGRAD ENTRANCE EXAM FOR
COMPUTER OPERATING SYSTEM

汤小丹《计算机操作系统（慕课版）》配套教材
详解课后习题、考研真题与典型考题，题量充足，题型丰富
提供多套考试模拟卷，配套录制复习指导微课视频



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

高等学校计算机专业核心课

名师精品·系列教材

计算机 操作系统

习题与考研真题解析

姜华 赵阳 王艳秋 主编

汤小丹 主审

ANALYSIS OF EXERCISES AND
POSTGRAD ENTRANCE EXAM FOR
COMPUTER OPERATING SYSTEM

人民邮电出版社

北京

内 容 提 要

本书是《计算机系统（慕课版）》一书配套的习题与考研真题解析教材，旨在帮助读者熟悉基础知识，巩固理论所学，逐步掌握操作系统的基本原理与核心知识点。

全书共 4 部分。第 1 部分课后习题参考答案，给出了《计算机系统（慕课版）》各章课后习题的参考答案，以及详细的解题思路与步骤。第 2 部分拓展习题及其参考答案与解析，通过整理各章的知识导图，筛选历年的考研真题与操作系统典型习题，给出了选择题、填空题、简答题、综合应用题等多种习题，并配以相应的参考答案与解析，以供读者进行拓展练习。第 3 部分考研真题卷及其参考答案，给出了全国考研真题卷（多套）以及相应的参考答案与解析，旨在帮助读者综合认知统考真题，并自行对课程学习效果进行严格检验。第 4 部分考试模拟卷及其评分标准，给出了操作系统课程的期末考试模拟卷（多套）以及参考答案与评分标准，它们可作为院校教师出卷的重要参考。

本书可作为计算机专业以及其他相关专业操作系统课程的补充教材，也可供操作系统爱好者或者准备参加操作系统相关考试的学生参考使用。

《计算机操作系统（慕课版）》一书是汤子瀛主编的《计算机操作系统》经典教材的最新版。为了使该经典教材的配套教材更加完善，编者根据多年的教学实践经验，结合操作系统教学大纲与研究生入学考试要求，在参考国内外多种操作系统相关资料的基础上，编写了本书，希望本书在巩固读者理论知识方面起到积极作用。

编者在编写本书的过程中，紧密结合主教材以及操作系统考研大纲，以知识点归纳为引导，以求解方法阐述为延伸，以提高实战能力为目标，精选考研真题和典型习题，并在本书中给出了各类习题详细的解题思路与步骤，以帮助读者深入理解操作系统课程的基本内容，实现讲练结合的学习效果。

本书共含4部分内容，各部分内容及其作用介绍如下。

本书内容及其作用

| 序号 | 内容 | 作用 |
|------|---------------|---------------------------------------|
| 第1部分 | 课后习题参考答案 | 帮助学生进行自我检验；方便教师布置作业，并对学生的学习效果进行检测 |
| 第2部分 | 拓展习题及其参考答案与解析 | 帮助读者拓展学习思路，熟悉知识体系，深刻理解操作系统的核心原理 |
| 第3部分 | 考研真题卷及其参考答案 | 帮助读者宏观掌握操作系统考研命题的规律，并深入了解操作系统课程的重难知识点 |
| 第4部分 | 考试模拟卷及其评分标准 | 辅助学生在理论课程学习结束后进行自测，同时为院校教师出卷提供参考 |

本书特色介绍如下。

1. 课后思考 + 拓展习题，题型多样，题量充足

本书根据主教材的章节编排顺序与理论知识的巩固练习需求，配以历年考研真题（含全国统考真题和名校考研真题）和典型考题，题量充足，题型丰富（含选择题、填空题、简答题、计算机、综合应用题等），可以满足不同目标、不同层次的读者需求。同时，充足的习题按主教材章节编排顺序与理论知识点进行清晰分类，可以帮助读者全面且深入地掌握操作系统的知识要点。

2. 知识导图 + 答案解析, 逻辑清晰, 有的放矢

根据主教材知识体系结构和操作系统考研大纲, 编者总结并整理了逻辑清晰的知识导图, 以帮助读者直观理解、快速掌握各章的知识要点, 进而进行针对性的拓展练习、归纳总结、查缺补漏。本书所提供的各类习题大都配有参考答案与解析, 并且通过图表、分步解答等形式对重难点知识进行了解析, 以帮助读者拓展思路, 实现举一反三的学习效果。

3. 考研真题 + 模拟试卷, 资源丰富, 紧扣要点

为了使读者综合认知操作系统考研命题规律, 深入了解考试重难点, 本书编排了多套考研真题卷和考试模拟卷, 并且面向读者提供多套电子版考研真题卷, 全方位地帮助读者对自己的学习效果进行综合检验, 进而实现多知识点的融会贯通。同时, 本书还配套了复习指导微课视频以及电子版试题库, 题量充足、题型丰富, 可满足院校教师针对不同考查目标的出卷需求。

本书第1部分和第2部分按主教材章节进行划分, 各含12章, 其中第1、2、3、10章由赵阳编写, 第4、5、6、12章由王艳秋编写, 第7、8、9、11章由姜华编写; 第3部分和第4部分由姜华、赵阳、王艳秋共同整理; 全书由姜华统稿。

由衷感谢汤小丹老师对本书全稿所进行的细致审阅与定稿把关, 感谢在本书各次线上线下评审会中提出宝贵修改建议的专家学者与院校老师, 感谢张童与黄明在资料收集、书稿整理等方面所付出的辛勤劳动。用书教师可以通过人邮教育社区(www.ryjiaoyu.com)下载本书配套的各类教辅资源。

限于编者水平, 书中难免存在不妥之处, 敬请同行和广大读者批评指正。

编者

2021年春于聊城

第1部分 课后习题参考答案

| | |
|---------------------|-----------|
| 第1章 操作系统引论 | 1 |
| 1.1 简答题参考答案 | 1 |
| 1.2 计算题参考答案 | 6 |
| 1.3 综合应用题参考答案 | 7 |
| 第2章 进程的描述与控制 | 9 |
| 2.1 简答题参考答案 | 9 |
| 2.2 综合应用题参考答案 | 13 |
| 第3章 处理机调度与死锁 | 14 |
| 3.1 简答题参考答案 | 14 |
| 3.2 计算题参考答案 | 17 |
| 3.3 综合应用题参考答案 | 20 |
| 第4章 进程同步 | 22 |
| 4.1 简答题参考答案 | 22 |
| 4.2 计算题参考答案 | 25 |
| 4.3 综合应用题参考答案 | 25 |
| 第5章 存储器管理 | 30 |
| 5.1 简答题参考答案 | 30 |
| 5.2 计算题参考答案 | 32 |
| 5.3 综合应用题参考答案 | 34 |
| 第6章 虚拟存储器 | 37 |
| 6.1 简答题参考答案 | 37 |
| 6.2 计算题参考答案 | 39 |

| | |
|---------------|----|
| 6.3 综合应用题参考答案 | 41 |
|---------------|----|

第7章 输入/输出系统 43

| | |
|---------------|----|
| 7.1 简答题参考答案 | 43 |
| 7.2 计算题参考答案 | 46 |
| 7.3 综合应用题参考答案 | 47 |

第8章 文件管理 49

| | |
|---------------|----|
| 8.1 简答题参考答案 | 49 |
| 8.2 计算题参考答案 | 53 |
| 8.3 综合应用题参考答案 | 54 |

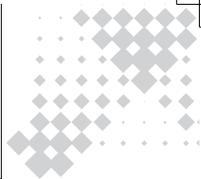
第9章 磁盘存储器管理 55

| | |
|---------------|----|
| 9.1 简答题参考答案 | 55 |
| 9.2 计算题参考答案 | 58 |
| 9.3 综合应用题参考答案 | 61 |

第10章 多处理机操作系统 63**第11章 虚拟化和云计算 71****第12章 保护和安全的 77****第2部分 拓展习题及其参考
答案与解析**

| | |
|-------------------|-----------|
| 第1章 操作系统引论 | 85 |
| 1.1 知识导图 | 85 |
| 1.2 拓展习题 | 86 |
| 1.3 参考答案与解析 | 90 |

| | | | |
|---------------------|------------|----------------------|------------|
| 第2章 进程的描述与控制 | 95 | 7.2 拓展习题 | 163 |
| 2.1 知识导图 | 95 | 7.3 参考答案与解析 | 168 |
| 2.2 拓展习题 | 96 | 第8章 文件管理 | 174 |
| 2.3 参考答案与解析 | 100 | 8.1 知识导图 | 174 |
| 第3章 处理机调度与死锁 | 105 | 8.2 拓展习题 | 175 |
| 3.1 知识导图 | 105 | 8.3 参考答案与解析 | 178 |
| 3.2 拓展习题 | 107 | 第9章 磁盘存储器管理 | 183 |
| 3.3 参考答案与解析 | 113 | 9.1 知识导图 | 183 |
| 第4章 进程同步 | 122 | 9.2 拓展习题 | 184 |
| 4.1 知识导图 | 122 | 9.3 参考答案与解析 | 189 |
| 4.2 拓展习题 | 123 | 第10章 多处理机操作系统 | 195 |
| 4.3 参考答案与解析 | 129 | 10.1 知识导图 | 195 |
| 第5章 存储器管理 | 138 | 10.2 拓展习题 | 196 |
| 5.1 知识导图 | 139 | 10.3 参考答案与解析 | 198 |
| 5.2 拓展习题 | 140 | 第11章 虚拟化和云计算 | 200 |
| 5.3 参考答案与解析 | 145 | 11.1 知识导图 | 200 |
| 第6章 虚拟存储器 | 150 | 11.2 拓展习题 | 201 |
| 6.1 知识导图 | 150 | 11.3 参考答案与解析 | 202 |
| 6.2 拓展习题 | 151 | 第12章 保护和安全的 | 204 |
| 6.3 参考答案与解析 | 155 | 12.1 知识导图 | 205 |
| 第7章 输入/输出系统 | 161 | 12.2 拓展习题 | 206 |
| 7.1 知识导图 | 162 | 12.3 参考答案与解析 | 207 |

**第3部分 考研真题卷及其
参考答案**

| | |
|----------------------|-----|
| 考研真题卷(一) | 210 |
| 考研真题卷(二) | 212 |
| 考研真题卷(一) 参考答案 与解析 | 214 |
| 考研真题卷(二) 参考答案 与解析 | 215 |

**第4部分 考试模拟卷及其
评分标准**

| | |
|------------------------|-----|
| 考试模拟卷(一) | 217 |
| 考试模拟卷(二) | 220 |
| 考试模拟卷(一) 参考答案 与评分标准 | 222 |
| 考试模拟卷(二) 参考答案 与评分标准 | 227 |

人民邮电出版社

第1部分

课后习题参考答案

1

第1章 操作系统引论

1.1 简答题参考答案

1. 在计算机系统上配置 OS (operating system, 操作系统) 的目标是什么? 作用主要表现在哪几个方面?

【参考答案】在计算机系统上配置OS, 主要目标是实现: 方便性、有效性、可扩充性和开放性。

OS的作用主要表现在以下3个方面: ①OS作为用户与计算机硬件系统之间的接口; ②OS作为计算机系统资源的管理者; ③OS实现对计算机资源的抽象。

2. 试说明 OS 与硬件、其他系统软件以及用户之间的关系。

【参考答案】OS是覆盖在硬件上的第一层软件, 它负责管理计算机的硬件和软件资源, 并向用户提供良好的界面。OS与硬件紧密相关, 它直接管理着硬件资源, 为用户完成所有与硬件相关的操作, 极大地方便了用户对硬件资源的使用并提高了硬件资源的利用率。OS是一种特殊的系统软件, 也是其他系统软件与硬件之间的接口。其他系统软件运行在OS的基础之上, 可获得OS提供的大量服务。而一般用户使用计算机除了需要OS支持外, 还需要用到大量的其他系统软件和应用软件, 以使其工作更高效。可见, 在OS、硬件、其他系统软件以及用户之间存在着图1-1-1所示的层次关系。

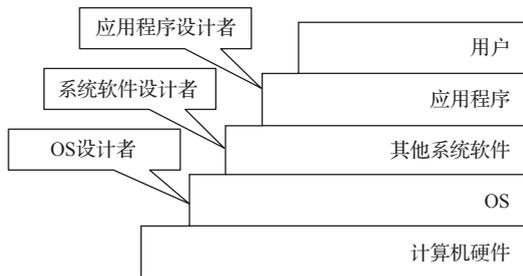


图 1-1-1 计算机系统的层次关系

3. 试说明推动 OS 发展的主要动力是什么。

【参考答案】推动OS发展的主要动力表现在：①计算机系统资源的利用率不断提高；②方便用户；③器件不断更新换代；④计算机体系结构不断发展；⑤新的应用需求不断被提出。

4. 在 OS 中，何谓脱机 I/O (input/output, 输入 / 输出) 方式和联机 I/O 方式？

【参考答案】脱机I/O方式是指事先将装有用户程序或数据的纸带或卡片装入纸带输入机或卡片机，在外围机的控制下，把纸带或卡片上的数据或程序输入到磁带上，该方式下的I/O操作由外围机控制完成，并且是在脱离主机的情况下进行的，因此叫作脱机I/O方式。联机I/O方式是指程序和数据的I/O都是在主机的直接控制下进行的。

5. 试说明推动分时系统形成和发展的主要动力是什么。

【参考答案】推动分时系统形成和发展的主要动力是“为了更好地满足用户的需要”。其主要表现在：①CPU (central processing unit, 中央处理机) 的分时使用缩短了作业的平均周转时间；②人机交互功能的提供使用户能方便地直接控制自己的作业；③主机的共享使多个用户 (包括远程用户) 能同时使用一台计算机独立地、互不干扰地处理各自的作业。

6. 实现分时系统的关键问题是什么？应如何解决？

【参考答案】实现分时系统的关键问题是使用户能与自己的作业进行交互，即用户在自己的终端上输入一条命令以请求系统服务后，系统能及时地接收并处理该命令，并在用户能接受的时延内将结果返回给用户。

及时地接收命令和返回输出结果的实现方式是在系统中配置一个多路卡，并为每个终端配置一个缓冲区以暂存用户输入的命令和输出的结果。因此，关键要解决的问题是确保在较短的时间内系统中所有的用户程序都能执行一次，从而使用户输入的命令能够得到及时响应。为此，一方面，用户作业被提交后应立即进入内存；另一方面，系统应设置一个被称为时间片的很短的时间，并规定每个程序每次最长只能连续运行一个时间片，如果时间片用完，则不管它是否运行完毕，都必须将CPU让给下一个作业。通过使作业分时共享CPU，所有的作业都可以得到及时的处理，用户的请求亦可得到及时的响应。

7. 为什么要引入实时系统？

【参考答案】实时系统是指系统能及时响应外部事件的请求，在规定的时间内完成对外部事件的处理，并控制所有实时任务协调一致地运行。引入实时系统是为了更好地满足实时控制领域和实时信息处理领域的需要。

8. 什么是硬实时任务和软实时任务？试举例说明。

【参考答案】①硬实时任务是指系统必须满足任务对截止时间的要求，否则可能出现难以

预测的结果，如运载火箭的控制等。②软实时任务对截止时间的要求并不严格，偶尔错过任务的截止时间，对系统产生的影响也不大，如网页内容的更新、火车售票系统剩余票数的更新等。

9. 试从及时性、交互性及可靠性方面对分时系统与实时系统进行比较。

【参考答案】本题分步解答如下。

(1) 从交互性方面来考虑。交互性问题是分时系统的关键问题。在分时系统中，用户可以通过终端与系统进行广泛的人机交互，如文件编辑、数据处理和资源共享等。实时系统也具有交互性，但在实时系统中，交互性仅限于访问系统中某些特定的专用服务程序，也就是说，它的交互性具有很大的局限性。

(2) 从及时性方面来考虑。分时系统的及时性是指用户能在很短的时间间隔内获得系统的响应，这一时间间隔是根据人们能接受的等待时间来确定的，一般为2~3s。对实时系统来说，及时性是它的关键问题之一，实时信息系统的及时性要求与分时系统相似，而实时控制系统的及时性要求则是由被控制对象所要求的开始截止时间和完成截止时间决定的，一般为秒级、百毫秒级、毫秒级，甚至更低。

(3) 从可靠性方面来考虑。可靠性是实时系统的另一个关键问题，实时系统中的任何差错都可能带来巨大的经济损失，甚至导致无法预料的灾难性后果，因此实时系统往往采取多级容错措施来保证系统高度可靠。分时系统虽然也要求可靠，但比实时系统的要求要低。

10. 微机 OS 按运行方式来分，可以分为哪几类？举例说明。

【参考答案】现在流行的微机OS按运行方式可以分为以下3类。

(1) 单用户单任务OS：只允许一个用户上机（使用计算机），且只允许用户程序作为一个任务运行，主要配置在8位和16位微机上。最具代表性的单用户单任务OS是CP/M（8位）和MS-DOS（16位）。

(2) 单用户多任务OS：只允许一个用户上机，但允许用户把程序分为若干个任务并发执行，从而有效地改善了系统性能。最具代表性的单用户多任务OS是由微软公司推出的Windows系列，如Windows 3.1、Windows 95、Windows 98等。

(3) 多用户多任务OS：允许多个用户通过各自的终端使用一台机器，共享主机系统中的各种资源，而每个用户程序又可进一步分为若干个任务并发执行，从而可以进一步提高资源利用率和系统吞吐量。最具代表性的多用户多任务OS是UNIX系统，各种类UNIX系统（如Solaris、Linux系统等）以及Windows NT/Server系列的系统。

11. OS 具有哪几大特征？它们之间有何关系？

【参考答案】OS具有并发、共享、虚拟和异步这4个基本特征。它们之间的关系包含以下几个方面。①并发和共享是OS最基本的特征。为了提高计算机资源的利用率，OS必然要采用多道程序设计技术，使多个程序共享系统的资源、并发地执行。②并发性和共享性互为存在的条件。一方面，资源的共享是以程序（进程）的并发执行为条件的，若系统不允许程序并发执行，就不会存在资源共享问题；另一方面，若系统不能对资源共享实施有效管理，协调好各进程对共享资源的访问，则必将影响程序的并发执行，甚至会使程序无法并发执行。③虚拟性以并发性和共享性为前提。为了使并发进程能更方便、更有效地共享资源，OS常采用多种虚拟技术在逻辑上增加CPU和设备的数量以及存储器的容量，从而解决并发进程对有限系统资源的共享问题。④异步性是并发性和共享性的必然结果。OS允许多个并发进程共享资源、相互合作，使得每个进程的运行过程受到了其他进程的制约，不再“一气呵成”，这必然会导致异步这一特征的产生。

12. 是什么原因使 OS 具有异步特征?

【参考答案】在多道程序环境下,允许多个进程并发执行。但由于资源等因素的限制,进程的执行通常并非“一气呵成”,而是以“走走停停”的方式进行。内存中的每个进程在何时执行、何时暂停、以怎样的速度向前推进,都是不可预知的。故而作业完成的先后次序与进入内存的先后次序并不完全一致,亦即进程是以异步方式运行的。但在有关进程控制及同步机制等的支持下,只要运行环境相同,作业经多次运行都会获得完全相同的结果,因而异步方式是允许的。因此,OS具有异步特征。

13. 何谓 OS 内核? OS 内核的主要功能是什么?

【参考答案】在现代OS理论中,一般将OS划分为若干层次,再将OS的不同功能分别设置在不同层次中。通常将一些与硬件紧密相关的模块(如中断处理程序等)、各种常用设备的驱动程序以及运行频率较高的模块(如时钟管理模块、进程调度模块和公用基本操作模块等)都安排在紧靠硬件的软件层次中,让它们常驻内存,进而形成了所谓的OS内核。

OS内核的主要功能有:①支撑功能,包括中断处理、时钟管理和原语操作等;②资源管理功能,包括进程管理、存储器管理、设备管理等。

14. 何谓原语?何谓原子操作?

【参考答案】原语,是指由若干条指令组成的,用于完成一定功能的一个过程。原子操作,是指一个操作中的所有动作要么全做,要么全不做,换言之,它是一个不可分割的基本单位。因此,原语在执行过程中是不允许被中断的。原子操作在内核态下执行,常驻内存。

15. 简要描述处理机的双重工作模式。

【参考答案】一般地,处理机至少需要两种单独运行模式:用户态和内核态。

(1) 用户态也称为目态,计算机硬件可以通过一个模式位为1来表示它。当计算机系统执行用户程序时,系统处于用户态。

(2) 内核态也称为管态或系统态,计算机硬件可以通过一个模式位为0来表示它。每当OS能够控制计算机时,它就处于内核态。例如,当用户程序通过系统调用请求OS服务时,系统必须从用户态切换到内核态,以响应请求。

16. 简述中断处理过程。

【参考答案】一旦CPU响应中断,系统就会开始进行中断处理。中断处理过程主要包括以下3步。

(1) 保护被中断进程的现场。为了在中断处理结束后能使进程正确地返回中断点,系统必须保存当前处理机状态字和程序计数器的值。

(2) 分析中断原因,转去执行相应的中断处理程序。在多个中断请求同时发生时,处理优先级最高的中断源所发出的中断请求。

(3) 恢复被中断进程的现场,CPU继续执行被中断的原进程。

17. 处理机管理有哪些主要功能?它们的主要任务是什么?

【参考答案】处理机管理的主要功能有进程管理、进程同步、进程通信和处理机调度。各部分的主要任务为:①进程管理,为作业创建进程,撤销已结束进程,控制进程在运行过程中的状态转换;②进程同步,协调多个进程(含线程)的运行;③进程通信,实现在相互合作的进程之间进行信息交换;④处理机调度,包括作业调度和进程调度。作业调度是按照一定

的算法从后备队列里选出若干个作业，为它们分配运行所需的资源；进程调度是按照一定的算法从进程的就绪队列中选出一个进程，把处理机分配给它，并设置运行现场，使进程投入执行。

18. 内存管理有哪些主要功能？它们的主要任务是什么？

【参考答案】内存管理的主要功能有内存分配、内存保护、地址映射和内存扩充。各部分的主要任务为：①内存分配，为每道程序分配内存；②内存保护，确保每道程序仅在自己的内存空间运行，彼此互不干扰；③地址映射，将地址空间的逻辑地址转换为内存空间对应的物理地址；④内存扩充，即从逻辑上扩充内存，实现请求调入功能和置换功能等。

19. 设备管理有哪些主要功能？它们的主要任务是什么？

【参考答案】设备管理的主要功能包括缓冲管理、设备分配、设备处理以及设备虚拟等。各部分的主要任务为：①完成用户提出的I/O请求，为用户进程分配所需的I/O设备，并完成指定的I/O操作；②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。

20. 文件管理有哪些主要功能？它们的主要任务是什么？

【参考答案】文件管理的主要功能有：文件存储空间的管理、目录管理、文件的读/写管理和保护。其主要任务是对用户文件和系统文件进行管理以方便用户使用，并保证文件安全。

21. 现代 OS 的新功能有哪些？

【参考答案】现代OS是在传统OS的基础上发展而来的，它除了具有传统OS的功能外，还具有保障系统安全、支持用户通过联网获取服务和可处理多媒体信息等功能。

22. 什么是微内核 OS？它具有哪些优点？

【参考答案】微内核OS具有如下4个特征：①足够小的内核；②基于客户/服务器模式；③应用策略与机制分离原则；④采用面向对象技术。

微内核OS的主要优点包括：①提高了系统的可扩展性；②增强了系统的可靠性和可移植性；③提供了对分布式系统的支持；④融入了面向对象技术。

23. 外核 OS 的基本思想是什么？

【参考答案】外核OS的基本思想是：内核不提供传统OS中的进程、虚拟存储等抽象事物，而是专注于物理资源的隔离（保护）与复用。具体来说，在基于外核结构的OS中，内核负责保护系统资源，而硬件资源的管理职责则委托给了应用程序，这样，OS就可以做到在保证资源安全的前提下，减少对应用程序的限制，充分满足应用程序对硬件资源的不同需求。

24. 什么是系统调用？系统调用与一般用户程序和库函数有何区别？

【参考答案】系统调用是OS提供给程序员的唯一接口。程序员利用系统调用，在源程序层面动态请求和释放系统资源，并调用系统中已有的系统功能来完成那些与机器硬件部分相关的工作以及控制程序的执行速度等。因此，系统调用像一个“黑箱子”，对用户屏蔽了OS的具体动作而只提供有关的功能。

系统调用与一般用户程序、库函数的区别在于：①系统调用（程序）在内核态执行，调用它们时需要一个类似于硬件中断处理机制的中断处理机制来提供系统服务；②普通的用户程序是直接为用户完成某特定功能而设计的，它们一般在用户态执行；③库函数是把函数放到库里供别人使用的一种方式，是面向应用开发、方便人们编程的。

1.2 计算题参考答案

25. 设有3道程序A、B、C，它们按照优先次序(A→B→C)顺序执行，它们的计算时间和I/O操作时间如表1-1-1所示，假设3道程序以串行方式使用相同的设备进行I/O操作，试画出单道程序运行和多道程序运行的时间关系图，并计算完成这3道程序所须花费的时间。

表1-1-1 时间表

| 程序 | 时间 (ms) | | |
|----|---------|-------|----|
| | 计算 | I/O操作 | 计算 |
| A | 30 | 40 | 10 |
| B | 60 | 30 | 10 |
| C | 20 | 40 | 20 |

【参考答案】计算时间时，CPU处理时间，I/O操作时间是输入与输出时间。3个程序共享I/O设备，即串行使用I/O设备。

(1) 单道程序运行时，3道程序串行执行，即先执行A、再执行B、最后执行C，时间关系图如图1-1-2所示。

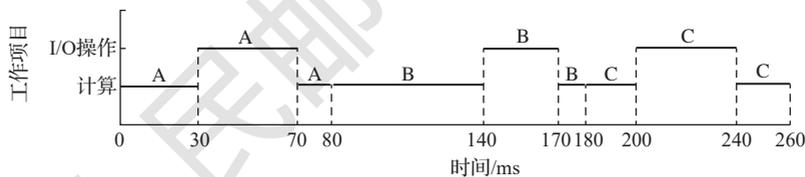


图 1-1-2 单道程序运行时间关系图

单道程序运行时，3道程序使用的时间为： $(30+40+10) + (60+30+10) + (20+40+20) = 260\text{ms}$ 。

(2) 多道程序运行时，3道程序的计算与I/O操作可部分并行，分为非立即抢占式和立即抢占式两种，时间关系图如图1-1-3和图1-1-4所示。

多道程序运行时，3道程序非立即抢占式的总时间为180ms，立即抢占式的总时间为190ms。

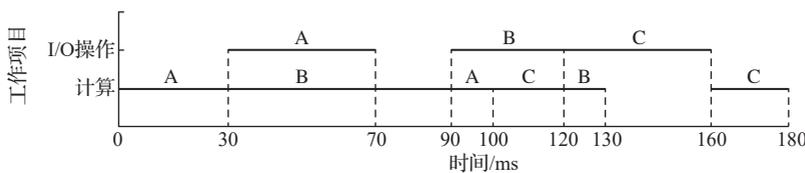


图 1-1-3 多道程序非立即抢占式运行时间关系图

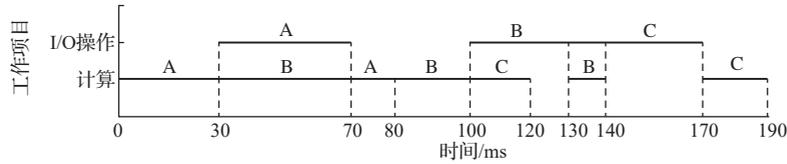


图 1-1-4 多道程序立即抢占式运行时间关系图

26. (考研真题) 一个多道批处理系统中仅有 P_1 和 P_2 两个作业, P_2 比 P_1 晚 5ms 到达, 它们的计算和 I/O 操作顺序如下。

P_1 : 计算 60ms, I/O 操作 80ms, 计算 20ms。

P_2 : 计算 120ms, I/O 操作 40ms, 计算 40ms。

不考虑调度和切换时间, 请计算完成两个作业需要的最少时间。

【参考答案】作业执行时间关系图如图1-1-5所示。由于在多道批处理系统中, P_1 与 P_2 可以部分并行, 因此, P_1 先到达系统, 其会先占用CPU进行计算(到60ms), 然后执行I/O操作的时间是60ms~140ms; 而在 P_1 执行I/O操作的过程中, P_2 可获得CPU运行120ms, 到180ms结束; 当 P_1 执行完它的I/O操作后, 执行计算, 此时CPU正被 P_2 占用, 因此 P_1 须等 P_2 执行完后才能获得CPU执行剩余的20ms, 执行完成后退出系统; 此时, P_2 会执行I/O操作40ms, 到220ms; 最后 P_2 获得CPU运行剩余的40ms(计算), 到260ms结束。由图1-1-5可知, 完成两个作业需要的最少时间为260ms。

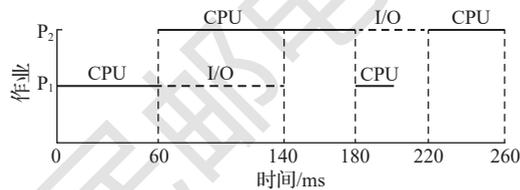


图 1-1-5 作业执行时间关系图

1.3 综合应用题参考答案

27. OS 的概念、特征和功能是什么?

【参考答案】该问题分步解答如下。

(1) OS是一组控制和管理计算机硬件和软件资源、合理地对各类作业进行调度以方便用户使用计算机的程序集合。OS是配置在计算机硬件上的第一层系统软件, 是对硬件系统的首次扩充; 是硬件系统和应用软件间的桥梁; 是用户与计算机硬件进行交互的接口; 是计算机系统资源的管理者。

(2) OS的4个特征: 并发、共享、虚拟、异步。①并发: 一段时间间隔内多个进程(线程)并发执行, 是宏观上的并行, 微观上的串行。②共享: 系统中的资源可供内存中多个并发执行的进程或线程共同使用。③虚拟: 通过某种技术将物理实体变为若干个逻辑上的对应物。④异步: 进程以人们不可预知的速度向前推进, 每次运行只要环境相同, 则结果必定一致。

(3) OS的5大功能: 处理机管理、存储器管理、设备管理、文件管理、接口管理。①处理

机管理：进程（线程）是处理机调度的单位，因此对外理器的管理实际上是对进程（线程）的管理。②存储器管理：内存的分配与回收、地址转换、虚拟内存的实现等。③设备管理：设备的分配与回收、缓冲区管理、磁盘调度、设备虚拟等。④文件管理：文件存储空间的管理、文件目录管理、文件共享与保护等。⑤接口管理：用户接口、程序接口、命令接口和网络接口。

28. (考研真题) 若某计算问题的执行情况如图 1-1-6 所示。

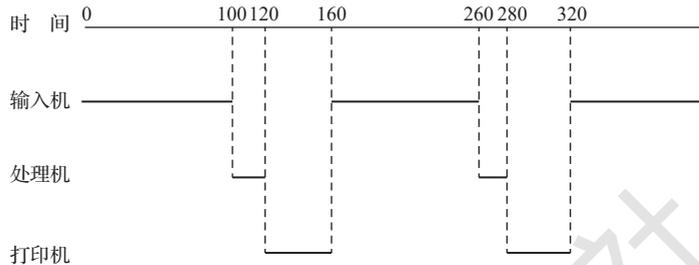


图 1-1-6 计算问题执行情况

则请回答下列问题。

- (1) 叙述该计算问题中处理机、输入机和打印机是如何协同工作的。
- (2) 计算在图 1-1-6 所示执行情况下处理机的利用率。
- (3) 简述处理机利用率不高的原因。
- (4) 请画出能提高处理机利用率的执行方案。

【参考答案】 (1) 处理机、输入机和打印机是按照输入→处理→打印的顺序依次执行的，输入机为处理机提供数据，处理机得到数据后进行计算，处理结果通过打印机打印输入。输入机读取一批数据，花费时间为 100；处理机对这批数据进行计算，花费时间为 20；打印机打印计算结果，花费时间为 40。

(2) 处理机的利用率 = $[20 / (100 + 20 + 40)] \times 100\% = 12.5\%$ 。

(3) 当一道程序在运行中发出 I/O 请求后，处理机只能处于等待状态，即必须等 I/O 完成后才能继续运行，因此处理机会长时间处于空闲状态，这会导致其利用率不高。

(4) 采用多道程序设计技术使处理机、输入机和打印机并行工作，可以提高处理机的利用率，如图 1-1-7 所示。

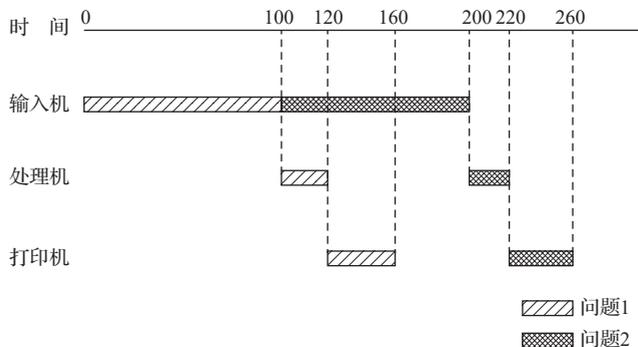


图 1-1-7 并行工作时计算问题执行情况

第2章 进程的描述与控制

2.1 简答题参考答案

1. 什么是前趋图？请画出下列4条语句的前趋图。

$S_1: a=x+y; S_2: b=z+1; S_3: c=a-b; S_4: w=c+1;$

【参考答案】本题分步解答如下。

(1) 前趋图 (precedence graph) 是一个有向无环图, 记为 DAG (directed acyclic graph), 用于描述进程间执行的前后关系。

(2) 题中4条语句对应的前趋图如图1-2-1所示。

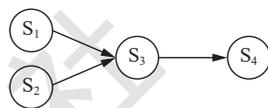


图 1-2-1 前趋图

2. 什么是进程？OS 中为什么要引入进程？它会产生什么样的影响？

【参考答案】①进程是一段可并发执行的具有独立功能的程序, 是关于某个数据集的一次执行过程, 也是OS进行资源分配和保护的基本单位。②在OS中引入进程, 是为了实现多个程序的并发执行。传统的程序与其他程序并发执行时, 执行结果不可再现, 因此, 传统的程序不能与其他程序并发执行, 只有在为之创建进程后, 其才能与其他程序 (进程) 并发执行。这是因为并发执行的程序“停停走走”地执行, 只有在为它创建进程后, 在它停下时, 方能将其CPU现场信息保存在它的PCB (processing control block, 进程控制块) 中, 待下次被调度执行时再从PCB中恢复CPU现场而继续执行, 但传统的程序却无法违反上述要求。③建立进程所带来的好处是多个程序能并发执行, 这极大地提高了资源利用率和系统吞吐量。但管理进程也须付出一定的代价, 包括PCB及协调各运行机构所占用的内存空间开销, 以及为进行进程间的切换、同步与通信等所付出的时间开销。

3. 进程最基本的状态有哪些？哪些事件可能会引起不同状态间的转换？

【参考答案】进程最基本的状态有3种。①运行态：进程占有处理机, 正在运行。②就绪态：进程具备运行条件, 等待系统分配处理机以便运行。③等待态 (又称为阻塞态或睡眠态)：进程不具备运行条件, 正在等待某个事件的完成。

进程不同状态间的转换及引发原因介绍如下。①运行态→等待态：等待使用资源或某事件发生；②等待态→就绪态：资源得到满足或某事件已经发生；③运行态→就绪态：运行时间片到达或出现有更高优先级的进程；④就绪态→运行态：CPU空闲时调度选中一个就绪进程需要其运行。

4. 为什么要引入进程的挂起状态？

【参考答案】所谓挂起状态, 实际上就是一种静止的状态。一个进程被挂起后, 不管它是否处于就绪状态, 系统都不会分配给它处理机。因此, 引入挂起状态是基于系统和用户的如下需要。①终端用户的需要：当终端用户在自己的程序运行期间发现问题时, 希望暂停进程的运行。②父进程请求：父进程挂起自己的某个子进程, 检查并修改该子进程, 或者协调各子进程之间的活动。③负荷调节的需要：当实时系统中的工作负荷较重、实时任务受到影响时, 挂起

一些不重要的进程。④OS的需要：OS挂起某些进程，检查或统计运行中的资源使用情况。

5. 叙述组成进程的基本要素，并说明它们的作用。

【参考答案】每个进程均由PCB、程序块、数据块和堆栈这4个基本要素组成。它们的作用分别描述如下。

(1) PCB：用来存储进程的标志信息、现场信息和控制信息。进程创建时建立PCB，进程撤销时回收PCB，其与进程一一对应。

(2) 程序块：即被执行的程序。它规定了进程一次运行应实现的功能，通常是纯代码，其作为一种系统资源可被多个进程共享。

(3) 数据块：即程序运行时须加工处理的对象，包括全局变量、局部变量和常量等的存放区以及开辟的工作区，常常为一个进程专用。

(4) 堆栈：每个进程都将捆绑一个堆栈，用来存储进程的标志信息、现场信息和控制信息。

6. (考研真题) 请给出 PCB 的主要内容。描述当进程状态发生转换 (就绪→运行、运行→阻塞) 时，OS 需要使用 / 修改 PCB 的哪些内容？

【参考答案】本题分步解答如下。

(1) PCB主要用于描述进程的基本情况以及进程的运行变化过程，是进程存在的唯一标志。PCB的内容可以分成调度信息和现场信息两部分。①调度信息供进程调度时使用，描述了进程当前所处的状况，包括进程名、进程号、存储信息、优先级、当前状态、资源清单、家族关系、消息队列指针、进程队列指针和当前打开文件的情况等。②现场信息用于保留运行进程发生状态转换时所要保存的CPU现场信息，以保证当该进程重新恢复运行时能恢复CPU现场，从断点处继续运行。常用的现场信息包括通用寄存器的内容、控制寄存器的内容、用户堆栈指针、系统堆栈指针等。

(2) OS需要使用/修改的PCB内容包括“就绪→运行”和“运行→就绪”。①就绪→运行是将PCB当前的就绪态改为运行态，修改PCB的队列指针，将PCB从就绪队列中移出等；利用PCB中的CPU现场信息，布置CPU现场并投入运行。②运行→就绪是将CPU的当前状态（各寄存器的内容）保存到PCB中，将进程状态由“运行”改为“就绪”，并修改PCB中相应的队列指针信息，然后将其放入PCB就绪队列中。

7. 试说明引起进程创建的主要事件。

【参考答案】为使程序之间能并发运行，应先为它们分别创建进程。导致一个进程去创建另一个进程的典型事件有4类：用户登录、作业调度、提供服务、应用请求。

8. (考研真题) 在创建一个进程时，OS 需要完成的主要工作是什么？

【参考答案】OS发现请求创建新进程事件后，首先，调用进程创建原语；其次，申请一个空白PCB，并向该PCB中填写用于控制和管理进程的信息；再次，为该进程分配运行时所需的资源；最后，把该进程的PCB转入就绪状态并插入就绪队列中。

9. 试说明引起进程终止的主要事件。

【参考答案】引起进程终止的主要事件介绍如下。①正常结束：指进程的任务已经完成，准备退出运行。②异常结束：指进程在运行时发生了某种异常事件，使程序无法继续运行。常见的异常事件包括越界错误、保护错、指令非法、特权指令错、运行超时、等待超时、算术运算错、I/O故障等。③外界干预：指进程应外界的请求而终止运行，这些干预有操作员干预、OS干预、父进程请求、因父进程终止等。

10. 在终止一个进程时，OS 要完成的主要工作是什么？

【参考答案】如果系统中发生了要求终止进程的事件，OS便会调用进程终止原语，按下述过程终止指定的进程：

- (1) 根据被终止进程标识符，从PCB集中检索出指定进程的PCB，读出该进程的状态；
- (2) 若被终止进程处于执行状态，则立即终止该进程的执行，置调度标志为真，用于指示该进程被终止后应重新进行调度；
- (3) 若该进程还有子孙进程，则应将其所有子孙进程也都予以终止，以防它们成为不可控的进程；
- (4) 将被终止进程拥有的全部资源或者归还给其父进程，或者归还给系统；
- (5) 将被终止进程的PCB从所在队列或链表中移出，等待其他程序来搜集信息。

11. 试说明引起进程阻塞或被唤醒的主要事件。

【参考答案】引起进程阻塞或被唤醒的主要事件介绍如下。①向系统请求共享资源失败。进程在向系统请求共享资源时，由于系统已无足够的资源分配给它，此时进程会因不能继续运行而转变为阻塞状态。②等待完成某种操作。当进程启动某种操作后，如果该进程必须在操作完成之后才能继续执行，则应先阻塞该进程。③尚未到达新数据。对于相互合作的进程，如果一个进程需要获得另一个进程提供的数据后才能处理该数据，但所需数据尚未到达，则进程阻塞。④等待到达新任务。在某些系统中，特别是在网络环境下的OS中，往往会设置一定量的系统进程，这种进程每当完成任务后便会把自己阻塞起来，等待新任务的到来。

12. 试比较进程间的低级与高级通信工具。

【参考答案】用户使用低级通信工具实现进程通信效率低，通信对用户不透明，所有操作都必须由程序员来实现；高级通信工具弥补了这些缺陷，用户可以直接利用OS提供的一组通信命令来高效地传送大量的数据。

13. 当前有哪几种高级通信机制？

【参考答案】高级通信机制可以归结为4类。①共享存储器系统通信机制：在共享存储器系统中，相互通信的进程共享某些数据结构或存储区，进程之间通过这些“空间”进行通信。②管道通信系统通信机制：发送进程和接收进程利用管道进行通信。③消息传递系统通信机制：进程不必借助任何共享存储区或数据结构，而是以格式化的消息（message）为单位，将通信的数据封装在消息中，并利用OS提供的一组通信命令在进程间进行消息传递，完成进程间的数据交换。④客户机-服务器系统通信机制：在网络环境的各种应用领域，客户机-服务器系统的通信机制已成为当前主流的通信实现机制。

14. 说明使用管道文件（pipe 文件）进行通信的优缺点。

【参考答案】在Linux、UNIX和OS/2等系统中都可以使用管道文件实现进程间的相互通信。其优点是可以在进程间不断地传递大量信息，且无须占用宝贵的内存空间。缺点是进程之间的信息传递要启动读/写磁盘，相对于消息缓冲队列而言通信速度较慢。另外，接收进程和发送进程间的同步也比较复杂，需要一定的时间开销。

15. 试比较直接通信方式和间接通信方式。

【参考答案】可以从以下几个方面来比较直接通信方式和间接通信方式。

- (1) 发送原语和接收原语。直接通信原语通常为send（receiver, message）、receive（sender, message）；间接通信原语通常为send（mailbox, message）、receive（mailbox,

message)，而且它还需要提供有关信箱创建与撤销的原语。

(2) 提供对方的标识符。直接通信要求发送双方显式地提供对方的标识符，对于接收进程，如果允许它同时接收多个进程发来的消息，则接收原语中的发送进程标识符可以是通信完成后返回的值；间接通信则不要求它们显式地提供对方的标识符，而只须提供信箱标识符。

(3) 通信链路。直接通信时，进程只须提供对方的标识符便可进行通信，在收发双方之间建立通信链路的任务由系统自动完成，此外，在收发双方之间有且仅有一条通信链路。间接通信时，仅当一对进程共享某个信箱时，它们之间才有通信链路；每对进程间可以有 multiple 链路。

(4) 实时性。直接通信通常只能提供实时的通信；间接通信既可实现实时通信，也可实现非实时通信。

16. 为什么要在 OS 中引入线程？

【参考答案】在OS中引入线程是为了减少进程在并发执行时所付出的时空开销，使OS具有更好的并发性，提高CPU的利用率。进程是分配资源的基本单位，而线程则是系统调度的基本单位。由于进程是资源的拥有者，因此在其创建、撤销、切换操作中需要较大的时空开销，限制了并发程度的进一步提高。为减少进程切换的开销，通常把进程作为资源分配单位和调度单位这两个属性进行分开处理，即进程还是作为资源分配的基本单位，但是不作为调度的基本单位（很少调度或切换），把调度执行与切换的责任交给“线程”。这样做的好处是不但可以提高系统的并发度，还能使系统适应新的对称多处理机环境，充分发挥其性能。

17. 试说明线程的属性。

【参考答案】线程是一个轻型实体，是可独立调度和分派的基本单位；线程可并发执行，可共享所属进程的资源。

18. 何谓用户级线程和内核支持线程？

【参考答案】①用户级线程是仅存在于用户空间中的线程，无须内核支持。这种线程的创建与撤销，线程间的同步与通信等功能，都无须利用系统调用实现。用户级线程的切换通常发生在一个应用进程的诸多线程之间，同样无须内核支持。②内核支持线程是在内核支持下运行的线程。无论是用户进程中的线程，还是系统中的线程，它们的创建、撤销和切换等操作都是依靠内核、在内核空间中实现的。在内核空间中还为每个内核支持线程设置了TCB（thread control block，线程控制块），内核根据该TCB感知某线程的存在并对其实施控制。

19. (考研真题) 用户级线程和内核支持线程有何区别？

【参考答案】内核支持线程是在内核支持下实现的，即每个线程的TCB均设置在内核中，所有对线程的操作（如创建、撤销和切换等）都是通过系统功能调用由内核中的相应处理程序完成的。而用户级线程则仅存在于用户空间中，即每个线程的TCB均设置在用户空间中，所有对线程的操作也在用户空间中完成，而无须内核帮助。具体可从以下几个方面比较内核支持线程和用户级线程。①内核支持。用户级线程可在一个不支持线程的OS中实现，而内核支持线程则需要得到OS内核的支持。②处理机的分配。在多处机环境下，对纯粹的用户级线程来说，内核只为一个进程分配一个处理机，即进程无法享用多处机带来的好处；而当设置有内核支持线程时，内核可调度一个应用中的多个线程同时在多个处理机上并行运行，从而提高程序的执行速度和效率。③调度和线程执行时间。对设置有内核支持线程的系统，内核的调度方式和调度算法与进程的十分相似，但调度的单位是线程；而对仅设置了用户级线程的系统，内核调度的单位为进程，当进程获得CPU时，隶属于该进程的多个线程可通过用户态下的线程调度分

享由内核分配给进程的CPU执行时间。因此，在相同条件下，内核支持线程通常会得到比用户级线程更多的CPU执行时间。④切换速度。用户级线程的切换通常发生在一个应用程序的各线程之间，由于无须陷入内核，而且切换的规则也相当简单，因此切换速度比内核支持线程至少快一个数量级。⑤系统调用。在典型的OS中，许多系统调用都会引起阻塞。当一个用户级线程执行这些系统调用时，被阻塞的是整个进程；而当一个内核支持线程执行这些系统调用时，则仅阻塞这一线程，但仍可调度其所属进程的其他线程执行。

20. 试说明用户级线程和内核支持线程的实现方法。

【参考答案】本题分步解答如下。

(1) 用户级线程在用户空间中实现，运行在“运行时系统”与“内核控制线程”的中间系统上。“运行时系统”用于管理和控制线程函数的集合。“内核控制线程”可通过系统调用获得内核提供的服务，将轻型进程（light weight process, LWP）作为中间系统。

(2) 在设置了内核支持线程的OS中，系统在创建新进程时，会为其分配一个任务数据区（per task data area, PTDA），其中包括若干个TCB空间。创建一个线程，分配一个TCB，将线程的有关信息写入TCB，并为线程分配必要的资源。当任务数据区中的TCB用完而进程又有新线程时，只要所创建的线程数目未超过系统允许值，系统即可再为之分配新的TCB。在撤销一个线程时，也应回收该线程的所有资源和TCB。

2.2 综合应用题参考答案

21. 试从调度、并发、拥有资源和系统开销这4个方面比较传统进程和线程。

【参考答案】传统进程与线程的比较如表1-2-1所示。

表1-2-1 传统进程与线程的比较

| | 调度 | 并发 | 拥有资源 | 系统开销 |
|------|--|---|---|--|
| 线程 | 线程是OS中调度和分派的基本单位，具有唯一的标识符和TCB | 同一进程的多个线程可在一个/多个处理机上并发执行 | 同一进程的所有线程共享但不拥有进程的状态和资源，且驻留在进程的同一个内存地址空间中，可以访问相同的数据，通信和同步的实现也十分方便 | 作为系统调度和分派的基本单位，线程会被频繁地调度和切换；同一进程中的多线程共享同一地址空间，这能使线程实现快速切换 |
| 传统进程 | 传统进程具有独立的虚地址空间；以传统进程为单位进行任务调度时，系统必须切换地址空间，且切换时间长 | 许多“多任务OS”会限制用户拥有的最大进程数目，该数目对于许多并发应用来说是不够的 | 传统进程是系统中资源分配和保护的基本单位，也是系统调度的独立单位；每个传统进程都能以各自独立的速度在CPU上运行 | 对多个传统进程进行管理（如创建、调度、终止等）时，系统开销大，如在为了响应客户请求而须建立一个新的服务进程的服务器应用过程中，创建开销大 |

22. (考研真题) 现代 OS 一般都提供多进程 (或称多任务) 运行环境, 回答以下问题。

(1) 为支持多进程的并发执行, 系统必须建立哪些关于进程的数据结构?

(2) 为支持进程状态的变迁, 系统至少应提供哪些进程控制原语?

(3) 在执行每一个进程控制原语时, 进程状态会发生什么变化? 相应的数据结构会发生什么变化?

【参考答案】 (1) 为支持多进程的并发执行, 系统必须建立关于进程的相关数据结构, 包括PCB和队列结构 (如就绪队列、等待队列、运行指针等)。

(2) 为支持进程状态的变迁, 系统应提供的进程控制原语包括: 创建原语、阻塞原语、唤醒原语、撤销原语。

(3) 在执行每一个进程控制原语时, 进程状态及相应的数据结构有4种变化情况。①创建原语: 系统为进程创建PCB, 并对它进行初始化。进程状态由无变为就绪状态, 新创建的进程加入就绪队列中。②阻塞原语: 进程状态从运行状态变为阻塞状态, 并将阻塞进程的PCB插入相应的阻塞队列中。③唤醒原语: 进程状态从阻塞状态变为就绪状态, 从阻塞队列中删除该进程, 并将其插入就绪队列中。④撤销原语: 进程状态从运行状态变为消亡状态, 系统撤销该进程的PCB。

第3章 处理机调度与死锁

3.1 简答题参考答案

1. 高级调度与低级调度的主要任务是什么? 为什么要引入中级调度?

【参考答案】 ①高级调度的对象是作业。它的主要任务是根据某种算法, 决定将外存上处于后备队列中的哪几个作业调入内存, 为它们创建进程、分配必要的资源, 并将它们放入就绪队列。②低级调度的对象是进程 (或内核级线程)。它的主要任务是根据某种算法, 决定就绪队列中的哪个进程获得处理机, 并由分派程序将处理机分配给被选中的进程。

引入中级调度的主要目的是提高内存利用率和系统吞吐量。为此, 应把那些暂时不能运行的进程调至外存等待, 把进程状态改为就绪驻外存状态或挂起状态。当它们已具备运行条件且内存又有空间时, 由中级调度来决定把外存上的那些已具备运行条件的就绪进程再重新调入内存, 并修改它们的状态为就绪状态, 挂在就绪队列上等待。

2. 何谓作业和 JCB (job control block, 作业控制块) ?

【参考答案】 ①作业包含通常所说的程序和数据, 还配有作业说明书。系统根据该说明书对程序的运行进行控制。批处理系统中是以作业为基本单位将其从外存调入内存的。②JCB是作业在系统中存在的标志。为了管理和调度作业, 为每个作业设置一个JCB, 用于记录管理和调度作业所需的全部信息。

3. 在什么情况下需要使用 JCB ? 其中包含了哪些内容?

【参考答案】 当一个作业进入系统时, 便由“作业注册”程序为该作业建立一个JCB, 再

根据作业类型，将其插入相应的后备队列中等待调度。调度程序依据一定的调度算法来调度它们，被调度到的作业将被装入内存。在作业运行期间，系统会按照JCB中的信息和作业说明书来对作业进行控制。当一个作业执行结束并进入完成状态时，系统负责回收已分配给它的资源，并撤销其JCB。

JCB包含的内容通常有作业标志、用户名称、用户账户、作业类型（CPU繁忙型、I/O繁忙型、批量型、终端型）、作业状态、调度信息（优先级、作业已运行）、资源要求、入系统时间、开始处理时间、作业完成时间、作业退出时间、资源使用情况等。

4. 在作业调度中应如何确定接纳多少个作业和接纳哪些作业？

【参考答案】作业调度每次接纳进入内存的作业数取决于多道程序度，即取决于系统允许多少个作业同时在内存中运行。应选择哪些作业从外存调入内存，取决于所采用的作业调度算法，最简单的是FCFS（first come first served，先来先服务）调度算法，较常用的是SJF（short job first，短作业优先）调度算法和基于优先级的调度算法。

5. 试说明低级调度的主要功能。

【参考答案】低级调度是在就绪队列中选择某个进程，由分派程序把CPU分配给该进程。低级调度的主要功能有：①保存当前进程的CPU现场信息；②按某种算法选择投入执行的新进程；③恢复新进程的CPU现场，从而将CPU分配给新进程。

6.（考研真题）简述引起进程调度的原因。

【参考答案】引起进程调度的常见原因有：①正在执行的进程正常终止或异常终止；②正在执行的进程因某种原因而阻塞，如提出I/O请求后被阻塞、在调用wait操作时因资源不足而阻塞、因其他原因执行block原语而阻塞等；③在引入时间片的系统中，时间片用完；④在抢占式调度方式中，就绪队列中某进程的优先级比当前正在执行进程的高，或者有优先级更高的进程进入就绪队列。

7. 在抢占式调度算法中，抢占的原则是什么？

【参考答案】抢占的原则有：①优先级原则，指允许优先级高的新到进程抢占当前进程的处理机；②短进程优先原则，指允许新到的短进程抢占当前长进程的处理机；③时间片原则，即各进程按时间片轮转运行时，当正在执行的进程的一个时间片用完后，便停止该进程的执行而重新进行调度。

8. 在选择调度方式和调度算法时，应遵循哪些准则？

【参考答案】一般而言，在设计一个OS时，应如何选择调度方式和调度算法，很大程度上取决于OS的类型及其设计目标。在批处理系统、分时系统和实时系统中，通常会采用不同的调度方式和调度算法。遵循的准则包括：①处理机调度算法的共同目标，即资源利用率高、保证公平性与平衡性以及策略强制执行；②批处理系统的目标，即平均周转时间短、系统吞吐量大、处理机利用率高；③分时系统的目标，即响应时间快、保证均衡性；④实时系统的目标，即保证截至时间和可预测性。

9. 何谓静态优先级和动态优先级？确定进程优先级的依据是什么？

【参考答案】①静态优先级是指在创建进程时确定的、在进程的整个运行期间保持不变的优先级。②动态优先级是指在创建进程之初，先赋予进程一个优先级，然后其值会随进程的推进或等待时间的增加而改变，如此而为目的的是获得更好的调度性能。此外，确定进程优先级

的依据有进程类型、进程对资源的需求和用户要求等。

10. 试比较 FCFS 和 SJF 这两种调度算法。

【参考答案】本题分步解答如下。

(1) 相同点：这两种调度算法都可以用于作业调度和进程调度。

(2) 不同点：FCFS调度算法每次都从后备队列中选择一个或多个最先进入该队列的作业，将它们调入内存、分配资源、创建进程、插入就绪队列。该算法有利于长作业/长进程，不利于短作业/短进程。SJF调度算法每次调度都从后备队列中选择一个或若干个估计运行时间最短的作业，将其调入内存运行。该算法有利于短作业/短进程，不利于长作业/长进程。

11. 在基于时间片的 RR (round robin, 轮转) 调度算法中, 应如何确定时间片的大小?

【参考答案】在RR调度算法中, 时间片的大小对系统性能有很大的影响。通常, 时间片应略大于一次典型的交互所需的时间, 使大多数交互式进程能在一个时间片内完成, 从而获得很小的响应时间。在确定时间片的大小时, 一般应考虑3个因素: 系统对相应时间的要求、就绪队列中进程的数目和系统的处理能力。

12. 为什么说多级反馈队列调度算法能较好地满足各方面用户的需求?

【参考答案】本题分步解答如下。

(1) 对于终端型作业用户, 其所提交的作业大都属于较小的交互型作业, 系统只要使这些作业在第1队列规定的时间片内完成, 终端型作业用户就会感到满足。

(2) 对于短批处理作业用户, 如果其作业在第1队列中执行一个时间片即可完成, 便可获得与终端作业一样的响应时间。对于稍长批处理作业用户, 其作业通常只须在第2队列和第3队列各执行1个时间片即可完成, 周转时间仍然较短。

(3) 对于长批处理作业用户, 其将依次在第1, 2, ..., n 个队列中运行, 然后再按轮转方式运行, 用户不必担心其作业长期得不到处理。因此, 多级反馈队列调度算法能满足多用户需求。

13. 为什么在实时系统中要求系统 (尤其是 CPU) 具有较强的处理能力?

【参考答案】在实时系统中都存在着若干个实时进程或任务, 它们用来反映或控制某个 (些) 外部事件, 往往带有某种程度的紧迫性, 因而对实时系统中的调度提出了某些特殊要求。若CPU的处理能力不够强, 则有可能会因CPU忙不过来而使某些实时任务得不到及时处理, 从而导致发生难以预料的后果。

14. 按照调度方式可将实时调度算法分为哪几种?

【参考答案】按照调度方式可将实时调度算法分为非抢占式调度算法和抢占式调度算法两种。①非抢占式调度算法, 包括非抢占式RR调度算法和非抢占式优先级调度算法; ②抢占式调度算法, 根据抢占时机的不同分为基于时钟中断的抢占式优先级调度算法和立即抢占的优先级调度算法。

15. 实时系统常用的调度算法有哪些? 请分别介绍它们。

【参考答案】实时系统的调度算法主要是基于任务开始/截止时间的EDF (earliest deadline first, 最早截止时间优先) 调度算法, 以及基于任务紧急/松弛程度的LLF (least laxity first, 最低松弛度优先) 调度算法。

(1) EDF调度算法。该算法是根据任务的开始/截止时间确定任务的优先级的, 截止时间越早则优先级越高。算法要求在系统中保持一个实时任务就绪队列, 该队列按各任务截止时间的

先后顺序进行排序。

(2) LLF调度算法。该算法是根据任务的紧急/松弛程度确定任务的优先级的。任务的紧急程度越高, 赋予该任务的优先级就越高, 以使其优先执行。例如, 一个任务在200ms时必须完成, 而它本身所需的运行时间就是100ms, 因此, 调度程序必须在100ms或之前调度执行, 该任务的紧急程度(松弛程度)为100ms。

16. 在批处理系统、分时系统和实时系统中, 各采用哪几种进程(作业)调度算法?

【参考答案】①在批处理系统中可采用的调度算法有SJF、优先级、高响应比优先、多级反馈队列等调度算法。②在分时系统中常用的调度算法是RR调度算法。③在实时系统中可采用的调度算法有EDF调度算法和LLF调度算法。

17. (考研真题) 什么是死锁? 产生死锁的原因和必要条件是什么? 如何预防死锁?

【参考答案】死锁是指多个进程在运行过程中因争夺资源而造成的一种僵局, 当进程处于这种僵持状态时, 若无外力作用, 它们都将无法向前推进。产生死锁的原因有: ①竞争不可抢占性资源; ②竞争可消耗资源; ③进程间推进顺序不当。产生死锁必须同时具备4个必要条件: 互斥条件、请求和保持条件、不可抢占条件、循环等待条件。

预防死锁是通过破坏产生死锁的4个必要条件中的一个或几个来实现的。其中互斥条件是设备固有属性, 不能改变, 因此主要破坏产生死锁的其他3个必要条件。①破坏“请求和保持”条件。当一个进程在请求资源时, 它不能持有不可抢占性资源。②破坏“不可抢占”条件。当一个已经保持了某些不可抢占性资源的进程提出新的资源请求而不能得到满足时, 它必须释放已经保持的所有资源, 待以后需要时再重新申请。③破坏“循环等待”条件。对系统所有资源类型进行线性排序, 并赋予它们不同的序号, 规定每个进程必须按序号递增的方式请求资源。

18. 在解决死锁问题的几个方法中, 哪个方法最易于实现? 哪个方法可使资源利用率最高?

【参考答案】解决死锁的方法有预防死锁、避免死锁、检测和解除死锁, 其中预防死锁这一方法最容易实现, 但由于所施加的限制条件过于严格, 资源利用率和系统吞吐量会降低; 而检测和解除死锁这一方法可使系统获得较好的资源利用率和系统吞吐量。

3.2 计算题参考答案

19. (考研真题) 有5个进程(见表1-3-1)需要调度执行, 若采用非抢占式优先级(短进程优先)调度算法, 问这5个进程的平均周转时间是多少?

表1-3-1 进程执行时间表

| 进程 | 到达时间 | 执行时间 |
|----------------|------|------|
| P ₁ | 0.0 | 9 |
| P ₂ | 0.4 | 4 |
| P ₃ | 1.0 | 1 |
| P ₄ | 5.5 | 4 |
| P ₅ | 7 | 2 |

【参考答案】本题分步解答如下。

(1) 采用非抢占式优先级（短进程优先）调度算法，各进程执行次序如图1-3-1所示。在0时刻，进程 P_1 到达， P_1 获得处理机。由于不可抢占，因此 P_1 会一直运行到9时刻结束；在9时刻，所有进程到达，根据短进程优先调度， P_3 获得处理机，然后是 P_5 ； P_2 和 P_4 的预计运行时间一样，按照FCFS原则， P_2 优先获得处理机，最后是 P_4 。

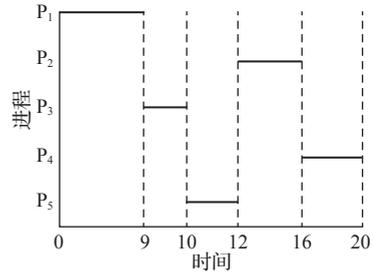


图 1-3-1 进程运行时间关系图

(2) 周转时间=完成时间-作业到达时间。因此 P_1 的周转时间是 $9-0=9$ ， P_2 的周转时间是 $16-0.4=15.6$ ， P_3 的周转时间是 $10-1=9$ ， P_4 的周转时间是 $20-5.5=14.5$ ， P_5 的周转时间是 $12-7=5$ 。这5个进程的平均周转时间为 $(9+15.6+9+14.5+5)/5=10.62$ 。

20. (考研真题) 假定要在—台处理机上执行表 1-3-2 所示的作业，且假定这些作业在时刻 0 以 1, 2, 3, 4, 5 的顺序到达。请说明分别采用 FCFS、RR (时间片为 1)、SJF 及非抢占式优先级调度算法时，这些作业的执行情况 (优先级的高低顺序依次为 1 到 5)。针对上述每种调度算法，给出平均周转时间和平均带权周转时间。

表 1-3-2 作业执行时间表

| 作业 | 执行时间 | 优先级 |
|----|------|-----|
| 1 | 10 | 3 |
| 2 | 1 | 1 |
| 3 | 2 | 3 |
| 4 | 1 | 4 |
| 5 | 5 | 2 |

【参考答案】本题分步解答如下。

(1) 作业执行顺序如图1-3-2所示。

FCFS:



RR:



SJF:



非抢占式优先级:



图 1-3-2 作业执行顺序图

(2) 各个作业对应于各个算法的周转时间和带权周转时间见表1-3-3。

表1-3-3 作业运行情况比较表

| 算法 | 时间类型 | P ₁ | P ₂ | P ₃ | P ₄ | P ₅ | 平均时间 |
|-------------|--------|----------------|----------------|----------------|----------------|----------------|------|
| | | 运行时间 | 10 | 1 | 2 | 1 | 5 |
| FCFS | 周转时间 | 10 | 11 | 13 | 14 | 19 | 13.4 |
| | 带权周转时间 | 1 | 11 | 6.5 | 14 | 3.8 | 7.26 |
| RR | 周转时间 | 19 | 2 | 7 | 4 | 14 | 9.2 |
| | 带权周转时间 | 1.9 | 2 | 3.5 | 4 | 2.8 | 2.84 |
| SJF | 周转时间 | 19 | 1 | 4 | 2 | 9 | 7 |
| | 带权周转时间 | 1.9 | 1 | 2 | 2 | 1.8 | 1.74 |
| 非抢占式 优先级 | 周转时间 | 16 | 1 | 18 | 19 | 6 | 12 |
| | 带权周转时间 | 1.6 | 1 | 9 | 19 | 1.2 | 6.36 |

因此，FCFS的平均周转时间为13.4，平均带权周转时间为7.26。RR的平均周转时间为9.2，平均带权周转时间为2.84。SJF的平均周转时间为7，平均带权周转时间为1.74。非抢占式优先级的平均周转时间为12，平均带权周转时间为6.36。注意：SJF的平均周转时间最短。

21. (考研真题) 将一组进程分为 4 类，如图 1-3-3 所示。各类进程之间采用优先级调度算法，而各类进程的 内部采用 RR 调度算法。请简述 P₁，P₂，P₃，P₄，P₅，P₆，P₇，P₈ 进程的调度过程。

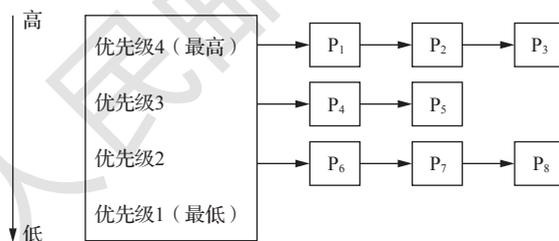


图 1-3-3 进程分类图

【参考答案】由于不同类进程间采用优先级调度算法，同类进程间采用RR调度算法，因此，系统首先对优先级为4的进程P₁、P₂、P₃采用RR调度算法进行运行；当P₁、P₂、P₃运行结束或阻塞时，再对优先级为3的进程P₄、P₅采用RR调度算法进行运行。在此期间，若P₁、P₂、P₃队列中有转为就绪状态的进程，则优先级3队列的当前时间片用完后回到优先级4队列进行进程调度。类似地，当进程P₁~P₅运行结束或阻塞时，对优先级为2的进程P₆、P₇、P₈采用RR调度算法进行运行，一旦进程P₁~P₅中有一个转为就绪状态，当前时间片用完后就立即回到相应的优先级队列进行RR调度。

22. 由 5 个进程组成进程集合 $P=\{P_0, P_1, P_2, P_3, P_4\}$ ，系统中有 3 类资源 A, B, C，假设在某时刻有表 1-3-4 所示的进程资源分配情况。

表1-3-4 进程资源分配情况

| 进程 | Allocation | Max | Available |
|----------------|------------|-------|-----------|
| | A B C | A B C | A B C |
| P ₀ | 0 0 3 | 0 0 4 | x y z |
| P ₁ | 1 0 0 | 1 7 5 | |
| P ₂ | 1 3 5 | 2 3 5 | |
| P ₃ | 0 0 2 | 0 6 4 | |
| P ₄ | 0 0 1 | 0 6 5 | |

请问当 x, y, z 取下列值时, 系统是否处于安全状态?

(1) 1, 4, 0; (2) 0, 6, 2; (3) 1, 1, 1; (4) 0, 4, 7。

【参考答案】由银行家算法中Need、Max、Allocation间的关系可得:

$$\text{Need} = \text{Max} - \text{Allocation} = \begin{bmatrix} 0 & 0 & 4 \\ 1 & 7 & 5 \\ 2 & 3 & 5 \\ 0 & 6 & 4 \\ 0 & 6 & 5 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 3 \\ 1 & 0 & 0 \\ 1 & 3 & 5 \\ 0 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 7 & 5 \\ 1 & 0 & 0 \\ 0 & 6 & 2 \\ 0 & 6 & 4 \end{bmatrix}。$$

(1) 当Available为(1, 4, 0)时, 根据Need矩阵可知, 可满足P₂的需求; P₂结束后释放资源, Available为(2, 7, 5)可满足P₀、P₁、P₃、P₄中任一进程的需求, 因此系统不会出现死锁, 处于安全状态。

(2) 当Available为(0, 6, 2)时, 可满足进程P₀、P₃的需求; 进程结束后释放资源, Available为(0, 6, 7), 可满足P₄的需求; P₄结束后释放资源, Available为(0, 6, 8), 此时不能满足其余任一进程的需求, 系统出现死锁, 因此当前处在非安全状态。

(3) 当Available为(1, 1, 1)时, 可满足进程P₀、P₂的需求; 这两个进程结束后释放资源, Available为(2, 4, 9), 此时不能满足其余任一进程的需求, 系统出现死锁, 处于非安全状态。

(4) 当Available为(0, 4, 7)时, 可满足P₀的需求, 进程结束后释放资源, Available为(0, 4, 10), 此时不能满足其余任一进程的需求, 系统出现死锁, 处于非安全状态。

3.3 综合应用题参考答案

23. (考研真题) 假设系统中有下述 3 种解决死锁的方法:

- (1) 银行家算法;
- (2) 检测死锁, 终止处于死锁状态的进程, 释放该进程所占有的资源;
- (3) 资源预分配。

简述上述哪种方法允许最大的并发性？请按“并发性”从大到小对上述3种方法进行排序。

【参考答案】①题中给出的3种方法中，检测死锁能允许更多的进程无须等待地向前推进，并发性最大。因为该方法允许进程最大限度地申请并分配资源，直至出现死锁，再由系统解决死锁。②银行家算法允许进程自由申请资源，只是在某个进程申请资源时会检查系统是否处于安全状态，若是，则可立即分配；若不是，则拒绝。其并发性的大小次于检测死锁这一方法。③最后是资源预分配，因为此方法要求进程在运行之前申请所需的全部资源，这会使许多进程因申请不到全部资源而无法开始，得到部分资源的进程也会因未得到全部资源而不释放已占用的资源，所以此方法会导致资源浪费。因此，上述3种方法按并发性从大到小排序为：检测死锁、银行家算法、资源预分配。

24. (考研真题) 某银行要实现一个电子转账系统，基本业务流程是：首先对转出方和转入方的账户进行加锁，然后办理转账业务，最后对转出方和转入方的账户进行解锁。若不采取任何措施，则系统会不会发生死锁？为什么？请设计一个能够避免死锁的方法。

【参考答案】本题分步解答如下。

(1) 系统会发生死锁。原因如下：假如两个账号A和B有两个转账业务，分别是A转入B和B转入A。这两个业务在执行时可能会发生以下情况：一个业务锁定A账户，试图锁定B账户失败而等待；另一个业务锁定B账户，试图锁定A账户失败，也在等待，进而即会导致系统处于死锁状态。

(2) 为了避免死锁，可采用两阶段加锁方法，即为每个账户设定一个唯一的账号，在业务执行前，必须按照账号的大小顺序依次获得所有账户的锁，业务完成后再依次按照锁定的先后次序将后锁定的账户先解锁。

25. (考研真题) 设有进程 P_1 和进程 P_2 并发执行，它们都需要使用资源 R_1 和 R_2 ，使用资源情况如表 1-3-5 所示。

表1-3-5 进程使用资源情况

| 进程 P_1 | 进程 P_2 |
|------------|------------|
| 申请资源 R_1 | 申请资源 R_2 |
| 申请资源 R_2 | 申请资源 R_1 |
| 释放资源 R_1 | 释放资源 R_2 |

试判断是否会发生死锁，并解释和说明发生死锁的原因与必要条件。

【参考答案】这两个进程在不同的推进速度下，可能会产生死锁。比如：进程 P_1 先申请资源 R_1 ， P_1 得到 R_1 后进程 P_2 申请 R_2 ， P_2 得到 R_2 后， P_1 又申请资源 R_2 ，此时则因 R_2 已分配，故 P_1 阻塞。 P_1 和 P_2 两个进程因申请不到所需资源而形成死锁。如果改变进程的运行顺序，则这两个进程可能又不会发生死锁。因此，产生死锁的原因可归结为两点：①竞争资源；②进程推进顺序非法。产生死锁的必要条件：①互斥条件；②请求和保持条件；③不可抢占条件；④循环等待条件。

第4章 进程同步

4.1 简答题参考答案

1. 什么是临界资源？什么是临界区？

【参考答案】①在计算机中有许多资源一次仅允许一个进程使用，我们把一次仅允许一个进程使用的资源称为临界资源，如打印机和一些共享变量等。②进程中访问临界资源的那段代码称为临界区。

2. 同步机制应遵循的准则有哪些？

【参考答案】同步机制应遵循的准则主要有4个。

(1) 空闲让进。当无进程处于临界区时，表明临界资源处于空闲状态，应允许一个请求进入临界区的进程立即进入临界区，以有效地利用临界资源。

(2) 忙则等待。当已有进程在临界区时，表明临界资源正在被访问，因而其他试图进入临界区的进程必须等待，以保证对临界资源的互斥访问。

(3) 有限等待。对要求访问临界资源的进程，应保证其能在有限时间内进入临界区，以免陷入“死等”状态。

(4) 让权等待。当进程不能进入临界区时，其应立即释放处理机，以免进程陷入“忙等”。

3. 为什么各进程对临界资源的访问必须互斥？

【参考答案】临界资源本身的特性决定了它们只能被各个进程互斥地访问，如果并发执行的多个进程同时访问临界资源，则会造成系统混乱或程序执行结果不确定。这样，进程运行结果就可能不正确或者不确定。比如，两个进程并发执行如下程序段：

```
mov ax, (counter);  
inc ax;  
mov (counter), ax;
```

其中，共享变量counter初值为0，对counter执行加1操作。如果允许一个进程访问counter，另一个进程也可以对其进行操作，则counter的值最终可能是正确结果2，也可能是错误结果1，即计算结果出现了不确定性。因此，各进程对临界资源的访问必须互斥地进行。

4. 如何保证各进程互斥地访问临界资源？

【参考答案】为了互斥地访问临界资源，系统必须保证进程互斥地进入临界区。为此，必须在临界区前增加一段进入区代码，以检查是否有其他进程已进入临界区而在使用临界资源。若有，则进程必须等待；否则，允许进程进入临界区，同时设置标志以表示有进程正在临界区内。同样，在临界区后必须增加一段退出区代码，用于将已有进程进入临界区访问临界资源的标志改为无进程进入临界区使用临界资源。进入区和退出区可用多种同步机制实现，如锁、信

号量机制等。

5. 何谓“忙等”？它有什么缺点？

【参考答案】①“忙等”是指“不让权”的等待，即进程因某事件的发生而无法继续执行时，它仍占用CPU，并通过不断地执行循环测试指令来等待该事件的完成。②“忙等”的主要缺点是浪费CPU时间，另外，它还可能会引起预料不到的后果。例如，考虑某个采取高优先级优先调度原则的系统，目前有两个进程A和B共享某个临界资源，A的优先级较高，B的优先级较低，且B已处于临界区内，而A欲进入自己的临界区，则A、B都不可能继续向前推进，进而即会陷入“死等”状态。

6. 试述采用 Peterson 算法实现临界区互斥的原理。

【参考答案】Peterson算法中 P_i 和 P_j 两个进程共享turn和flag两个变量。turn=i表示 P_i 进程可以进入临界区，flag[i]=TRUE表示 P_i 进程准备进入临界区。因此，可能会存在以下几种请求情况。

(1) 当 P_i 在临界区中时，若 P_j 请求进入临界区，则flag[j]=TRUE，turn=i，flag[j]=TRUE，即 P_j 中flag[j]&&turn=i为TRUE，此时 P_j 会循环执行while语句——“忙等”而无法进入临界区，满足“忙则等待”。

(2) 如果临界区目前空闲，但 P_i 请求进入，而 P_j 未请求，此时flag[i]=TRUE，turn=j，flag[j]=FALSE，因此 P_i 的while条件为FALSE， P_i 可进入临界区执行。如果两个进程都要进入临界区，即flag[i]=flag[j]=TRUE，则turn只能取0或1，只能有一个进程进入临界区。当一个进程退出临界区后，另一个进程即可进入临界区，满足“有限等待”。

7. 哪些硬件方法可以解决进程互斥问题？简述它们的用法。

【参考答案】解决进程互斥问题可采用的硬件方法主要有下列3种。

(1) 利用“关中断”实现，将临界区放在关中断与开中断之间，关中断后不允许当前进程被中断，因此进程在临界区执行期间都不允许被中断，不能发生进程切换；进程访问完临界区后，再执行开中断指令，此时其他进程才能获得处理机并访问临界区，进而有效保证进程互斥。

(2) Test-and-Set指令，简称TS指令或TSL指令。该指令是一条硬件指令，指令执行过程中不允许被中断，即TS指令把“上锁”和“检查”操作作用硬件的方式变成了一气呵成的原子操作。指令执行过程为：①为每个临界资源设置一个布尔变量lock，表示当前临界区是否加锁；②进程进入临界区前，首先用TS指令测试lock，若其值为FALSE，则表示没有进程在临界区内，while循环条件不满足，进入临界区，并将TRUE值赋给lock，即关闭临界区；③任何其他进程再利用TS指令测试lock，while都会一直循环，直到当前访问临界区的进程在退出区进行“解锁”，从而实现了进程互斥。

(3) Swap指令，该指令是用硬件实现的，执行过程不允许中断。其用法是为每个临界资源设置一个全局布尔变量lock，初值为FALSE，在每个进程中再利用一个局部布尔变量key，使用Swap指令与lock进行数值交换，循环判断lock的取值。只有当key为FALSE时，进程才可进入临界区进行操作。从逻辑上看，Swap指令和TS指令并无太大区别，都是先记录下此时临界区是否已经被上锁，再将上锁标记lock设置为TRUE，最后检查局部布尔变量key，如果key为FALSE，则说明之前没有别的进程对临界区上锁，此时可跳出循环，进入临界区。

8. (考研真题) 如果用于进程同步的信号量的 P、V 操作不用原语实现，则会产生什么后果？举例说明。

【参考答案】例如：利用P、V操作实现A、B进程对临界资源的互斥使用，代码如下。

```

semaphore S=1;
A(){
    while(1){
        P(S);
        临界区;
        V(S);
        剩余区;
    }
}
B(){
    while(1){
        P(S);
        临界区;
        V(S);
        剩余区;
    }
}

```

若P、V操作不被设计成原语，则执行P、V操作时进程可以被中断。A、B并发执行，初始状态下，临界资源空闲，故应允许第一个申请临界资源的进程（假设为A进程）进入临界区而使用临界资源。但如果A执行到P操作的语句S.value--后（此时S.value的值为0）被B中断，B进程执行P操作，则当B进程执行语句S.value--且S.value的值变为-1时，由于S.value<0，B会被阻塞，A进程再次获得CPU后，同样也会因为S.value<0而被阻塞，这就出现了临界资源虽然空闲但进程申请不到的情况，即此时P、V操作无法满足同步机制中“空闲让进”的要求。同样，一个执行P操作的进程被中断后，另一个进程去执行V操作；或一个执行V操作的进程被中断后，另一个进程去执行P或V操作，都将发生混乱，难以实现进程同步。因此，P、V操作必须设计成原语的方式。

9. AND 信号量机制的基本思想是什么？它能解决什么问题？

【参考答案】①AND信号量机制的基本思想是将进程在整个运行过程中所需要的所有临界资源一次性全部分配给进程，待该进程使用完后再一起释放。只要尚有一个所需资源未能分配给该进程，则其他所有将为之分配的资源都不分配给它。亦即，对若干个临界资源的分配采取原子操作方式，要么全部分配到进程，要么一个也不分配。②它能解决的问题是防止死锁的发生，因为该方法在资源分配过程中使产生的死锁必要条件中的“请求和保持”条件不被满足。

10. 利用信号量机制实现进程互斥时，对互斥信号量的wait()和signal()操作为什么要成对出现？

【参考答案】利用信号量机制实现进程互斥时，对互斥信号量mutex的wait()和signal()操作必须成对出现，缺少wait(mutex)将会导致系统混乱，不能保证进程对临界资源的互斥访问；而缺少signal(mutex)则将会使临界资源永远不被释放，从而使因等待该资源而阻塞的进程不能被唤醒。

11. 什么是管程？它有哪些特性？

【参考答案】由代表共享资源的数据结构以及由对该共享数据结构实施操作的一组过程所组成的资源管理程序共同构成的一个OS资源管理模块，称为管程。

管程是一种程序设计语言结构成分，从语言的角度看，管程主要有以下特性：①模块化，管程是一个基本程序单位，可以单独编译；②抽象数据类型，管程中不仅有数据，而且有针对数据的操作；③信息掩蔽，管程中的数据结构只能被管程中的过程访问，这些过程在管程内部被定义，供管程外的进程调用，而管程中的数据结构以及过程（函数）的具体实现在外部不可见。

12. 试述管程中条件变量的含义和作用。

【参考答案】条件变量是管程内的一种数据结构，且只有在管程中才能被访问，它对于管程内的所有过程而言是全局变量，只能通过两个原语操作来控制它。①x.wait()原语。调用

进程阻塞并移入与条件变量x相关的队列中，释放管程，直到另一个进程在该条件变量x上执行signal()以唤醒等待进程，并将其移出条件变量x的队列。②x.signal()原语。如果存在其他进程由于对条件变量x执行wait()操作而被阻塞，则释放之；如果没有进程在等待，则信号被丢弃。

条件变量是一种信号量，起到了维护等待进程队列的作用。当管程中的进程被阻塞或挂起而不能运行时，如果该进程不释放管程，则其他进程就无法进入管程，此时就需要条件变量来进行控制。

4.2 计算题参考答案

13. 若信号量的初值为2，当前值为-1，则表示有多少个等待进程？请分析。

【参考答案】信号量的初值表示系统中资源的数目，每次的P操作表示进程请求一个单位的资源，信号量进行减1操作，当信号量小于0时，表示资源已分配完毕，进程自我阻塞。如果信号量小于0，那么信号量的绝对值表示当前阻塞队列中进程的个数。因此，当前值为-1，表示有1个等待进程。

14. 有 m 个进程共享同一临界资源，若使用信号量机制实现对某个临界资源的互斥访问，请求出信号量的变化范围。

【参考答案】某个临界资源的信号量初值为1，其是信号量的最大值。 m 个进程分别对临界资源发出1次请求，信号量均要执行减1操作，因此，最多可允许 m 个进程同时申请，此时信号量的值是 $1-m$ ，为最小值。因此，信号量值的范围是 $1-m$ 至1。

15. 若有4个进程共享同一程序段，而且每次最多允许3个进程进入该程序段，则信号量值的变化范围是什么？

【参考答案】程序段作为共享资源，最多允许3个进程进入其中，因此设置信号量初值为3。当4个进程共享该程序段时，在每个进程申请进入时，信号量都会执行减1操作。当第1个进程申请进入时，信号量值变为2；第2个进程申请进入时，信号量值变为1；第3个进程申请进入时，信号量值变为0，第4个进程申请进入时，信号量值变为-1。因此，信号量的变化范围是3，2，1，0，-1。

4.3 综合应用题参考答案

16. (考研真题) 3个进程 P_1 、 P_2 、 P_3 互斥地使用一个包含 N ($N > 0$)个单元的缓冲区。 P_1 每次用produce()生成一个正整数，并用put()将其送入缓冲区的某一空单元中； P_2 每次用getodd()从该缓冲区中取出一个奇数，并用countodd()统计奇数的个数； P_3 每次用geteven()从该缓冲区中取出一个偶数，并用counteven()统计偶数的个数。请用信号量机制实现这3个进程的同步与互斥活动，并说明所定义的信号量的含义。要求用伪代码描述。

【参考答案】定义资源信号量empty、odd、even，用于控制生产者与消费者之间的同步，其中，empty表示空缓冲区的数目，odd表示缓冲区中奇数的个数，even表示缓冲区中偶数的个数；定义互斥信号量mutex，用于实现进程对缓冲区的互斥访问。伪代码描述如下：

```

semaphore empty=N,even=0,odd=0,mutex=1;
P1:
while(1){
    x=produce( );
    P(empty);
    P(mutex);
    put(x);
    V(mutex);
    if (x%2==0)
        V(even);
    else
        V(odd);
}

P2:
while(1){
    P(odd);
    P(mutex);
    getodd();
    countodd();
    V(mutex);
    V(empty);
}

P3:
while(1){
    P(even);
    P(mutex);
    geteven( );
    counteven( );
    V(mutex);
    V(empty);
}

```

17. (考研真题) 某银行提供了 1 个服务窗口和 10 个供顾客等待时使用的座位。顾客到达银行时, 若有空座位, 则到取号机上领取一个号, 等待叫号。取号机每次仅允许一位顾客使用。当营业员空闲时, 通过叫号选取一位顾客, 并为其服务。顾客和营业员的活动过程描述如下。

```

cobegin {
    process 顾客 {
        从取号机上获得一个号码;
        等待叫号;
        获得服务;
    }
    process 营业员 {
        while (TRUE) {
            叫号;
            为顾客服务;
        }
    }
} coend

```

请添加必要的信号量和 P、V 操作或 wait()、signal() 操作, 实现上述过程中的互斥与同步。要求写出完整的过程, 说明信号量的含义并赋初值。

【参考答案】

```

semaphore numget=1, seats=10, custom=0; //numget 是关于取号机互斥的信号量; 信号量
seats 是座位的个数; 信号量 custom 是顾客的个数

```

```

process 顾客 {
    P(seats); // 看有没有空座位
    P(numget); // 取号
    取号;
    V(numget); // 取完号后释放取号机
    V(custom);
    等待叫号;
    V(seats);
    接受服务;
}

process 营业员 {
    P(custom);
    叫号;
    为顾客服务;
}

```

18. 如图 1-4-1 所示, 有 1 个计算进程和 1 个打印进程, 它们共享一个单缓冲区, 计算进程不断计算出一个整型结果, 并将它放入单缓冲区中; 打印进程则负责从单缓冲区中取出每个结果并进行打印。请用信号量机制来实现它们的同步关系。



图 1-4-1 共享单缓冲区的计算进程和打印进程

【参考答案】由题意可知, 本题中计算进程和打印进程为合作的同步关系。计算进程需要向空闲缓冲区中放入计算好的数据, 因此要设置它所需要的empty信号量, 由于开始时缓冲区为空, 因此empty初值为1; 打印进程需要输出已放入缓冲区中的打印结果, 因此需要设置它所需要的信号量full, 初始状态下缓冲区中无结果可供打印, 故full的初值为0。

```

semaphore full=0, empty=1;
int buffer;
cp() {
    int nextc;
    while(1) {
        compute the next number nextc;
        P(empty);
        buffer=nextc;
        V(full);
    }
}

pp() {
    int nextp;
    while(1) {
        P(full);
        nextp=buffer;
        V(empty);
        print the number in nextp;
    }
}

main() {
    cobegin
        cp();
        pp();
    coend
}

```

19. 有 3 个进程 P_1 、 P_2 、 P_3 协作解决文件打印问题。 P_1 将文件记录从磁盘读入内存的缓冲区 1，每执行一次读一个记录； P_2 将缓冲区 1 中的内容复制到缓冲区 2 中，每执行一次复制一个记录； P_3 将缓冲区 2 中的内容打印出来，每执行一次打印一个记录。缓冲区的大小与记录大小一样。请用信号量机制来保证文件的正确打印。

【参考答案】对缓冲区 1 来说， P_1 是生产者， P_2 是消费者；对缓冲区 2 来说， P_2 是生产者， P_3 是消费者。缓冲区 1 和缓冲区 2 都只能存放一个记录，它们都是临界资源，但无须使用信号量来实现互斥。 P_2 对于缓冲区 1 是消费者，对于缓冲区 2 是生产者，因此要对 P_2 设置两个信号量来分别控制其对不同缓冲区的不同操作。该文件打印过程的同步算法可描述为：

```
semaphore empty1=1,full1=0,empty2=1,full2=0;

P1() {
    while(1) {
        从磁盘读一个记录;
        P(empty1);
        将记录存放到缓冲区 1 中;
        V(full1);
    }
}

P2() {
    while(1) {
        P(full1);
        P(empty2);
        从缓冲区 1 中取一个记录;
        将记录复制到缓冲区 2 中;
        V(empty1);
        V(full2);
    }
}

P3() {
    while(1){
        P(full2);
        从缓冲区 2 中取一个记录;
        V(empty2);
        将取出的记录打印出来;
    }
}

main() {
    cobegin
        P1();
        P2();
        P3();
    coend
}
```

20. 桌上有一个能盛得下 5 个水果的空盘子。爸爸不停地向盘中放苹果或橘子，儿子不停地从盘中取出橘子享用，女儿不停地从盘中取出苹果享用。规定 3 人不能同时向（从）盘子中放（取）水果。试用信号量来实现爸爸、儿子和女儿这 3 个“循环进程”之间的同步。

【参考答案】分析：本题是生产者-消费者问题的变形，相当于一个能生产两种产品的生产者（爸爸）向两个消费者（儿子和女儿）提供产品的同步问题，因此，须设置两个不同的 full 信号量 apple 和 orange，它们的初值均为 0。为了描述上述同步问题，可定义如下信号量：

```
semaphore empty=5, orange=0, apple=0, mutex=1;
```

爸爸、儿子、女儿的算法可描述为：

```

Dad() {
    while(1) {
        P(empty);
        P(mutex);
        将水果放入盘中;
        V(mutex);
        if(放入的是橘子)
            V(orange);
        else
            V(apple);
    }
}

Son() {
    while(1) {
        P(orange);
        P(mutex);
        从盘中取一个橘子;
        V(mutex);
        V(empty);
        享用橘子;
    }
}

Daughter() {
    while(1) {
        P(apple);
        P(mutex);
        从盘中取一个苹果;
        V(mutex);
        V(empty);
        享用苹果;
    }
}

```

21. 试用记录型信号量写出一个不会死锁的哲学家进餐问题的算法。

【参考答案】此题有多种解法。其中之一是只允许4个哲学家同时进餐，以保证至少有1个哲学家可以进餐，最终才可能由他释放出其所用过的两根筷子，从而使更多的哲学家可以进餐。为此，须设置一个信号量Sm来限制同时进餐的哲学家数目，使它不超过4，因此可将Sm的初值设置为4。

除了为每根筷子设置一个初值为1的信号量chopstick[i] (i=0, ..., 4) 外，还须再设置一个初值为4的信号量Sm。第i个哲学家的活动可描述为：

```

Pi() {
    while(1) {
        P(Sm);
        P(chopstick[ i ]);
        P(chopstick[ (i+1)% 5 ]);
        eat;
        V(chopstick[ i ]);
        V(chopstick[ (i+1)% 5 ]);
        V(Sm);
        think;
    }
}

```

第5章 存储器管理

5.1 简答题参考答案

1. 存储器管理的基本任务,是为多道程序的并发执行提供良好的存储器环境。请问:“良好的存储器环境”应包含哪几个方面?

【参考答案】“良好的存储器环境”应包含:①让每道程序“各得其所”,在不受干扰的环境中运行,还可以使用户从存储空间的分配、保护等烦琐事务中解脱出来;②向用户提供更大的存储空间,使更多的作业能同时运行,或使更大的作业能在较小的内存空间中运行;③为用户在信息的访问、保护、共享以及动态链接等方面提供方便;④使存储器有较高的利用率。

2. 内存保护是否可以完全由软件实现?为什么?

【参考答案】内存保护的主要任务,是确保每道程序都只能在自己的内存区中运行。这就要求系统能对每条指令所访问的地址是否超出自己内存区的范围进行越界检查,一旦发生越界,系统应能立即发出越界中断请求,抛弃该指令。若此检查完全用软件实现,则每执行一条指令时,都需要增加若干条指令去执行越界检查,这大大降低了程序的执行速度。因此,越界检查通常由硬件实现,以使指令能够与越界检查并行执行,从而不降低程序的执行速度。当然,越界后的处理仍须由软件配合完成。因此,内存保护是由硬件和软件协同完成的。

3. (考研真题)请解释什么是重定位?为什么要重定位?

【参考答案】①将用户程序的相对地址(逻辑地址)转换为绝对地址(物理地址)的过程,称为重定位。②由于在多道程序环境下,多个目标模块的起始地址通常是0,程序中的其他地址都是相对于0这一起始地址而言的。采用重定位,可根据内存的当前地址使用情况,将装入模块装入内存的适当位置,并确定装入的物理地址,以保证程序运行时存取指令或数据地址的正确。

4. 动态重定位的实现方式有哪几种?

【参考答案】动态重定位的实现必须有硬件地址转换机构的支持,其具体实现方式主要有以下2种。

(1)连续分配方式下的动态重定位。该方式须在整个系统中设置一个重定位寄存器,用来存放正在执行的作业在内存中的起始地址。当CPU要存取指令或数据时,硬件地址转换机构会自动将逻辑地址与重定位寄存器的值相加,形成指令或数据的物理地址。

(2)离散分配方式下的动态重定位。离散分配方式主要是指分页和分段存储管理方式。重定位过程为:①系统首先必须为每个作业配置一张页(段)表,用来记录作业的每个页(段)对应的内存块号(内存起始地址和段长),页(段)表被存放在内存中;②整个系统须设置一个页(段)表控制寄存器,用来存放正在执行的作业的页(段)表起始地址和长度;③当CPU要存取指令或数据时,硬件的地址转换机构会自动将逻辑地址分成页号和页内地址两部分(或

直接从逻辑地址中获得段号），根据页（段）号到控制寄存器所指示的页（段）表中获得对应的物理块号（或段的内存起始地址），并与页（段）的地址相加，最终形成物理地址。

5. 可采用哪几种方式将程序装入内存？它们分别适用于何种场合？

【参考答案】将程序装入内存可采用的方式有：绝对装入方式、重定位装入方式、动态运行时装入方式。绝对装入方式适用于单道程序环境中，重定位装入方式和动态运行时装入方式适用于多道程序环境中。

6. 何谓静态链接？静态链接时需要解决哪两个问题？

【参考答案】静态链接是指在程序运行之前，先将各目标模块及它们所需的库函数链接成一个完整的装入模块，以后不再拆开的链接方式。将几个目标模块链接成一个装入模块时，须解决以下2个问题。

- (1) 修改相对地址，即将除第一个模块外的相对地址修改成装入模块中相应的相对地址。
- (2) 变换外部调用符号，即将每个模块中所用的外部调用符号变换为相对地址。

7. (考研真题) 编写程序时，源代码必须经过编译和链接生成目标代码，请问什么是链接？链接主要解决了什么问题？简述链接的主要类型及其优缺点。

【参考答案】链接是指由链接程序将编译后形成的一组目标模块以及所需库函数链接在一起，进而形成一个完整的装入模块。

链接程序按各个模块的相对地址依次构成统一的从0号单元开始编址的逻辑地址空间。链接主要有3种类型。

(1) 静态链接：在程序运行之前，先将各目标模块及它们所需的库函数链接成一个完整的可执行程序，以后不再拆开。优点：适用范围比较广，无须担心用户机器缺少某个库函数。缺点：①修改或更新某个目标模块时需要重新打开装入模块，这不仅涉及效率问题，而且在很多时候甚至是不可能实现的；②在静态链接中，每个模块必含有目标模块的复制，无法实现共享。

(2) 装入时动态链接：将用户程序编译后所得到的一组目标模块，采用边装入边链接的方式装入内存。优点：①便于修改和更新，针对动态链接方式，由于各目标模块是分开存放的，因此非常容易修改或更新各目标模块；②便于实现对目标模块的共享，针对装入时链接方式，很容易将一个目标模块链接到几个应用模块上，实现多个应用模块对目标模块的共享。缺点：由于应用程序事先无法确定本次要运行的模块，因此该程序的所有模块要全部装入内存，并在装入时链接在一起。有部分目标模块不会被运行，但也要装入，低效且浪费空间。

(3) 运行时动态链接：将某些目标模块的链接推迟到程序执行时才进行。优点：在运行时动态链接过程中未用到的目标模块都不会被调入内存和链接，这样不仅能加快程序的装入过程，而且可以大大节省内存空间。

8. 为什么要引入对换？对换可分为哪几种类型？

【参考答案】在多道程序环境下，一方面，在内存中的某些进程会因某事件尚未发生而阻塞，但此时它占用了大量的内存空间，这样，内存中的所有进程都有可能被阻塞，进而发生迫使CPU停止下来等待的情况；另一方面，有许多作业因内存空间不足会一直驻留在外存上，而不能进入内存运行。这不仅浪费系统资源，而且会降低系统吞吐量。为了解决这一问题，在OS中引入了对换（也称交换）技术。

对换可分为整体对换和部分对换两种类型。整体对换是将整个进程换入/换出，主要用于缓

解目前系统中内存不足的这一情况。部分对换是将进程的一部分（页、段）换入/换出，主要用于实现虚拟存储器。

9. 在对换技术中，对文件区管理的目标和对对换空间管理的目标有何不同？

【参考答案】对文件区管理和对对换空间管理的目标分别为：①对文件区管理的主要目标是提高文件存储空间的利用率，然后才是提高对文件的访问速度，因此，对文件区管理采取的是离散分配方式；②对对换空间管理的主要目标是提高进程换入/换出的速度，然后才是提高文件存储空间的利用率，因此，对对换空间管理采取的是连续分配方式，较少考虑外存中的碎片问题。

10. 为什么说分段系统较分页系统更易实现信息共享与保护？

【参考答案】①对于分页系统，每个页面是分散存储的，为了实现信息共享与保护，页面之间需要一一对应起来，为此需要建立大量的页表项。②对于分段系统，每个段都从0开始编址，并采用一段连续的地址空间，这样在实现信息共享与保护时，只须为所要共享与保护的程序设置一个段表项，将其中的起始地址与内存一一对应起来即可。

11. 提高内存利用率的途径主要有哪些？

【参考答案】内存利用率不高，主要表现为4种形式：①内存中存在大量的、分散的、难以利用的碎片；②暂时或长期不能运行的程序和数据占据了大量的存储空间；③当作业较大时，内存中只能装入少量作业，当它们被阻塞时，将使CPU空闲，从而降低了内存的利用率；④内存中存在着重复的复制。

针对上述问题，可采用下述方法提高内存的利用率：①将连续分配方式改为离散分配方式，以减少内存中的碎片；②增加对换机制，将暂时不能运行的进程或暂时不需要的程序和数据换出至外存，以腾出内存来装入可运行的进程；③引入动态链接机制，当运行中的程序需要调用某段程序时，才将该段程序由外装入内存；④引入虚拟存储器机制，使更多的作业能装入内存，提高了CPU的利用率，也可避免装入本次运行中不会用到的那部分程序和数据；⑤引入存储器共享机制，允许一个正文段或数据段被若干个进程共享，以减少内存中重复的复制。

5.2 计算题参考答案

12. (考研真题) 假设一个分页存储系统具有快表，多数活动页表项都可以存在于其中。若页表放在内存中，内存访问时间是 1ns，快表的命中率是 85%，快表的访问时间为 0.1ns，则有效存取时间为多少？

【参考答案】当快表的命中率为85%时，执行一次访问内存操作（有效存取）需要的时间为 $T=0.1 \times 85\% + 1 \times (1-85\%) + 1 = 1.235\text{ns}$ 。

13. 对一个将页表存放在内存中的分页系统：

(1) 如果访问内存需要 0.2 μs ，则有效访问时间为多少？

(2) 如果加一快表，且假定在快表中找到页表项的概率高达 90%，则有效访问时间又是多少（假定查快表须花费的时间为 0）？

【参考答案】

(1) 有效访问时间为 $2 \times 0.2 = 0.4\mu\text{s}$ 。

(2) 有效访问时间为 $90\% \times 0.2 + (1-90\%) \times 2 \times 0.2 = 0.22\mu\text{s}$ 。

14. 某系统采用分页存储管理方式, 拥有逻辑空间 32 页, 每页 2KB; 拥有物理空间 1MB。

(1) 写出逻辑地址的格式。

(2) 若不考虑访问权限等, 则进程的页表有多少项? 每项至少有多少位?

(3) 如果物理空间减少一半, 则页表结构应相应地做怎样的改变?

【参考答案】

(1) 该系统拥有逻辑空间32页, 故逻辑地址中页号必须用5位来描述。而每页为2KB, 因此, 页内地址必须用 11位来描述, 这样即可得到它的逻辑地址格式如下。



(2) 每个进程最多有32个页面, 因此, 进程的页表项最多为32项。若不考虑访问权限等, 则页表项中只须给出页所对应的物理块块号, 1MB的物理空间可分成 2^9 个内存块, 故每个页表项至少有9位。

(3) 如果物理空间减少一半, 则页表中的页表项数仍不变, 但每项的长度可减少1位。

15. 已知某分页系统, 内存容量为 64KB, 页面大小为 1KB, 对一个 4 页大的作业, 其 0、1、2、3 页分别被分配到内存的 2、4、6、7 块中。

(1) 将十进制的逻辑地址 1 023、2 500、3 500、4 500 转换为物理地址。

(2) 以十进制的逻辑地址 1 023 为例, 画出地址转换过程图。

【参考答案】

(1) 对上述逻辑地址, 可首先计算出它们的页号和页内地址(逻辑地址除以页面大小, 得到的商为页号, 余数为页内地址), 然后通过页表将其转换成对应的物理地址。

① 逻辑地址1 023。 $\lfloor 1\ 023/1K \rfloor = 0$, $1\ 023 \% 1K = 1\ 023$, 因此页号为0, 页内地址为1 023, 查页表找到对应的物理块号为2, 故物理地址为 $2 \times 1K + 1\ 023 = 3\ 071$ 。

② 逻辑地址2 500。 $\lfloor 2\ 500/1K \rfloor = 2$, $2\ 500 \% 1K = 452$, 因此页号为2, 页内地址为452, 查页表找到对应的物理块号为6, 故物理地址为 $6 \times 1K + 452 = 6\ 596$ 。

③ 逻辑地址3 500。 $\lfloor 3\ 500/1K \rfloor = 3$, $3\ 500 \% 1K = 428$, 因此页号为3, 页内地址为428, 查页表找到对应的物理块号为7, 故物理地址为 $7 \times 1K + 428 = 7\ 596$ 。

④ 逻辑地址4 500。 $\lfloor 4\ 500/1K \rfloor = 4$, $4\ 500 \% 1K = 404$, 因此页号为4, 页内地址为404, 因页号大于页表长度, 故产生越界中断。

(2) 逻辑地址1 023的地址转换过程如图1-5-1所示。

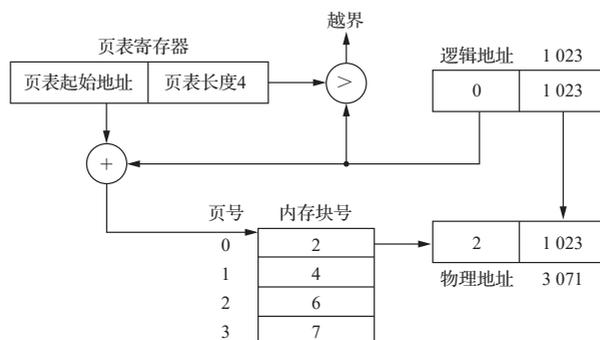


图 1-5-1 地址转换过程图

16. (考研真题) 已知某系统页面大小为 4KB, 每个页表项的大小为 4B, 采用多层分页策略映射 64 位的用户地址空间。若限定最高层页表只占 1 页, 问它可采用几层分页策略。

【参考答案】由题意可知, 该系统的用户地址空间为 2^{64} B, 而页的大小为4KB, 故作业最多可有 $2^{64}/2^{12}=2^{52}$ 个页, 其页表的大小则为 $2^{52} \times 4=2^{54}$ B。因此, 又可将页表分成 2^{42} 个页表项, 并为它建立二级页表, 二级页表的大小为 2^{44} B。依次类推, 可知道它的3、4、5、6级页表的长度分别是 2^{34} B、 2^{24} B、 2^{14} B、 2^4 B, 故必须采取6层分页策略。

17. 对于表 1-5-1 所示的段表, 请将逻辑地址 (0,137), (1,4 000), (2,3 600), (5,230) 转换成物理地址。

表1-5-1 段表

| 段号 | 内存起始地址 | 段长 |
|----|--------|-----|
| 0 | 50K | 10K |
| 1 | 60K | 3K |
| 2 | 70K | 5K |
| 3 | 120K | 8K |
| 4 | 150K | 4K |

【参考答案】本问题分步解答如下。

(1) 段号0小于段表长度5, 故段号合法; 由段表的第0项可获得段的内存起始地址为50K, 段长为10K; 由于段内地址137小于段长10K, 故段内地址也合法, 因此可得出对应的物理地址为 $50K+137=51\ 337$ 。

(2) 段号1小于段表长度, 故段号合法; 由段表的第1项可获得段的内存起始地址为60K, 段长为3K; 经检查, 段内地址4000超过了段长3K, 因此产生越界中断。

(3) 段号2小于段表长度, 故段号合法; 由段表的第2项可获得段的内存起始地址为70K, 段长为5K; 故段内地址3600也合法。因此, 可得出对应的物理地址为 $70K+3\ 600=75\ 280$ 。

(4) 段号5等于段表长度, 故段号不合法, 产生越界中断。

5.3 综合应用题参考答案

18. (考研真题) 某系统采用动态分区分配方式管理内存, 内存空间为 640KB, 低端 40KB 存放 OS。系统为用户作业分配空间时, 从低地址区开始。对下列作业请求序列, 画图表示使用首次适应算法进行内存分配和回收后内存的最终映像。作业请求序列如下:

作业 1 申请 200KB, 作业 2 申请 70KB ;

作业 3 申请 150KB, 作业 2 释放 70KB ;

作业 4 申请 80KB, 作业 3 释放 150KB ;

作业 5 申请 100KB, 作业 6 申请 60KB ;

作业 7 申请 50KB, 作业 6 释放 60KB。

【参考答案】首次适应算法的思想是把空闲分区按照地址递增的顺序组成一个链表, 为进程分配内存时从链首开始查找, 直至找到能容纳进程的分区。本题使用首次适应算法进行内存分配与回收后, 具体过程如下。

- (1) 初始时OS存放在0K ~ 40K, 40K ~ 600K为空。
- (2) 作业1申请200K, 作业2申请70K: OS存放在0K ~ 40K, 作业1存放在40K ~ 240K, 作业2存放在240K ~ 310K, 310K ~ 600K为空。
- (3) 作业3申请150K, 作业2释放70K: OS存放在0K ~ 40K, 作业1存放在40K ~ 240K, 240K ~ 310K为空, 作业3存放在310K ~ 460K, 460K ~ 600K为空。
- (4) 作业4申请80K, 作业3释放150K: OS存放在0K ~ 40K, 作业1存放在40K ~ 240K, 240K ~ 460K为空, 作业4存放在460K ~ 540K, 540K ~ 600K为空。
- (5) 作业5申请100K, 作业6申请60K: OS存放在0K ~ 40K, 作业1存放在40K ~ 240K, 作业5存放在240K ~ 340K, 作业6存放在340K ~ 400K, 400K ~ 460K为空, 作业4存放在460K ~ 540K, 540K ~ 600K为空。
- (6) 作业7申请50K, 作业6释放60K: OS存放在0K ~ 40K, 作业1存放在40K ~ 240K, 作业5存放在240K ~ 340K为空, 作业7存放在400K ~ 450K, 450K ~ 460K为空, 作业4存放在460K ~ 540K, 540K ~ 600K为空。

因此, 内存的最终映像如图1-5-2所示。

19. 某 OS 采用分段存储管理方式, 用户区内存为 512KB, 空闲块链入空闲块表, 分配时截取空闲块的前半部分 (低地址部分), 初始时全部空闲。执行申请、释放操作序列 request (300KB)、request (100KB)、release (300KB)、request (150KB)、request (50KB)、request (90KB) 后 :

- (1) 若采用首次适应算法, 则空闲块表中有哪些空闲块 (指出大小及起始地址)?
- (2) 若采用最佳适应算法, 则空闲块表中有哪些空闲块 (指出大小及起始地址)?
- (3) 若随后又要申请 80KB, 则针对上述两种情况会产生什么后果? 这说明了什么问题?

【参考答案】

(1) 采用首次适应算法时的内存分配情况如图1-5-3 (a) 所示。内存中有两个空闲块。第1块: 起始地址290K, 大小10KB。第2块: 起始地址400K, 大小112KB。

(2) 采用最佳适应算法时的内存分配情况如图1-5-3 (b) 所示。内存中有两个空闲块。第1块: 起始地址240K, 大小60KB。第2块: 起始地址450K, 大小62KB。

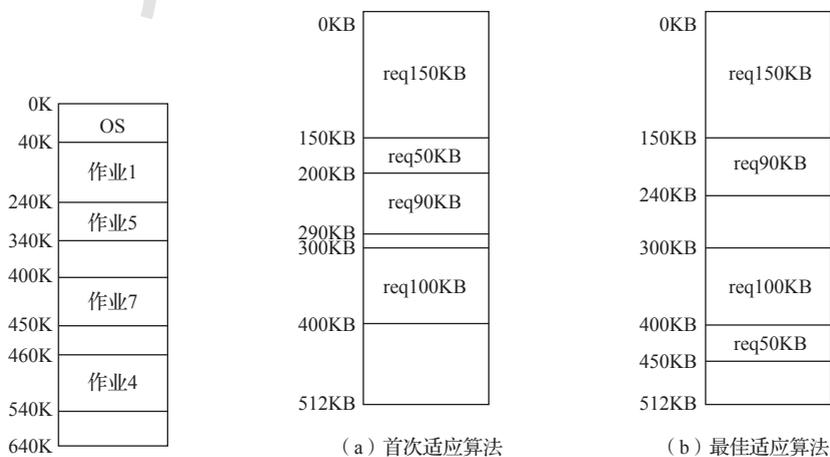


图 1-5-2 内存最终映像

图 1-5-3 内存分配情况

(3) 若随后又要申请80KB, 则首次适应算法可以分配成功, 而最佳适应算法没有足够大的空闲区以供分配。这说明首次适应算法尽可能地使用了低地址部分空闲区域, 而留下了高地址部分的大空闲区, 因此更有可能满足进程的请求。

20. 某系统的空闲分区如表 1-5-2 所示, 采用可变分区分配策略处理作业。现有作业序列 96KB、20KB、200KB, 若采用首次适应算法和最佳适应算法来处理这些作业序列, 则哪种算法能满足该作业序列的请求? 为什么?

表1-5-2 空闲分区表

| 分区号 | 分区大小 | 分区起始地址 |
|-----|-------|--------|
| 1 | 32KB | 100K |
| 2 | 10KB | 150K |
| 3 | 5KB | 200K |
| 4 | 218KB | 220K |
| 5 | 96KB | 530K |

【参考答案】本问题分步解答如下。

(1) 采用首次适应算法时, 96KB作业进入4号空闲分区, 20KB作业进入1号空闲分区, 此时空闲分区如表1-5-3所示。

表1-5-3 采用首次适应算法时的空闲分区表

| 分区号 | 分区大小 | 分区起始地址 |
|-----|-------|--------|
| 1 | 12KB | 120K |
| 2 | 10KB | 150K |
| 3 | 5KB | 200K |
| 4 | 122KB | 316K |
| 5 | 96KB | 530K |

此时再无空闲分区可以满足200KB大小的作业, 因此该作业序列的请求无法被满足。

(2) 采用最佳适应算法时, 作业序列分别进入5、1、4号空闲分区, 可以满足该作业序列的请求。分配处理之后的空闲分区如表1-5-4所示。

表1-5-4 采用最佳适应算法时的空闲分区表

| 分区号 | 分区大小 | 分区起始地址 |
|-----|------|--------|
| 1 | 12KB | 120K |
| 2 | 10KB | 150K |
| 3 | 5KB | 200K |
| 4 | 18KB | 420K |

第6章 虚拟存储器

6.1 简答题参考答案

1. 常规存储器管理方式具有哪两大特征？它们对系统性能有何影响？

【参考答案】常规存储器管理方式具有一次性和驻留性这两大特征。一次性指进程必须全部装入内存；驻留性指在程序运行过程中，进程全部驻留在内存中。一次性对内存空间的浪费非常大；驻留性使内存中暂时不用的程序或数据无法被释放。

2. 什么是虚拟存储器？如何实现页式虚拟存储器？

【参考答案】虚拟存储器是具有请求调入功能和置换功能，能从逻辑上对内存容量进行扩充的一种存储系统。从用户观点看，虚拟存储器具有比实际内存大得多的容量，其逻辑容量由逻辑地址结构以及内存和外存容量之和决定，其运行速度接近于内存的存取速度，而每位成本却又接近于外存。

为实现虚拟存储器，首先需要扩充页表，增加状态位以指出所需页是否在内存中，增加外存起始地址以便调入页面，增加引用位以供置换算法使用，增加修改位以减少换出时写入磁盘的次数。另外，还要使用两种关键技术。①请求调页技术。该项技术是指及时将进程所要访问的不在内存中的页调入内存。该功能是由硬件（缺页中断机构发现缺页）和软件（将所需页调入内存）配合实现的。②置换页技术。该项技术是指当内存中已无足够空间用于装入即将调入的页时，为了保证进程能继续运行，系统必须换出内存中的部分页，以腾出足够的空间。具体的置换操作并不复杂，其关键是确定应将哪些页换出，即采取什么置换算法。

3. “整体对换从逻辑上也扩充了内存，因此也实现了虚拟存储器的功能”这种说法是否正确，请说明理由。

【参考答案】这种说法是错误的。整体对换将内存中暂时不用的某个程序及其数据换出至外存，腾出足够的内存空间以装入在外存中具备运行条件的进程所对应的程序和数据。虚拟存储器是指仅把作业的一部分装入内存便可运行作业的存储器系统，亦指具有请求调入功能和置换功能、能从逻辑上对内存容量进行扩充的存储器系统，它的实现必须建立在离散分配的基础上。虽然整体对换和虚拟存储器均能从逻辑上扩充内存空间，但整体对换不具备离散性。实际上，在具有整体对换功能的系统中，进程的大小仍将受到实际内存容量的限制。

4. 在请求分页系统中，为什么说一条指令执行期间可能产生多次缺页中断？

【参考答案】在请求调页时，只要作业的部分页在内存中，该作业就能执行；而在执行过程中发现所要访问的指令或数据不在内存中时，系统会产生缺页中断，将所需的页面调入内存。在请求调页系统中，一条指令（如copy A to B）可能跨了两个页，而其中要访问的操作数可能与指令不在同一个页上，且操作数本身也可能跨了两个页。当要执行这类指令而相应的页又都不在内存中时，就将产生多次缺页中断。

5. 试比较缺页中断与一般的中断，它们之间有何明显区别？

【参考答案】缺页中断与一般中断的区别主要有：①一般中断只需要保护现场然后即可直接跳到须及时处理的地方；②缺页中断除了需要保护现场外，还需要判断内存中是否有足够的空间来存储所需的页或段，然后再把所需的页或段调进来使用。

6. 试说明在请求分页系统中页面的调入过程。

【参考答案】每当程序所要访问的页面未在内存时（存在位为“0”），便向CPU发出缺页中断，中断处理程序保存CPU现场信息，分析中断原因后，转入缺页中断处理程序。该程序通过查找页表得到该页在外存的物理地址，如果此时内存能容纳新页，则启动磁盘，将所缺页面调入内存，然后修改页表。如果内存已满，则须按照某种置换算法进行页面置换；如果换出页面被修改过（修改位为“1”），则必须将它写回磁盘，再把缺页调入内存，将页表中调入页的存在位改为“1”，并将此页表项写入快表中。利用修改后的页表，形成要访问数据的物理地址，再去访问内存数据。整个页面调入过程对用户是透明的。

7. (考研真题) 简述在具有快表的请求分页系统中，将逻辑地址转换为物理地址的完整过程。

【参考答案】在具有快表的请求分页系统中，将逻辑地址转换为物理地址的完整过程为：①检索快表，试图从中找出所要访问的页；②如果找到，那么修改页表项中的访问位，供置换算法选择淘汰页时参考，将写指令的修改位置1，然后利用页表项中给出的物理块号和页内地址形成物理地址，地址转换结束；③如果没找到，那么应到内存中查找页表，再根据查找到的页表项中的状态位来判断该页是否已调入内存。若该页已调入内存，则将该页的页表项写入快表。当快表已满时，先调出按某种算法所确定的页的页表项，再写入该页的页表项。若该页未调入内存，则产生缺页中断，请求OS从外存中将该页调入内存，再转到步骤②进行地址转换。

8. 何谓固定分配局部置换和可变分配全局置换的内存分配策略？

【参考答案】

(1) 固定分配局部置换：固定分配是指，为每个进程分配一组固定数目的物理块，在进程运行期间不再改变；局部置换是指，如果进程在运行中发现缺页，则只能从分配给该进程的 n 个页面中选出1页换出，然后再调入1页。

(2) 可变分配全局置换：可变分配是指，先为每个进程分配一定数目的物理块，在进程运行期间，可根据情况做适当的改变；全局置换是指，如果进程在运行中发现缺页，则将OS所保留的空闲物理块或者所有进程的全部物理块选择1块换出然后再将缺页调入。

9. 实现 LRU (least recently used, 最近最久未使用) 页面置换算法所需要的硬件支持是什么？

【参考答案】需要寄存器和栈等硬件支持。寄存器用于记录某进程在内存中各页的使用情况；栈用于保存当前使用的各页的页面号。

10. 什么是“抖动”？产生“抖动”的原因是什么？

【参考答案】①“抖动”是指刚被换出的页很快被访问，须重新调入，因此须再选一页调出，而此时被换出的页很快又要被访问，因而又须将它调入，如此频繁地更换页面，使得系统把大部分时间用在了页面的换进/换出上，而几乎不能完成任何有效的工作，我们称这一现象为“抖动”。②产生“抖动”的根本原因是同时在系统中运行的进程太多，分配给每个进程的物理块数太少，其不能满足进程正常运行的基本要求，致使每个进程在运行时频繁缺页。

11. 何谓工作集? 它是基于什么原理而确定的?

【参考答案】①工作集(或驻留集)是指在某段时间间隔内进程实际要访问的页面集合。经常被使用的页面需要存在于工作集中,而长期不被使用的页面则要从工作集中丢弃。②确定工作集的原理是:基于程序运行时的局部性原理,程序在运行期间对页面的访问在一段时间内局限于较少的页面,这些页面称为活跃页面,页面集合即为工作集。如果能预先将活跃页面调入内存,则将大大降低缺页率。因此须使OS跟踪每个进程的工作集,并为进程分配大于其工作集的物理块。如果还有空闲物理块,则可以再调一个进程到内存以增加多道程序数。如果所有工作集之和超过了可用物理块的总数,那么OS会暂停一个进程,将其页面调出并且将其物理块分配给其他进程,以防止出现抖动现象。正确选择工作集的大小,对存储器利用率和系统吞吐量的提高具有重要影响。

12. 为了实现请求分段存储管理,应在系统中增加配置哪些硬件机构?

【参考答案】为了实现请求分段存储管理,应在系统中配置多种硬件机构,以支持快速完成请求分段功能。所需的硬件支持有段表机制、缺段中断机构以及地址转换机构。

6.2 计算题参考答案

13. (考研真题)某虚拟存储器的用户空间共有32个页面,每页1KB,内存16KB。假定某时刻系统为用户的第0、1、2、3页分配的物理块号分别为5、10、4、7,而该用户作业的长度为6页,试将十六进制逻辑地址0A5C、103C、1A5C转换成物理地址。

【参考答案】由题目所给条件可知,该系统的逻辑地址有15位,其中高5位为页号,低10位为页内地址;物理地址有14位,其中高4位为块号,低10位为块内地址。另外,由于题目中给出的逻辑地址是十六进制数,故可先将其转换成二进制数以直接获得页号和页内地址,再完成地址的转换。

(1)逻辑地址0A5CH转化为二进制为000 1010 0101 1100,页号为0 0010,即2号页,页号合法。从页表中找到对应的内存物理块号为4,即0100;与页内地址10 0101 1100拼接而形成物理地址01 0010 0101 1100,即125CH。

(2)逻辑地址103CH的页号为4,页号合法,但该页未装入内存,故产生缺页中断。

(3)逻辑地址1A5CH的页号为6,页号非法,故产生越界中断。

14. 某请求调页系统,页表保存在寄存器中。若一个被替换的页未被修改过,则处理一个缺页中断需要8ms;若被替换的页已被修改过,则处理一个缺页中断需要20ms。内存存取时间为 $1\mu\text{s}$,访问页表的时间可忽略不计。假定70%被替换的页被修改过,为保证有效存取时间不超过 $2\mu\text{s}$,可接受的最大缺页率是多少?

【参考答案】如果用 p 表示缺页率,则有效访问时间不超过 $2\mu\text{s}$ 可表示为:

$$(1-p) \times 1\mu\text{s} + p \times (70\% \times 20\text{ms} + 30\% \times 8\text{ms} + 1\mu\text{s}) \leq 2\mu\text{s}$$

因此可计算出: $p \leq 1/16400 \approx 0.00006$,即可接受的最大缺页率为0.00006。

15. (考研真题)某分页式虚拟存储系统,用于页面交换的磁盘的平均访问与传输时间是20ms,页表保存在内存中,访问时间为 $1\mu\text{s}$,即每引用一次指令或数据就需要访问内存2次。为改善性能,可以增设一个联想寄存器,若页表项在联想寄存器中,则只要访问1次内存。假设80%的访问对应的页表项在联想寄存器中,剩下的20%中有10%的访问(即总数的2%)会产生缺页。请计算有效访问时间。

【参考答案】本题分步解答如下。

(1) 80%的访问的页表项在联想寄存器中, 访问耗时 $1\mu\text{s}$ 。

(2) 18%的访问的页表项不在联想寄存器中, 但在内存中, 访问耗时 $(1\mu\text{s}+1\mu\text{s})$ 。

(3) 2%的访问产生缺页中断, 访问耗时 $(1\mu\text{s}+1\mu\text{s}+20\text{ms}+1\mu\text{s})$ 。

因此有效访问时间为 $80\% \times 1 + (1-80\%) \times [(1-10\%) \times 1 \times 2] + 2\% \times (1 \times 3 + 20 \times 1000) = 401.22\mu\text{s}$

16. 假定某 OS 存储器采用页式存储管理, 一个进程在快表中的页表项如表 1-6-1 所示, 在内存中的页表项如表 1-6-2 所示。

表1-6-1 快表中的页表项

| 页号 | 页帧号 |
|----|-----|
| 0 | f1 |
| 1 | f2 |
| 2 | f3 |
| 3 | f4 |

表1-6-2 内存中的页表项

| 页号 | 页帧号 |
|----|-----|
| 4 | f5 |
| 5 | f6 |
| 6 | f7 |
| 7 | f8 |
| 8 | f9 |
| 9 | f10 |

注: 只列出不在快表中的页表项。

假定该进程长度为 320B, 每页 32B。现有逻辑地址 101、204、576 (八进制), 若这些逻辑地址能转换成物理地址, 则说明转换的过程, 并指出具体的物理地址; 若不能转换, 则说明原因。

【参考答案】1页的大小为32B, 逻辑地址结构为: 低5位为页内位移, 其余高位为页号。

101 (八进制) = 001 000 001 (二进制), 因此页号为2, 在联想寄存器中, 对应的块号为 f3, 即物理地址为 (f3, 1)。

204 (八进制) = 010 000 100 (二进制), 因此页号为4, 不在联想寄存器中, 查找内存的页表得块号为 f5, 即物理地址为 (f5, 4), 用其更新联想寄存器。

576 (八进制) = 101 111 110 (二进制), 因此页号为11, 已超出页表范围, 即产生越界中断。

17. 有一个矩阵 $\text{int } A [100,100]$ 以行优先方式进行存储。计算机采用虚拟存储系统, 物理内存共有 3 页, 其中 1 页用来存放程序, 其余 2 页用来存放数据。假设程序已在内存中占了 1 页, 其余 2 页空闲。若每页可存放 200 个整数, 则程序 1、程序 2 执行的过程中各会发生多少次缺页? 当每页只能存放 100 个整数时, 又会发生多少次缺页? 以上结果说明了什么问题?

程序 1 :

```
for(i=0;i<100;i++)
    for(j=0;j<100;j++)
        A[i,j]=0;
```

程序 2 :

```
for(j=0;j<100;j++)
    for(i=0;i<100;i++)
        A[i,j]=0;
```

【参考答案】本题分步解答如下。

(1) 程序1按行优先的顺序访问数组元素, 与数组在内存中存放的顺序一致, 每个内存页面可存放200个数组元素。这样, 程序1每访问两行数组元素就会产生1次缺页中断, 因此程序1在执行过程中会发生50次缺页。

(2) 程序2按列优先的顺序访问数组元素, 由于每个内存页面存放两行数组元素, 因此程

序2每访问两个数组元素就会产生1次缺页中断，整个执行过程会发生5000次缺页。

(3) 若每页只能存放100个整数，则每页仅能存放一行数组元素，同理可以计算出：程序1在执行过程中产生100次缺页；程序2在执行过程中产生10 000次缺页。

(4) 以上结果说明：缺页的次数与内存中数据存放的方式及程序执行的顺序有很大关系。同时说明：当缺页中断次数不多时，减小页面大小影响并不大；但当缺页中断次数很多时，减小页面大小会带来很严重的影响。

6.3 综合应用题参考答案

18. (考研真题) 有一个请求分页式虚拟存储器系统，分配给某进程3个物理块，开始时内存中预装入第1,2,3个页面，该进程的页面访问序列为1, 2, 4, 2, 6, 2, 1, 5, 6, 1。

(1) 若采用OPT(optimal, 最佳)页面置换算法，则访问过程发生的缺页率为多少？

(2) 若采用LRU页面置换算法，则访问过程中的缺页率为多少？

【参考答案】

(1) 采用OPT页面置换算法时，因为内存中预装入了前3个页面，所以前3个页面不发生缺页。访问过程中发生了2次缺页，缺页率 $f=2/10 \times 100\%=20\%$ 。页面置换走向如图1-6-1所示。

| | | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|---|
| 访问串 | 1 | 2 | 4 | 2 | 6 | 2 | 1 | 5 | 6 | 1 |
| 内存 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | | 2 | 2 | 2 | 2 | 2 | 2 | 5 | 5 | 5 |
| | | | 4 | 4 | 6 | 6 | 6 | 6 | 6 | 6 |
| 缺页 | | | | | √ | | | √ | | |

图1-6-1 采用OPT页面置换算法时的页面置换走向

(2) 采用LRU页面置换算法时，访问过程中发生了4次缺页，缺页率 $f=4/10 \times 100\%=40\%$ 。页面置换走向如图1-6-2所示。

| | | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|---|
| 访问串 | 1 | 2 | 4 | 2 | 6 | 2 | 1 | 5 | 6 | 1 |
| 内存 | 1 | 1 | 1 | 1 | 6 | 6 | 6 | 5 | 5 | 5 |
| | | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 6 | 6 |
| | | | 4 | 4 | 4 | 4 | 1 | 1 | 1 | 1 |
| 缺页 | | | | | √ | | √ | √ | √ | |

图1-6-2 采用LRU页面置换算法时的页面置换走向

19. 进程已分配到4个块，如表1-6-3所示(页号为十进制，从0开始编号)。当进程访问第4页时，产生缺页中断，请分别用FIFO(first in first out, 先进先出)页面置换算法和LRU页面置换算法决定缺页中断处理程序选择换出的页面。

表1-6-3 页表

| 块号 | 页号 | 装入时间 | 最近访问时间 | 访问位 | 修改位 |
|----|----|------|--------|-----|-----|
| 2 | 0 | 60 | 161 | 0 | 1 |
| 1 | 1 | 130 | 160 | 0 | 0 |
| 0 | 2 | 26 | 162 | 1 | 0 |
| 3 | 3 | 20 | 163 | 1 | 1 |

【参考答案】

(1) FIFO页面置换算法：按照先进先出规则，最先进入的页面应最先被换出，因此访问第4页时，缺页中断处理程序应选择的是3号页。由于该页的修改位是1，在换出内存后应先进行回写，即重新保存。

(2) LRU页面置换算法：最近一次访问时间离当前最远的页面应被选择换出，因此缺页中断处理程序应选择的是1号页。

20. 某系统有4个页，某个进程的页面使用情况如表1-6-4所示，问采用FIFO、LRU、简单Clock和改进型Clock页面置换算法，分别会置换哪一页？

表1-6-4 页面使用情况

| 页号 | 装入时间 | 上次引用时间 | R (读) | M (修改) |
|----|------|--------|-------|--------|
| 0 | 126 | 279 | 0 | 0 |
| 1 | 230 | 260 | 1 | 0 |
| 2 | 120 | 272 | 1 | 1 |
| 3 | 160 | 280 | 1 | 1 |

【参考答案】

(1) FIFO页面置换算法选择最先进入内存的页面进行替换。由表1-6-4中所示装入时间可知，2号页最先进入内存，因此FIFO页面置换算法将选择2号页替换。

(2) LRU页面置换算法选择最近最长时间未使用的页面进行替换。由表1-6-4中所示的上次引用时间可知，1号页是最近最长时间未使用的页面，因此LRU页面置换算法将选择1号页替换。

(3) 简单Clock页面置换算法从上一次的位置开始扫描，选择第一个访问位为0的页面进行替换。由表1-6-4中的R(读)标志位可知，按装入顺序依次扫描，页面0未被访问，扫描结束，因此简单Clock页面置换算法将选择0号页替换。

(4) 改进型Clock页面置换算法从上一次的位置开始扫描，首先寻找未被访问和修改的页面。由表1-6-4中的R(读)标志位和M(修改)标志位可知，只有页面0满足R=0和M=0，因此改进型Clock页面置换算法将选择0号页替换。

21. (考研真题) 请求分页管理系统中，假设某进程的页表内容如表1-6-5所示。

表1-6-5 进程页表内容

| 页号 | 页框号 | 有效位 (存在位) |
|----|------|-----------|
| 0 | 101H | 1 |
| 1 | — | 0 |
| 2 | 254H | 1 |

页面大小为4KB，一次内存的访问时间是100ns，一次快表的访问时间是10ns，处理一次缺页的平均时间是 10^8 ns(已含更新快表和页表的时间)，进程的驻留集大小固定为2，采用LRU页面置换算法和局部淘汰策略。假设：①快表初始为空；②地址转换时先访问快表，若快表未命中，则再访问页表(忽略访问页表之后的快表更新时间)；③有效位为0表示页面不在内存中，产生缺页中断，缺页中断被处理后，返回产生缺页中断的指令处重新

执行。设有虚地址访问序列 2362H、1565H、25A5H，请问：

- (1) 依次访问上述 3 个虚地址，各需要多少时间？给出计算过程。
- (2) 基于上述访问序列，虚地址 1565H 的物理地址是多少？请说明理由。

【参考答案】

(1) 页面大小为 4KB=2¹²B，故页内偏移量为 12 位，占虚地址的低 12 位，高 4 位为页号。虚地址 2362H 的页号为 2，页内地址为 362H。先访问快表（10ns），未命中；再去访问页表（100ns），获得对应的物理块号 254H，其与页内地址 362H 拼接成物理地址 254362H，同时将第 2 页的信息装入快表中；最后根据这个物理地址访问内存（100ns）。因此，访问到虚地址对应单元的数据总共需要 10ns+100ns+100ns=210ns。

同理，虚地址 1565H 的页号为 1，页内地址为 565H。先访问快表（10ns），未命中；访问页表（100ns），不在内存中，产生缺页中断，中断处理时间为 10⁸ns，然后返回产生缺页中断的指令处重新执行，并以此更新快表和页表；访问快表（10ns）便可得到对应的物理地址，根据物理地址访问内存（100ns）。总共需要 10ns+100ns+10⁸ns+10ns+100ns≈10⁸ns。

同理，虚地址 25A5H 的页号为 2，页内地址为 5A5H。先访问快表（10ns），该页信息已装入快表，获得对应的物理块号 254H，从而拼接成物理地址 2545A5H，根据该地址访问数据（100ns）。总共需要 110ns。

(2) 访问 1565H 时，产生缺页中断，在处理中断时，合法驻留集为 2，内存已满，LRU 页面置换算法将选择淘汰 0 号页，然后将 101H 号页框分配给 1 号页，这样便可得到页对应的物理块号 101H，其与页内地址 565H 拼接成的物理地址为 101565H。

第7章 输入/输出系统

7.1 简答题参考答案

1. 试说明 I/O 系统的基本功能。

【参考答案】I/O 系统的基本功能主要包括：①隐藏物理设备的细节；②保证 OS 与设备无关；③提高处理机和 I/O 设备的利用率；④控制 I/O 设备；⑤确保对设备的正确共享；⑥处理错误。

2. I/O 软件一般分为用户层软件、设备独立性软件、设备驱动程序和中断处理程序这 4 个层次，它们的基本功能分别是什么？请说明下列工作分别是在哪一层完成的？

- (1) 向设备寄存器写命令。
- (2) 检查用户是否有权使用设备。
- (3) 将二进制整数转换成 ASC II 的格式打印。
- (4) 缓冲管理。

【参考答案】

- (1) 向设备寄存器写命令是在设备驱动程序中完成的。

(2) 检查用户是否有权使用设备,属于设备保护,因此其在设备独立性软件中完成。

(3) 将二进制整数转换成ASCII格式打印是通过I/O库函数完成的,如C语言的库函数printf中就有打印格式的控制字符串,因此其在用户层软件中完成。

(4) 缓冲管理属于I/O的公有操作,因此是在设备独立性软件中完成的。

3. 设备控制器由哪几部分组成?为了实现CPU与设备控制器之间的通信,设备控制器应具备哪些功能?

【参考答案】①设备控制器的组成部分主要包括设备控制器与CPU的接口、设备控制器与设备的接口、I/O逻辑这三部分。②设备控制器应具备接收和识别命令、交换数据、标志和报告设备的状态、缓冲地址、识别数据、控制差错等功能。

4. (考研真题)什么是通道?通道经常采用图1-7-1所示的交叉连接方式,为什么?

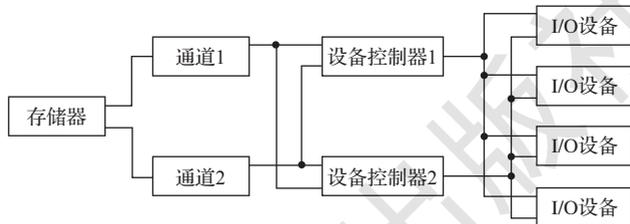


图1-7-1 通道交叉连接图

【参考答案】①通道是一种特殊的处理机,它具有执行I/O指令的能力,并且可以通过执行通道I/O程序来控制I/O操作。②图1-7-1中的交叉连接主要是为了解决通道的瓶颈问题。之所以会产生瓶颈问题,是因为通道价格昂贵,设置的通道数量较少,导致系统吞吐量下降。交叉连接的多通路方式不仅解决了瓶颈问题,还提高了系统可靠性。

5. 设备中断处理程序通常须完成哪些工作?它对中断进行处理的过程包含哪些步骤?

【参考答案】设备中断处理程序要完成:①唤醒被阻塞的驱动进程;②保存被中断进程的CPU现场信息;③转入相应的设备处理程序;④处理中断;⑤恢复被中断进程的CPU现场。对中断进行处理的过程为:①测定是否有未响应的中断信号;②保护被中断进程的CPU现场;③转入相应的设备处理程序;④处理中断;⑤恢复CPU现场并退出中断。

6. (考研真题)为什么要有设备驱动程序?用户进程是如何通过设备驱动程序来控制设备工作的?

【参考答案】设备驱动程序与硬件密切相关,主要负责接收上层软件发来的I/O指令,并将其转换成具体要求发送给设备控制器;反之,也将来自设备控制器的信号传送给上层软件。采用设备驱动程序实现I/O系统的高层与设备控制器之间的通信,驱动I/O设备工作。

用户进程通过设备驱动程序控制设备工作的过程为:①接收由用户进程发来的I/O命令和参数,并将命令中的抽象要求转换为具体要求,如将磁盘盘块号转换为磁盘的盘面、磁道和扇区号;②检查用户I/O请求的合法性,如果请求不合法,则拒绝接收I/O请求并反馈用户进程;③了解I/O设备的状态,如果设备准备就绪,则可由设备控制器设置设备的工作方式、传递有关参数;否则,将请求者的请求块挂到设备请求队列上等待;④发出I/O命令,如果设备空闲,则立即启动I/O设备,完成指定的I/O操作。

7. 推动 I/O 控制方式发展的主要因素是什么？

【参考答案】推动I/O控制方式发展的主要因素包括：①尽量减少CPU对I/O控制的干预，把CPU从繁杂的I/O控制中解脱出来，以便更多地去完成数据处理任务；②缓和CPU的高速性和设备的低速性之间速度不匹配的矛盾，以提高CPU利用率和系统吞吐量；③提高CPU和I/O设备操作的并行程度，使CPU和I/O设备都处于忙碌状态，提高整个系统的资源利用率和系统吞吐量。

8. 请说明中断驱动 I/O 方式和 DMA (direct memory access, 直接存储器访问) 方式有什么不同。

【参考答案】它们主要在以下两个方面有不同。

(1) I/O中断频率。在中断驱动I/O方式中，每当输入数据缓冲寄存器中装满输入数据或将输出数据缓冲寄存器中的数据输出之后，设备控制器就会发生一次中断，由于设备控制器中配置的数据缓冲寄存器通常较小，如1个字节或1个字，因此中断比较频繁。而在DMA方式中，在DMA控制器的控制下，一次能完成一批连续数据的传输，并在整批数据传送完成后才发生一次中断，因此可大大减少CPU处理I/O中断的时间。

(2) 数据的传送方式。在中断驱动I/O方式中，由CPU直接将输出数据写入控制器的数据缓冲寄存器中以供设备输出，或在中断发生后直接从数据缓冲寄存器中取出输入数据以供进程处理，即数据的传送必须经过CPU；而在DMA方式中，数据传送是在DMA控制器的控制下直接在内存和I/O设备间进行的，CPU只须将磁盘地址、内存地址和字节数传给DMA控制器即可。

9. 设备无关性的基本含义是什么？为什么要设置设备无关性软件？

【参考答案】为了提高OS的可适应性和可扩展性，在现代OS中都实现了设备无关性（设备独立性），其基本含义是应用程序独立于物理设备。为了实现设备无关性，引入了逻辑设备和物理设备这两个概念。在应用程序中，可通过逻辑设备名称请求使用某类设备；而系统在实际执行时，必须使用其对应的物理设备名称，应用程序应与实际使用的物理设备无关，由OS考虑因物理设备不同而需要使用不同的设备驱动程序等问题。

设置设备无关性软件的目的是：①增加设备分配的灵活性；②实现I/O重定向。当I/O设备被更换时不必改变应用程序，即I/O重定向。

10. 设备分配过程中可能会出现死锁吗？为什么？

【参考答案】在设备分配过程中，如果在进程发出I/O请求后，系统为其分配了I/O设备，则进程仍继续运行；运行中再次发出新的I/O请求，只有当某个请求不满足时，进程才会阻塞。这种情况下，一个进程可以同时使用多个设备。虽然进程推进迅速，但此时的资源分配方式是不安全的，可能会发生死锁。

11. SPOOLing (simultaneous peripheral operations on-line, 假脱机) 系统由哪几部分组成？以打印机为例，说明如何利用 SPOOLing 技术实现多个进程对打印机的共享？

【参考答案】SPOOLing系统由磁盘上的输入井和输出井、内存中的输入缓冲区和输出缓冲区、输入进程和输出进程以及井管理程序组成。

采用SPOOLing技术共享打印机时，对所有提出输出请求的用户进程，系统接受它们的请求时，并不真正分配打印机，而是由SPOOLing管理进程做两件事情：①在输出井中为它申请一空闲缓冲区，并将要打印的数据送入其中；②为用户进程申请一张空白的用户打印请求表，并将用户的打印请求填入该表中，再将该表挂到SPOOLing文件队列上。

至此，用户进程认为打印过程已经完成，而不必等待真正的慢速的打印过程完成。当打印机空闲时，SPOOLing打印进程将从SPOOLing文件队列的队首取出一张打印请求表，根据表中的要求将要打印的数据从输出井传送到内存输出缓冲区，再由打印机进行输出打印。打印完成后，再处理SPOOLing文件队列中的下一个打印请求表，直至队列为空。这样，虽然系统中只有一台打印机，但系统并未将它分配给任何进程，而只是为每个提出打印请求的进程在输出井中分配一个存储区（相当于一个逻辑设备），使每个进程都感觉自己独占一台打印机，从而实现了多个进程对打印机的共享。

12. (考研真题) 在单缓冲区情况下，为什么系统对一块数据的处理时间为 $\max(C, T) + M$ ？

【参考答案】以输入为例，假定从磁盘中把一块数据输入缓冲区的时间为 T ；OS将缓冲区数据传送到用户区的时间为 M ；CPU对这块数据进行计算的时间为 C 。在单缓冲区情况下，由于设备的输入操作和CPU的处理操作可以并行执行，因此系统对每一整块数据的处理时间为 $\max(C, T) + M$ 。

7.2 计算题参考答案

13. (考研真题) 设系统缓冲区和用户工作区均采用单缓冲区，从外设读入 1 个数据块到系统缓冲区的时间为 100，从系统缓冲区读入 1 个数据块到用户工作区的时间为 5，对用户工作区中的 1 个数据块进行分析的时间为 90（见图 1-7-2）。进程从外设读入并分析 2 个数据块的最短时间是多少？

【参考答案】数据块1从外设到用户工作区的总时间为 105，在这段时间内，数据块2没有进行操作。在数据块1进行分析处理时，数据块2从外设到用户工作区的总时间为 105，在这段时间内两个过程是并行的。加上处理数据块2的时间 90，总共是 $105 + 105 + 90 = 300$ 。

14. 假定把磁盘上一个数据块中的信息输入一单缓冲区的时间 T 为 $100\mu\text{s}$ ，将缓冲区中的数据传送到用户区的时间 M 为 $50\mu\text{s}$ ，CPU对这一块数据进行计算的时间 C 为 $50\mu\text{s}$ 。请问，系统对一块数据的处理时间为多少？如果将单缓冲区改为双缓冲区，则系统对一块数据的处理时间为多少？

【参考答案】单缓冲区处理一块数据的时间为 $\max(C, T) + M = \max(50\mu\text{s}, 100\mu\text{s}) + 50\mu\text{s} = 150\mu\text{s}$ ；双缓冲区处理一块数据的时间为 $\max(C, T) = \max(50\mu\text{s}, 100\mu\text{s}) = 100\mu\text{s}$ 。

15. (考研真题) 某磁盘的转速为 10 000r/min，平均寻道时间为 6ms，磁盘传输速率为 20MB/s，磁盘控制器时延为 0.2ms，读取一个 4KB 的扇区所需的平均时间约为多少？

【参考答案】磁盘转速是 10 000r/min，平均转 1 转的时间是 6ms，因此查询扇区的平均时间是 $(1/2) \times 6\text{ms} = 3\text{ms}$ 。因为磁盘传输速率是 20MB/s，所以读取 4KB 扇区信息的时间为 $4\text{KB} / (20\text{MB/s}) = 0.2\text{ms}$ 。时延为 0.2ms，平均寻道时间为 6ms，因此总时间为 $3\text{ms} + 6\text{ms} + 0.2\text{ms} + 0.2\text{ms} = 9.4\text{ms}$ 。

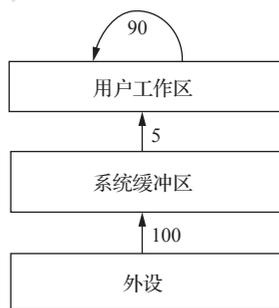


图 1-7-2 单缓冲区处理数据过程图

16. 某磁盘有 40 个柱面, 查找移过每个柱面需要 5ms, 若文件信息块凌乱存放, 则相邻逻辑块平均间隔 9 个柱面。文件信息块经优化分布后, 相邻逻辑块平均间隔 2 个柱面。假设磁盘时延为 100ms, 传输速度为 20ms/块, 请问在信息块非优化存放和优化存放两种情况下, 传输 100 块文件信息各需多长时间?

【参考答案】在两种情况下传输 100 块文件信息所需时间分析如下。

(1) 非优化存放。读 1 块数据需要时间: $9 \times 5 + 100 + 20 = 165\text{ms}$, 则传输 100 块文件信息需要时间 16 500ms。

(2) 优化存放。读 1 块数据需要时间: $2 \times 5 + 100 + 20 = 130\text{ms}$, 则传输 100 块文件信息要时间 13 000ms。

17. (考研真题) 假设有 11 个进程先后提出磁盘 I/O 请求, 当前磁头正在 110 号磁道处, 并预向磁道序号增加的方向移动。请求队列的顺序为 30、145、120、78、82、140、20、42、165、55、65, 分别用 FCFS 调度算法和 SCAN 调度算法完成上述请求, 写出磁道访问顺序和每次磁头移动的距离, 并计算平均移动磁道数。

【参考答案】本题分步解答如下。

(1) FCFS 调度算法: 访问顺序为 30、145、120、78、82、140、20、42、165、55、65; 移动距离为 80、115、25、42、4、58、120、22、123、110、10; 平均移动磁道数为 $(80 + 115 + 25 + 42 + 4 + 58 + 120 + 22 + 123 + 110 + 10) / 11 = 64.45$ 。

(2) SCAN 调度算法: 访问顺序为 120、140、145、165、82、78、65、55、42、30、20; 移动距离为 10、20、5、20、83、4、13、10、13、12、10; 平均移动磁道数为 $(10 + 20 + 5 + 20 + 83 + 4 + 13 + 10 + 13 + 10 + 10) / 11 = 18.18$ 。

18. (考研真题) 磁盘请求服务队列中要访问的磁道分别为 38、6、37、100、14、124、65、67, 磁头上次访问了 20 磁道, 当前处于 30 磁道上, 试采用 FCFS、SSTF (shortest seek time first, 最短寻道时间优先) 和 SCAN 调度算法, 分别计算磁头移动的磁道数。

【参考答案】本题分步解答如下。

(1) FCFS 调度算法: 磁头移动顺序为 30、38、6、37、100、14、124、65、67; 移动磁道数为 $8 + 32 + 31 + 63 + 86 + 110 + 59 + 2 = 391$ 。

(2) SSTF 调度算法: 磁头移动顺序为 30、37、38、14、6、65、67、100、124; 移动磁道数为 $7 + 1 + 24 + 8 + 59 + 2 + 33 + 24 = 158$ 。

(3) SCAN 调度算法: 磁头移动顺序为 30、37、38、65、67、100、124、14、6; 移动磁道数为 $7 + 1 + 27 + 2 + 33 + 24 + 110 + 8 = 212$ 。

7.3 综合应用题参考答案

19. (考研真题) 目前 PC (personal computer, 个人计算机) 上使用的外部存储设备的速度都相当快, 例如刻录一张 DVD (单面单层 DVD 的容量通常大约为 4.7GB) 需要几分钟到十几分钟时间。与 DVD 相比, 硬盘的速度更快。请问: 这样的高速设备采用的大概是什么样的 I/O 控制方式? 请说出你的推断理由。

【参考答案】程序 I/O 方式会使 CPU 陷入忙等状态, 这会极大地浪费 CPU 的计算资源, 因此程序 I/O 方式的效率比较低, 在现代的 OS 中已经十分少见了。中断 I/O 方式对于高速设备而言,

每秒产生的中断次数多达几百万甚至更多，这同样会给CPU带来极大的计算资源开销。并且PC现在通常没有通道机制。因此DMA方式是DVD这种高速设备通常采用的I/O控制方式。

20. 除了FCFS调度算法外,所有磁盘调度算法都不公平,例如会造成有些请求“饥饿”,试分析:

- (1) 为什么不公平?
- (2) 如何构建一种公平性调度算法?
- (3) 为什么公平性在分时系统中是一个很重要的指标?

【参考答案】

(1) 比如SSTF调度算法,其要求访问的磁道与当前磁头所在的磁道距离最近,以使每次的寻道时间最短,如果新进程请求要访问的磁道与磁头当前所在的磁道距离较近,则必会优先被满足,进而就会产生“饥饿”现象。在SCAN调度算法中,若磁头刚在某方向上越过某一磁道,某进程就发出此磁道的访问请求,则此时进程会等待,即等磁头继续从该方向访问剩余磁道后,反方向再扫描完所有要访问的磁道,然后处理此进程的请求,这会致使该进程的等待时间在最坏情况下为 $2T$ (T 为由里向外或由外向里单向扫描完要访问的磁道所需的时间)。CSCAN调度算法会始终保持一个方向依次访问所有磁道;但当有进程反复地请求某个磁道的I/O操作时,该进程就会垄断整个磁盘设备,进而发生磁臂粘着现象。

(2) 公平性调度算法的整体思想是避免某请求长时间等待。为此,可划定一个时间界限,把这段时间内尚未得到服务的请求强制移到队列首部,并标记任务新请求不能插到这些请求前。在SSTF调度算法中,可以重新排列这些老请求,以优先处理它们。

(3) 公平性在分时系统中可避免分时进程等待时间过长而产生超长的响应时间。

21. 假设有4个记录(A、B、C、D)被存放在磁盘的某个磁道上,该磁道被划分成4块,每块存放1个记录,其布局如表1-7-1所示。

现在要顺序处理这些记录。假定磁盘转速为 20ms/r ,处理程序每次从磁盘读出一个记录后要花 5ms 对其进行处理,若磁头现在处于首个逻辑记录的始点位置,则请问:

- (1) 处理程序处理完这4个记录所花费的时间是多少?
- (2) 按最优化分布重新安排这4个逻辑记录,写出记录的安排顺序,并计算处理所需要的时间。

【参考答案】

(1) 磁盘旋转速度为 20ms/r ,磁道划分为4块,每块存放一个记录,因此读出一个记录要花 5ms 。优化前处理记录所花费的总时间为 $(5+5) + (5 \times 4+5) + (5 \times 4+5) + (5 \times 4+5) = 85\text{ms}$ 。

(2) 优化后记录的安排顺序为A、C、B、D,如表1-7-2所示。

表1-7-1 记录存放布局情况

| 块号 | 记录号 |
|----|-----|
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | D |

表1-7-2 优化后记录存放布局情况

| 块号 | 记录号 |
|----|-----|
| 1 | A |
| 2 | C |
| 3 | B |
| 4 | D |

优化后处理记录所花费的总时间为 $(5+5) + (5+5) + (5+5+5) + (5+5) = 45\text{ms}$ 。

22. 假定磁盘的磁臂现在处于 6 号柱面上, 有表 1-7-3 所示的 6 个请求进程等待访问磁盘, 试列出最省时间的响应次序。

表1-7-3 请求进程等待访问磁盘位置

| 请求进程序号 | 柱面号 | 磁头号 | 块号 |
|--------|-----|-----|----|
| 1 | 7 | 6 | 2 |
| 2 | 5 | 5 | 6 |
| 3 | 15 | 20 | 6 |
| 4 | 7 | 4 | 4 |
| 5 | 20 | 9 | 5 |
| 6 | 5 | 15 | 2 |

【参考答案】由题意可知最省时间的调度算法为SSTF调度算法, 该调度算法要求访问的磁道离所在磁道最近, 以使每次的寻道时间最短, 但此调度算法不能保证平均寻道时间最短。由于磁臂刚开始处于6号柱面上, 由表1-7-3可知, 所有请求进程中离磁柱6号柱面最近的为1、2、4、6号请求进程, 均只有1个柱面的距离, 由于请求进程3和请求进程5的柱面号均大于当前的柱面号, 因此应选择2号或6号请求进程作为第1个请求进程来处理。由于请求进程6的块号为2, 比请求进程2的块号6要小, 旋转时间较短, 更靠近6号柱面, 因此第1个响应的是请求进程6, 然后是请求进程2。此时, 磁臂处于5号柱面。同理, 由于请求进程4的块号比请求进程1的块号小, 故先响应请求进程4, 然后响应请求进程1。以此类推, 再响应请求进程3、请求进程5。因此, 最省时间的响应次序是: 6→2→4→1→3→5。

第8章 文件管理

8.1 简答题参考答案

1. 何谓数据项、记录和文件?

【参考答案】①数据项是最低级的数据组织形式, 可分为基本数据项和组合数据项。基本数据项是描述一个对象某种属性的字符集, 是数据组织中可以命名的最小逻辑数据单位, 又称为数据元素或字段。组合数据项是若干个基本数据项所构成的数据项。②记录是一组相关数据项的集合, 用于描述一个对象某方面的属性。③文件是由创建者定义的、具有文件名的一组相关信息的集合。

2. 一个比较完善的文件系统应具备哪些功能?

【参考答案】一个比较完善的文件系统应具备5个功能。①文件存储空间管理。通过文件存

存储空间管理,能使文件“各得其所”,并且能尽量提高文件存储空间的利用率。②目录管理。通过目录管理,能实现文件“按名存取”,提高文件的检索速度,解决文件的命名冲突问题(允许文件重名),并能实现文件共享。③文件读/写管理。通过文件读/写管理,可以实现文件数据的快速读/写。④文件安全性管理。通过采取多级文件保护等措施,可以实现对系统中文件的保护,防止文件被偷窃、修改和破坏。⑤用户接口管理。文件系统向用户提供一个统一的、方便使用的接口,用户通过该接口可以方便地获得如文件存取、创建、删除、修改等文件管理服务。

3. 为什么在大多数 OS 中都引入了“打开”这一文件系统调用? 打开的含意是什么?

【参考答案】①当用户要求对一个文件实施多次读/写或者其他操作时,每次都从检索目录开始。为了避免多次重复检索目录,在大多数OS中都引入了“打开”这一文件系统调用,当用户第一次请求对某文件进行操作时,须先利用open系统调用将该文件打开。②所谓“打开”,是指系统将指定文件的属性(包括该文件在外存上的物理位置)从外存复制到内存中已打开文件表的一个表目中,并将该表目的编号(或称索引号)返回给用户。“打开”就是在用户和指定文件之间建立起一个连接。此后,用户通过该连接可直接得到文件信息,避免了再次通过目录检索文件,即当用户再次向系统发出文件操作请求时,系统可根据用户提供的索引号直接在已打开文件表中查找到文件信息。这样不仅节省了检索开销,而且提高了文件操作速度。如果用户不再对该文件实施操作,则可以利用“关闭”系统调用来关闭此文件,即断开此连接,此时,OS将会把该文件从已打开文件表的表目中删除。

4. 什么是文件的逻辑结构? 逻辑文件有哪几种组织形式?

【参考答案】①文件的逻辑结构是指从用户的角度出发所观察到的文件组织形式,也就是用户可以直接处理的数据及其结构。②逻辑文件根据其结构可分为两种:一种是无结构的流式文件,是指文件信息由一串字符流构成;另一种是有结构的记录式文件,是指将文件信息按照在逻辑上独立的含义划分为信息单位,每个信息单位称为一个逻辑记录(简称记录)。

5. 如何提高变长记录顺序文件的检索速度?

【参考答案】为了提高变长记录顺序文件的检索速度,可建立一张索引表,以主文件中每条记录的长度及指向对应记录的指针(即该记录在逻辑地址空间的起始地址)为表项的内容。由于索引表是一个定长记录的顺序文件,其可实现方便快速的直接存取。但是,如果文件较大,则可建立多级索引以提高检索效率。

6. 什么是“按名存取”? 文件系统如何实现文件的按名存取?

【参考答案】①“按名存取”指用户只要给出文件名就能存取外存空间中的文件信息,而不必给出文件的具体物理地址。②文件系统实现文件按名存取的步骤为:首先利用用户提供的文件名,检索文件目录中该文件的FCB(file control block,文件控制块)或索引节点;然后根据FCB中的文件物理地址,将文件读入内存。

7. UNIX 系统把文件描述信息从文件目录项中分离出来的原因是什么?

【参考答案】将文件描述信息从文件目录项中分离出来后,目录项中只包含文件名和索引节点编号,文件的其他描述信息放在索引节点中。这样的分离主要有两个优点。

(1)提高查找速度。查找文件时,用欲查找的文件名与文件目录中的文件名进行依次比较。由于文件目录存在于外存,比较时须将其读入内存。FCB包括全部文件描述信息时,一个外存盘块仅能保存较少的FCB,查找速度较慢;将FCB描述信息分离后,文件目录中仅保存FCB

文件名和索引节点编号信息，此时一个盘块中可容纳较多的FCB，进而即可减少磁盘块的读取次数，提高文件检索速度。

(2) 实现文件链接。链接就是将路径名作为文件名，使不同的用户均可使用。描述信息分离后，目录项只包括文件名和索引节点编号，索引节点中包含文件描述信息和标志该索引节点与对应索引号相连接的计数。

8. 目前广泛采用的目录结构是哪种？它有什么优点？

【参考答案】目前广泛采用的目录结构是多级树形目录，它具有以下优点。

(1) 能有效提高对目录的检索速度。假定文件系统中有 N 个文件，在单级目录中，最多要检索 N 个目录项；但对于 i 级树形目录，在目录中每检索一个指定的文件，最多可能要检索 $i \times \sqrt[i]{N}$ 项。

(2) 允许文件重名。在树形结构的文件系统中，不仅允许每个用户在自己的分目录中使用与其他用户的文件相同的名字，而且允许同一个用户的不同分目录中的文件重名。

(3) 便于实现文件共享。在树形目录中，用户可通过路径名共享他人的文件；也可将共享文件链接到自己的目录下实现共享，实现方式是系统在用户目录中增设一个目录项，并在其中填上用户赋予共享文件的新文件名，以及该共享文件的唯一标识符（或索引节点编号）。

(4) 能更有效地进行文件的管理和保护。在多级目录中，用户可按文件性质的不同，将它们存放到不同的子目录中，并且可以赋予各目录不同的存取权限，因此，能更有效地管理和保护文件。

9. 试说明在树形目录中线性检索法的检索过程，并画出相应的流程图。

【参考答案】在树形目录中，用户提供的是从根目录（或当前目录）开始的、由多个文件分量名所组成的文件路径名。系统在检索一个文件时：①系统读入给定文件路径名中的第一个文件分量名，用它与根目录（或当前目录）文件中各目录项的文件名进行顺序比较，若能找到匹配的目录项，则在它的FCB或索引节点编号中找到对应的文件；②系统读入第二个文件分量名，用它与刚检索到的目录文件中的各目录项文件名进行顺序比较，若能找到匹配者，则重复上述过程；③如此逐级检索指定文件的分量名，最后将会得到指定文件的FCB或索引节点编号。检索过程的流程（以UNIX系统为例）如图1-8-1所示。

10. 在树形目录中，利用链接方式共享文件有何好处？

【参考答案】利用链接方式共享文件主要有以下几方面的好处。

(1) 方便用户。此共享方式允许用户将共享文件链接到自己的子目录下，并赋予它新的文件名，方便用户管理和使用共享文件。

(2) 可防止共享文件被任意删除。每次链接时，系统对索引节点中的链接计数段 i_nlink 执行加1操作；删除时，先对该字段执行减1操作，只有当 i_nlink 值为0时，共享文件才会被真正删除，因此可防止共享文件被任意删除。

(3) 可加快检索速度。为了加快检索文件的速度，一般系统都引入了当前目录。若共享文件已被链接到当前目录下，则系统无须再逐级检索目录，进而提高了检索速度。

11. 什么是保护域？进程与保护域之间存在着怎样的动态联系？

【参考答案】保护域是进程对一组对象的访问权的集合。保护域规定了进程所能访问的一组对象（硬件或软件）以及所能执行的操作（即访问权）。

进程与保护域之间的动态联系是指进程的可用资源集在其整个生命周期中是变化的，进程运行在不同的阶段时可以根据需要从一个保护域切换到另一个保护域。

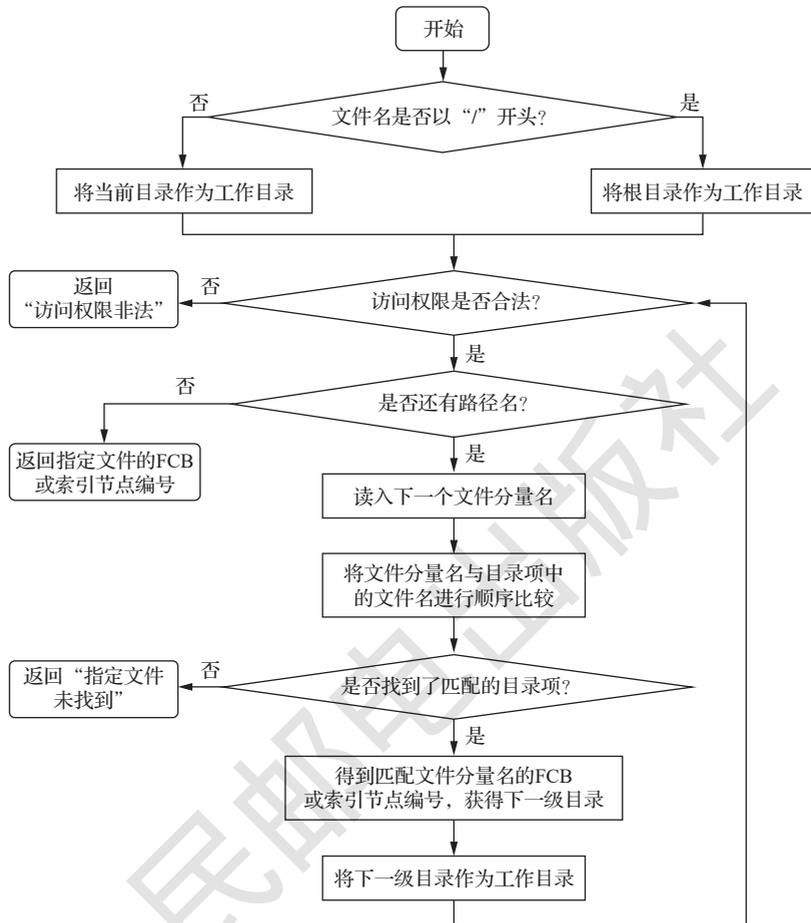


图 1-8-1 线性检索法流程图

12. 什么是访问控制表和访问权限表？系统如何利用它们来实现对文件的保护？

【参考答案】本题分步解答如下。

(1) 访问控制表是对访问矩阵按列（对象）进行划分，由有序对集{〈域，权集〉}组成，可用于描述不同用户（进程）对于同一对象的不同访问权限集。访问控制表也可用于定义各域对于某对象默认访问权集，其可作为资源能否使用的首要依据。

(2) 访问权限表是对访问矩阵按行（域）进行划分，由有序对集{〈对象，权集〉}组成，表示一个域对每个对象可以执行的一组操作。当域为用户（进程）、对象为文件时，访问权限表可用于描述一个用户（进程）对每个文件所能执行的一组操作。

(3) 系统为每个对象配置一张访问控制表。当一个进程第一次试图访问一个对象时，必须先检查访问控制表，确定进程是否具有对该对象的访问权。如果其无权访问该对象，则系统会拒绝本次访问，并构成一个例外（异常）事件；否则（有权访问），允许本次访问，为该进程建立一个访问权限，并将该权限连接到该进程。以后，进程便可直接利用这一返回的权限去访问该对象，亦可快速验证访问的合法性。当进程不再需要对该对象进行访问时，可撤销该访问权。

8.2 计算题参考答案

13. 一个文件系统中, FCB 占 64B, 一个盘块大小为 1KB, 采用单级文件目录, 假如文件目录中有 3 200 个目录项, 则检索一个文件平均需要访问磁盘大约多少次?

【参考答案】文件目录中共有 3 200 个目录项, 每个目录项占 64B, 因此, 该目录文件共占用 $3\ 200 \times 64\text{B} = 200\text{KB}$, 相当于 200 个盘块。查找一个目录项, 平均需要调入盘块 $(N+1)/2 = (200+1)/2 \approx 100$ 次。

14. 在某个文件系统中, 每个盘块为 512B, FCB 占 64B, 其中文件名占 8B。如果索引节点编号占 2B, 则针对一个存放在磁盘上的、具有 256 个目录项的目录, 请分别计算引入索引节点前后, 为找到某个文件的 FCB 而平均启动磁盘的次数。

【参考答案】①在引入索引节点前, 每个目录项中均会存放对应文件的 FCB, 故 256 个目录项的目录须占用 $256 \times 64/512 = 32$ 个盘块。因此, 该目录中检索到一个文件平均启动磁盘的次数为 $(32+1)/2 = 16.5 \approx 17$ 次。②在引入索引节点后, 每个目录项中须存放文件名和索引节点编号, 因此 256 个目录项的目录总共需要占用 $256 \times (8+2)/512 = 5$ 个盘块。因此, 找到匹配的目录项平均需要启动 $(5+1)/2 = 3$ 次磁盘; 得到索引节点编号后, 还须启动磁盘以将对应文件的索引节点读入内存, 故平均需要启动磁盘 4 次。因此, 引入索引节点后可减少启动磁盘的次数, 提高了检索文件的速度。

15. (考研真题) 设文件 F_1 的当前引用计数值为 1, 先建立 F_1 的符号链接 (软链接) 文件 F_2 , 再建立 F_1 的硬链接文件 F_3 , 然后删除 F_1 。此时, F_2 和 F_3 的引用计数值分别是多少?

【参考答案】①建立符号链接 (软链接) 时, 引用计数值直接被复制, 因此 F_2 的引用计数值为 1。②建立硬链接时, 引用计数值加 1, F_3 的引用计数值为 2; 但在删除文件时, 对于硬链接, 引用计数值减 1, 因此删除 F_1 后 F_3 的引用值变为 1。

16. (考研真题) 索引顺序文件可能是最常见的一种逻辑文件组织形式, 其不仅有效克服了变长记录文件不便于直接存取的缺点, 且付出的额外存储开销也不算大, 对于包含 40 000 条记录的主数据文件, 为了能检索到指定关键字的记录, 采用索引顺序文件组织方式, 平均检索效率可提高到顺序文件组织方式的多少倍 (假定主数据文件和索引表均采用顺序查找法) ?

【参考答案】若顺序文件中所含有的记录数为 N , 则检索到指定关键字的记录平均须查找 $N/2$ 个记录。但对于索引顺序文件, 检索到指定关键字的记录平均须查找 \sqrt{N} 个记录, 因而其检索效率比顺序文件提高了约 $\sqrt{N}/2$ 倍。因此, 包含 40 000 条记录的文件, 采用索引顺序文件组织方式的平均检索效率比顺序文件提高了 $\sqrt{40\ 000}/2 = 100$ 倍。

17. (考研真题) 某文件系统的目录由文件名和索引节点编号构成。若每个目录项的长度均为 64B, 其中 4B 存放索引节点编号, 60B 存放文件名。文件名由小写英文字母构成, 则该文件系统能创建的文件数量上限为多少?

【参考答案】创建的文件数量上限等于索引节点数量上限, 索引节点为 4B, 即为 32 位, 故最多有 2^{32} 个索引节点, 即最多能创建 2^{32} 个文件。

8.3 综合应用题参考答案

18. 针对图 1-8-2 所示的文件系统目录结构,若 C 和 D 分别是两个用户的目录,则请问:

- (1) C 用户在当前目录“/C”下欲共享文件 f2, 应具备什么条件?
- (2) 若 C 用户需要经常访问文件, 则其应如何操作才会更简单、更快捷?
- (3) 若 D 用户不愿意别人访问其文件 f3, 则其应如何操作?

【参考答案】

(1) 只有当 C 用户拥有对文件的访问权限时, 才能共享文件 f2, 此时, 通过绝对路径名“/D/H/f2”或相对路径名“.../D/H/f2”即可访问文件 f2。

(2) 如果 C 用户需要经常访问文件 f4, 则其可以在自己的工作目录下建立一个文件 f4 的链接文件 Df4, 以后他在工作目录“/C”下便可直接使用路径“Df4”来访问文件 f4, 操作更方便, 速度更快。

(3) D 用户可以将文件 f3 的访问权限设置成只允许文件主访问, 而不允许其他用户访问。

19. 有一共享文件, 它具有下列文件名: /usr/Wang/test/report、/usr/Zhang/report、/usr/Lee/report, 试填写图 1-8-3 中的 A、B、C、D、E。

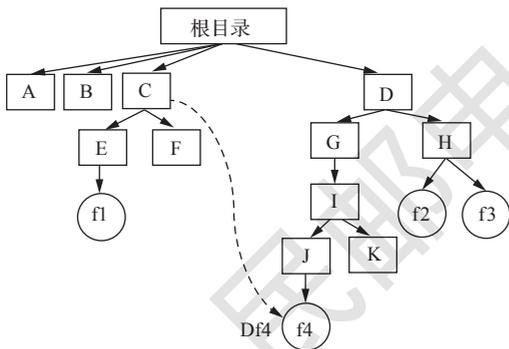


图 1-8-2 文件系统目录结构

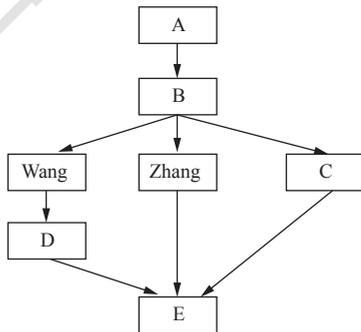


图 1-8-3 文件共享示意

【参考答案】根据题意可知 A 为根目录 root; 由图 1-8-3 可知, B 为 Zhang、Wang 的上一级目录, 根据目录路径可知访问 Zhang 和 Wang 都要经过 usr 目录, 因此 B 为 usr 目录; 根据路径 /usr/Lee/report 可知, C 为 Lee; D 为 Wang 的子目录, 由路径 /usr/Wang/test/report 可知, D 为 test; E 为该树形目录的最后一级目录, 根据题意可知 E 为 report。

20. (考研真题) 假设某系统的目录管理采用了索引节点方式。如果用户需要打开文件 /usr/student/myproc.c, 则请简要阐述目录检索的大致过程 (假设根目录内容已经读入内存且该文件存在)。

【参考答案】目录检索的大致过程为: ①在根目录中查找 usr 目录项; ②找到 usr 目录项后, 根据其索引节点编号, 获得索引节点位置并将其内容读入内存 (若不在内存), 然后根据索引节点中的文件外存地址读入 usr 的内容; ③在 usr 中查找 student 目录项, 获取索引节点并读入 student 的内容 (过程类似于第 ② 步, 若还有更深层次的目录, 则中间目录也采用类似的方法); ④在 student 中查找 myproc.c 目录项, 获取其索引节点 (过程类似于第 ② 步)。

第9章 磁盘存储器管理

9.1 简答题参考答案

1. (考研真题) 文件物理结构是指一个文件在外存上的存储组织形式, 主要有连续结构、链接结构和索引结构这 3 种, 请分别简述它们的优缺点。

【参考答案】 本题分步解答如下。

(1) 连续结构。优点: ①存储管理简单, 容易实现; ②支持顺序存取和随机存取; ③顺序访问速度快。缺点: ①要求为每个文件分配连续的存储空间; ②必须事先知道文件的长度, 要求能灵活地插入和删除记录; ③不利于文件的动态增长。

(2) 链接结构。优点: ①消除了磁盘的外部碎片, 提高了磁盘空间的利用率; ②能适应文件的动态增长; ③方便插入、修改和删除记录。缺点: ①存取速度较慢, 不适合随机存取; ②物理块间的链接指针错误会造成数据丢失, 可靠性差; ③需要较多的寻道次数和较长的寻道时间; ④链接指针会占用空间, 降低了空间利用率。

(3) 索引结构。优点: ①既能顺序存取, 又能随机存取; ②能适应文件的动态增长; ③方便插入、修改和删除记录。缺点: ①需要较多的寻道次数和较长的寻道时间; ②索引表增加了系统开销, 包括内存空间和存取时间。

2. 在 FAT 中为什么要引入“簇”的概念? 以“簇”为基本分配单位有什么好处?

【参考答案】 ①引入“簇”的原因是为了适应磁盘容量不断增大的需要, 在进行盘块分配时, 不再以盘块而是以簇为基本单位。一个簇所应包含的扇区的数量与磁盘容量直接相关。②以簇为基本分配单位能适应磁盘容量不断增大的情况, 还可以减少 FAT (file allocation table, 文件配置表) 中的项数 (在相同的磁盘容量下, FAT 中的项数与簇的大小成反比), 使 FAT 占用更少的存储空间, 进而减少访问 FAT 的存取开销。

3. 在 MS-DOS 系统中有两个文件 A 和 B, A 占用 11、12、16、14 这 4 个盘块; B 占用 13、18、20 这 3 个盘块。试画出文件 A 和 B 中各盘块间的链接情况及 FAT 的情况。

【参考答案】 文件 A、B 中各盘块间的链接情况及 FAT 的情况如图 1-9-1 所示。

4. 假定一个文件系统的外存组织方式为显示链接组织方式, 在 FAT 中可有 64K 个指针, 磁盘盘块大小为 512B。试问该文件系统能否表示一个 512MB 的磁盘?

【参考答案】 $512\text{MB}/512\text{B}=1\text{M}$ 个盘块, 每个盘块应设置一个指针来指示下一个盘块位置, 故应有 1M 个指针指示 1M 个盘块。

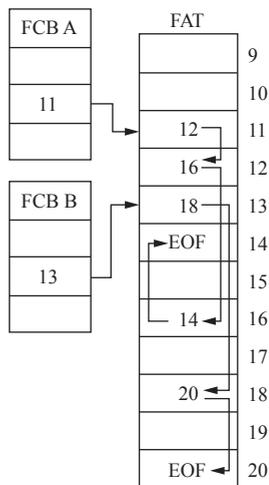


图 1-9-1 盘块间的链接情况及 FAT 的情况

因此, 64K个指针的文件系统不能指引512MB的磁盘。

5. (考研真题) 某文件系统为单级目录结构, 文件的数据一次性写入磁盘, 已写入的文件不可修改, 但可多次创建新文件。请回答如下问题。

(1) 在连续、链式、索引这 3 种文件数据块组织方式中, 哪种更合适? 说明理由。为定位文件数据块, 需要在 FCB 中设计哪些相关描述字段?

(2) 为快速找到文件, 对于 FCB, 是集中存储好, 还是与对应的文件数据块连续存储好? 说明理由。

【参考答案】

(1) 采用连续结构组织方式更合适。该方式组织文件所占磁盘块连续, 因此寻道时间更短, 文件随机访问效率更高。需要在FCB中加入〈起始块号, 块数〉或〈起始块号, 结束块号〉字段。

(2) 将所有的FCB集中存放更好。这样在随机查找文件名时, 只需要访问FCB对应的块, 因此可以减少磁头移动次数和磁盘I/O访问次数。

6. 有一计算机系统利用图 1-9-2 所示的位示图 (行号、列号都从 0 开始编号) 来管理空闲盘块。如果盘块从 1 开始编号, 每个盘块的大小为 1KB, 则请回答下列问题 :

| <i>ij</i> | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|-----------|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ... | | | | | | | | | | | | | | | | |

图 1-9-2 位示图

(1) 现要为文件分配两个盘块, 试具体说明分配过程 ;

(2) 若要释放磁盘的第 300 块, 则应如何处理 ?

【参考答案】

(1) 为某文件分配两个盘块的过程如下:

① 顺序检索位示图, 从中找到第一个值为0的二进制位, 得到其行号*i*=2, 列号*j*=3;

② 计算出该位所对应的盘块号

$$b=i \times 16+j+1=2 \times 16+3+1=36;$$

③ 修改位示图, 令 $\text{map}[2,3]=1$, 并将对应的盘块36分配给文件。按照同样的方式, 可找到第3行、第6列的值为0的位, 将其转换为盘块号55; 将位的值修改为1, 并将55号盘块分配给文件。

(2) 释放磁盘的第300块时, 应进行如下处理:

① 计算出磁盘第300块所对应的二进制位的行号*i*和列号*j*, 即

$$i=(300-1) \text{DIV} 16=18, j=(300-1) \text{MOD} 16=11;$$

② 修改位示图, 令 $\text{map}[18,11]=0$, 表示对应的盘块为空闲块。

7. (考研真题) 某系统采用成组链接法管理磁盘的空闲空间, 目前盘块的链接情况处于图 1-9-3 所示的状态, 先由进程 A 释放物理块 181、135、192, 再由进程 B 申请 4 个物理块。试分别画图说明进程 A 释放物理块后和进程 B 申请物理块后的盘块链接情况。

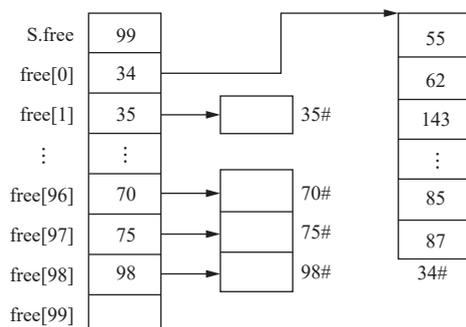


图 1-9-3 盘块链接情况

【参考答案】本题分步解答如下。

(1) 进程A释放物理块181、135、192的过程为：①目前栈中有99个物理块，当进程A释放181#物理块时，栈中的块数为100；②进程A继续释放135#物理块，并将栈中的100个盘块号记录在135#物理块中，然后将135#记录在栈中；③继续释放192#物理块并将其放入栈中，此时S.free的值为2，即栈中的空闲块数为2。进程A释放3个物理块后盘块的链接情况如图1-9-4所示。

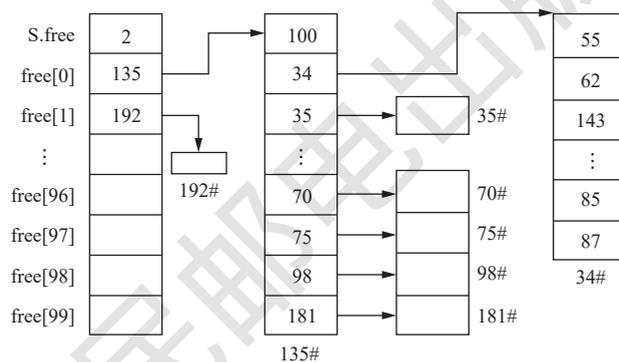


图 1-9-4 进程A释放3个物理块后盘块的链接情况

(2) 进程B申请4个物理块的过程为：①首先申请栈中的192#物理块；②然后将135#物理块中的内容（100个盘块号）复制到栈中，再将135#物理块分配出去，此时S.free的值为100；③依次分配栈中的181#物理块和98#物理块，此时S.free的值为98。进程B申请4个物理块后盘块的链接情况如图1-9-5所示。

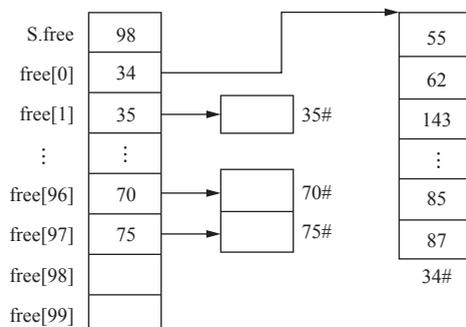


图 1-9-5 进程B申请4个物理块后盘块的链接情况

8. (考研真题) 磁盘长期使用后, 读 / 写磁盘中数据的速度就会变慢 ; 而执行磁盘碎片整理程序后, 读 / 写速度就提高了, 为什么 ?

【参考答案】①文件磁盘存储不连续。计算机系统经过一段时间的使用后, 由于文件修改、删除、新建等操作混合在一起产生的积累效应, 许多文件在磁盘上的存储位置变得不连续, 进而导致读 / 写数据速度变慢。②磁盘是一个靠盘片的高速旋转进行读/写的设备, 当系统需要读/写的数据连续存储在同一磁道时, 存储效率较高。当数据的存储不连续时, 由于寻道需要开销, 效率会有较大程度的下降。用户在读 / 写文件时, 文件的连续存储也有利于提高磁盘工作效率, 提高数据读/写速度。③通过磁盘碎片整理程序整理碎片, 可以将存放文件的数据集中排列, 有效提高了文件系统的性能。

9. (考研真题) 何谓磁盘高速缓存? 在设计磁盘高速缓存时需要考虑哪些问题?

【参考答案】磁盘高速缓存是指在内存中为磁盘盘块设置的一个缓冲区, 在该缓冲区中保存了某些盘块的副本。系统在访问磁盘时, 首先会查看磁盘高速缓存中是否存在要访问的盘块内容, 如果存在, 则可从磁盘高速缓存中去获取, 而无须启动磁盘, 进而提高了访问速度; 如果不存在, 则启动磁盘以将所需盘块内容读入, 并把该内容传送给磁盘高速缓存, 以方便再次访问时获取。

在设计磁盘高速缓存时需要考虑的问题有: ①如何将磁盘高速缓存中的数据传送给请求进程; ②应采用什么样的置换策略; ③已修改的盘块数据应在何时写回磁盘。

10. 什么是磁盘容错技术? 请简述基于集群技术实现容错的几种主要的工作模式。

【参考答案】磁盘容错技术是通过增加冗余的磁盘驱动器、磁盘控制器等方法来提高磁盘系统可靠性的一种技术。磁盘容错技术也叫作系统容错技术 (system fault tolerant, SFT)。

基于集群技术实现容错主要有3种工作模式。①双机热备份模式。在这种模式的系统中, 通常备有两台处理能力完全相同的服务器, 一台为主服务器, 另一台为备份服务器。该模式提高了系统的可用性, 易于实现。主、备份服务器完全独立, 可支持远程热备份, 可以消除由于火灾、爆炸等非计算机因素所造成的隐患。但是该模式中备份服务器处于被动等待状态, 使用效率只有50%。②双机互为备份模式。两台服务器平时均为在线服务器, 各自完成自己的任务。每台服务器内都配置了两个硬盘, 用其中的一个硬盘作为另一台服务器的镜像盘, 实现互相备份。这种模式中两台服务器都可处理任务, 因而系统效率较高。当其中一台服务器发生故障时, 系统可指定另一台服务器来接替它的工作。③公用磁盘模式。将多台服务器连接到一个公共的磁盘系统中, 以消除信息的复制时间, 减少了网络和服务器的开销。

11. 引入检查点的目的是什么? 引入检查点后应如何进行恢复处理?

【参考答案】①引入检查点的目的是使对事务记录表中事务记录的清理工作经常化。②引入检查点后的恢复处理工作由恢复例程来实现: 首先, 查找事务记录表, 确定在最近检查点以前开始执行的最后的事务 T_i ; 找到 T_i 后, 返回搜索事务记录表, 找到第一个检查点记录后, 从该检查点开始, 返回搜索相应的事务记录, 并利用redo和undo过程对它们进行相应的处理。

9.2 计算题参考答案

12. 请分别解释在连续分配方式、隐式链接分配方式、显式链接分配方式和索引分配方式中, 如何将文件的字节偏移量 3 500 转换为物理盘块号和块内位移量 (设物理盘块的大小为 1KB, 盘块号占 4B) ?

【参考答案】 $[3\ 500/1\ 024] = 3$, $3\ 500 \% 1\ 024 = 428$, 即逻辑盘块号为3, 块内位移量为428。

①在连续分配方式中,从文件的FCB中得到分配给该文件的起始物理盘块号,假设为 a_0 ,则字节偏移量3 500对应的物理盘块号为 a_0+3 ,块内位移量为428。②在隐式链接分配方式中,由于在每个盘块中都要留出4B(如最后的4B)来存放文件的下一个盘块号,因此字节偏移量3 500的逻辑盘块号为 $\lceil 3\ 500/1\ 020 \rceil = 3$,而块内位移量为 $3\ 500 \% 1\ 020 = 440$ 。从文件的FCB中获得该文件的首个盘块的盘块号,如 b_0 ,通过读第 b_0 块来获得分配给文件的第1个盘块的盘块号,如 b_1 ;再从 b_1 盘块中得到第2块的盘块号,如 b_2 ;从 b_2 盘块中得到第3块的盘块号,如 b_3 。如此,便可得到字节偏移量3 500对应的物理盘块号为 b_3 ,而块内位移量则为440。③在显式链接分配方式中,从文件的FCB中得到分配给文件的首个盘块的盘块号,如 c_0 ,然后在FAT的第 c_0 项中得到分配给文件的第1个盘块的盘块号,如 c_1 ;再在FAT的第 c_1 项中得到文件的第2个盘块的盘块号,如 c_2 ;在FAT的第 c_2 项中得到文件的第3个盘块的盘块号,如 c_3 。如此,便可获得字节偏移量3 500对应的物理盘块号为 c_3 ,而块内位移量则为428。④在索引分配方式中,从文件的FCB中得到索引表的地址。从索引表的第3项(距离索引表首字节12B的位置)可获得字节偏移量3 500对应的物理盘块号,而块内位移量为428。

13. (考研真题)对于容量为200GB的硬盘,若采用FAT文件系统且盘块大小设定为4KB,则请问其FAT表项长度应当选用16位还是32位(采用二进制表示)?其FAT共须占用多少字节的空间?

【参考答案】①该硬盘共有盘块号 $N=200\text{GB}/4\text{KB}=50\text{M}$ 个,因此FAT中至少要有50M个表项来记录各盘块的使用情况,由于 $32\text{M} < 50\text{M} < 64\text{M}$,表项长度至少用26位二进制表示。为了便于存取,FAT选用32位表示。②因为FAT有50M个表项,FAT表项长度为32位,即 $32/8=4\text{B}$,因此,FAT共占用 $50\text{M} \times 4\text{B}=200\text{MB}$ 的空间。

14. 某个容量为1.44MB的软盘,共有80个柱面,每个柱面上有18个盘块,盘块大小为1KB,盘块和柱面均从0开始编号。文件A依次占据了20、500、750、900这4个盘块,其FCB位于51号盘块上,磁盘最后一次访问的是50号盘块。若采用隐式链接分配方式,则请计算顺序存取该文件的全部内容需要的磁盘寻道距离。

【参考答案】要想顺序存取该文件,须先访问其FCB,得到首个物理盘块号,然后依次访问文件的所有盘块。由此可知,该文件访问顺序为51、20、500、750、900号盘块,对应的磁道号依次为2、1、27、41、50,磁头当前位于50号盘块(即2号磁道)上,因此,磁盘寻道距离为 $(2-2) + (2-1) + (27-1) + (41-27) + (50-41) = 50$ 。

15. (考研真题)某文件系统采用索引物理结构存储文件,磁盘空间为1 000GB。一个目录项可以存储10个盘块的地址,前9个为直接地址,最后一个为一级间址。若盘块的大小为512B,则该文件系统最大能支持的文件大小是多少?

【参考答案】9个直接地址支持的文件长度是 $9 \times 512\text{B} = 4.5\text{KB}$ 。由于一个目录项可以存放10个盘块的地址,因此一级间址可以存放10个目录项的地址,1个一级间址可以存放的文件大小为 $10 \times 512\text{B} = 5\text{KB}$ 。该系统能支持的最大文件长度为 $9 \times 512\text{B} + 10 \times 512\text{B} = 9.5\text{KB}$ 。

16. (考研真题)某文件系统采用混合索引分配方式,如图1-9-6所示,有10个直接块(每个直接块指向1个数据块)、1个一级间接块、1个二级间接块和1个三级间接块,间接块指向的是1个索引块,每个索引块和数据块的大小均为512B,索引块编号的大小为4B。

(1)若只使用直接块,则文件最大为多少字节?

(2)在该系统中能存储的文件最大是多少?

(3)若读取某文件第10MB的内容,则需要访问磁盘几次?

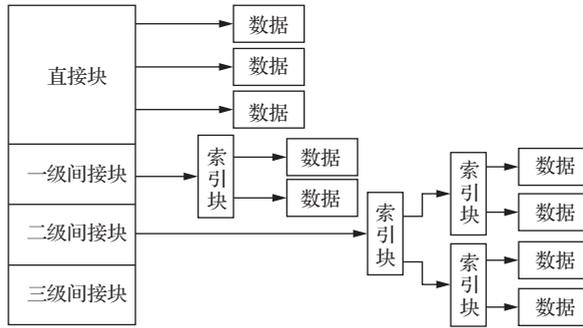


图 1-9-6 文件系统混合索引分配方式

【参考答案】

(1) 直接块指向数据块的块数为10, 因此若只使用直接块, 则文件最大为 $10 \times 512B = 5120B$ 。

(2) 一级间接块指向的索引块指针数为 $512B/4B = 128$ 个, 因此一级间接块指向数据块的块数是128块。二级间接块指向的索引块指针数也是128个, 每个索引块里还有128个指针数据块, 因此二级间接块指向的最大数据块数为 128×128 个。同理, 三级间接块指向数据块的块数是 $128 \times 128 \times 128$ 块。因此该系统能存储的最大文件为 $(10 + 128 + 128^2 + 128^3) \times 512B = 1082201088B = 1056837KB$ 。

(3) 10MB的文件需要的数据块个数为 $10MB/512B = 2048$ 块。直接块和一级间接块指向的数据块个数为 $10 + 128 = 138$ 块 < 2048 块。直接块、一级间接块和二级间接块指向的数据块个数为 $10 + 128 + 128^2 = 16552$ 块 > 2048 块, 因此第10MB的数据应该存放在二级间接块下的某个数据块中, 共需要访问磁盘3次。

17. 在 UNIX 系统中, 如果一个盘块的大小为 1KB, 每个盘块号占 4B, 即每块可放 256 个地址, 一个文件索引节点中磁盘的物理盘块号结构如图 1-9-7 所示。请转换下列文件的字偏移量为物理地址: (1) 9 999; (2) 18 000; (3) 420 000。

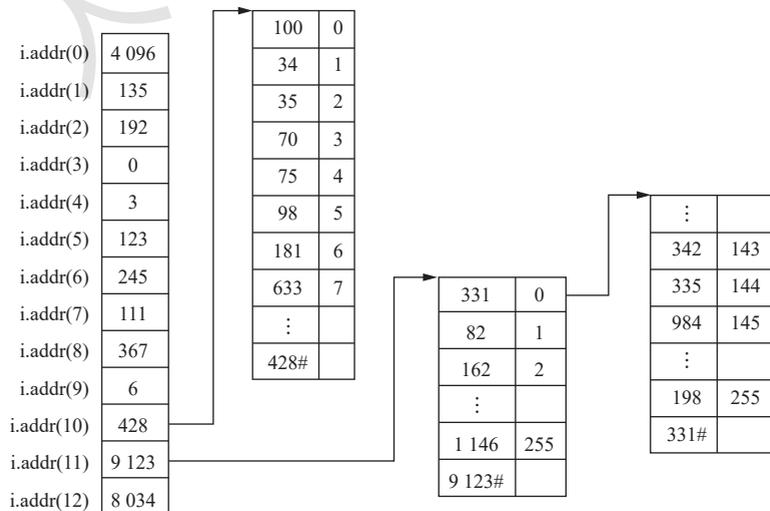


图 1-9-7 文件索引节点中磁盘的物理盘块号结构

【参考答案】在UNIX系统中，文件存储采用混合索引分配方式；即在FCB中，第0~9个地址为直接地址，第10个地址为一级间址，第11个地址为二级间址，第12个地址为三级间址。使用多重索引结构，在索引节点中根据逻辑盘块号通过直接索引或间接索引找到对应的物理块号。

(1) $[9\ 999/1\ 024] = 9$, $9\ 999 \% 1\ 024 = 783$, 因此逻辑盘块号为9；直接索引第9个地址，得到物理盘块号为6，块内偏移地址为783。

(2) $[18\ 000/1\ 024] = 17$, $18\ 000 \% 1\ 024 = 592$, 因此逻辑盘块号为17； $10 < 17 < (10+256)$, 故通过一次间接索引可在第10个地址处得到物理盘块号。 $i.addr(10)$ 中查到428号盘块为一级间址索引块，块内偏移量为 $17-10=7$, 对应的盘块号为633, 该盘块中的592字节即为文件的18 000字节的物理地址，因此块内偏移地址为592。

(3) $[420\ 000/1\ 024] = 410$, $420\ 000 \% 1\ 024 = 160$ 。逻辑盘块号为 $(10+256) < 410$, 通过二次间接索引 $i.addr(11)$ 查得9 123号盘块为二级索引块，因为 $(410-266)/256$ 商为0, 余数为144；从9 123号盘块中查0号盘块，得到一级索引块为331；从331中查144号偏移量，得到数据盘块号335, 该块中的块内偏移地址160即为420 000字节的物理地址。

18. 假设某系统磁盘共含 500 个盘块，盘块号为 0—499。若用位示图法管理这 500 个盘块的空间，则当字长为 32 位时，请问：①位示图需要多少个字？②第 i 字第 j 位对应的盘块号是多少？

【参考答案】①位示图需要 $500 \text{ DIV } 32 = 16$ 个字。②第 i 字第 j 位对应的盘块号是 $32 \times i + j$ 。

9.3 综合应用题参考答案

19. 如果一个文件存放在 100 个数据块中，FCB、FAT、索引块或索引信息等都驻留在内存中。在下面几种情况下，分别需要做几次磁盘 I/O 操作？

- (1) 采用连续分配方式，将最后一个数据块搬到文件头部；
- (2) 采用单级索引分配方式，将最后一个数据块搬到文件头部；
- (3) 采用显式链接分配方式，将最后一个数据块搬到文件头部；
- (4) 采用隐式链接分配方式，将首个数据块搬到文件尾部。

【参考答案】

(1) 采用连续分配方式时，若文件分配到的首个盘块的前面一块不是空闲的，那么将最后一个数据块搬到文件头部就意味着整个文件必须“搬家”，因此要读每个块，然后重写每个块，总共要做200次磁盘I/O操作。

(2) 采用单级索引分配方式时，仅须修改索引表的内容，而无须对文件数据块进行读/写，因此总共要做0次磁盘I/O操作。

(3) 采用显式链接分配方式时，仅须修改FAT中的指针，因此总共要做0次磁盘I/O操作。

(4) 采用隐式链接分配方式时，首先要读第0块以得到原来第1块的块号，将原来第1块的块号记录到FCB中的首块号字段中；然后依次读入第1块到第98块的内容以得到原来最后1块（即第99个数据块）的块号，然后再读入第99块的内容，修改其中的链接指针以使其指向原来的首个块，并重新将该块写入磁盘；把原来的首块（现在的最后一块）中的链接指针的内容修改为EOF，然后将该块重新写入磁盘。因此总共要做102次磁盘I/O操作。

20. 存放在某个磁盘上的文件系统，采用混合索引分配方式，其 FCB 中共有 13 个地址项，第 0—9 个地址项为直接地址，第 10 个地址项为一次间址，第 11 个地址项为二次

间址,第12个地址项为三次间址。如果每个盘块的大小为512B,盘块号占3B,则每个盘块最多存放170个盘块地址。

(1) 将文件的字节偏移量5 000、15 000、150 000转换为物理盘块号和块内偏移量。

(2) 假设某个文件的FCB已在内存中,但其他信息均在外存中,为了访问该文件中某个位置的内容,最少需要访问几次磁盘,最多需要访问几次磁盘?

【参考答案】

(1) $[5\ 000/512]=9$, $5\ 000\%512=392$,即字节偏移量5 000对应的逻辑盘块号为9,块内偏移量为392。由于 $9<10$,故可直接从该文件的FCB的第9个地址项处得到物理盘块号,块内偏移量为392。 $[15\ 000/512]=29$, $15\ 000\%512=152$,即字节偏移量15 000对应的逻辑盘块号为29,块内偏移量为152。由于 $10<29<(10+170)$,而 $29-10=19$,故可从FCB的第10个地址项(即一次间址项)中得到一次间址块的地址;读入该一次间址块并从它的第19项(即该一次间址块的第57~59这3个字节)中获得对应的物理盘块号,块内偏移量为152。 $[150\ 000/512]=292$, $150\ 000\%512=496$,即字节偏移量150 000对应的逻辑盘块号为292,块内偏移量为496。由于 $(10+170)<292<(10+170+170\times 170)$,而 $292-(10+170)=112$, $112/170$ 商为0,余数为112,故可从FCB的第11个地址项(即二次间址项)中得到二次间址块的地址,读入二次间址块并从它的第0项中获得1个一次间址块的地址,再读入该一次间址块并从它的第112项中获得对应的物理盘块号,块内偏移量为496。

(2) 由于文件的FCB已在内存中,为了访问文件中某个位置的内容,最少需要访问1次磁盘(即可通过直接地址直接读文件盘块),最多需要访问4次磁盘(第1次是读三次间址块,第2次是读二次间址块,第3次是读一次间址块,第4次是读文件盘块)。

21. (考研真题) 文件F由200条记录组成,记录从1开始编号,用户打开文件F后,欲将内存中的一条记录插入文件F中,作为其第30条记录。请回答下列问题,并说明理由。

(1) 若文件系统采用顺序分配方式,每个存储块存放1条记录,文件F的存储区域前后均有足够的空闲存储空间,则要完成上述操作最少要访问多少次存储块?文件FCB内容会有哪些改变?

(2) 若文件系统采用链接分配方式,每个存储块存放1条记录和1个链接指针,则要完成上述操作最少要访问多少次存储块?若每个存储块的大小为1KB,其中4B存放指针,则该系统支撑文件的最大长度是多少?

【参考答案】

(1) 文件系统采用连续分配方式,即文件F的200条记录连续。文件F的存储区域前后均有足够的空闲存储空间,在第30条记录处插入1条记录,若要使插入操作访问存储块的次数最少,则需要将前29条记录向前移动。每移动1条记录需要访问2次存储块(1次读出,1次存入),即总共需要访问58次存储块。将第30条记录存入文件F中需要访问1次存储块,故总共需要访问59次存储块。文件F的起始存储块号和文件长度的内容会因插入操作的改变而改变。

(2) 文件系统采用链接分配方式,插入1条记录不需要移动其他记录,只需要找到插入点的位置,然后插入并修改相邻记录的存储块号即可。在第30条记录处插入1条记录,需要从第1条记录开始查找至第29条记录,这个过程须访问29次存储块。插入新记录需要访问1次存储块,修改第29条记录的链接块号,使其指向新插入的第30条记录又需要访问1次存储块,故整个过程共需要访问31次存储块。存放指针需要4B,每字节8位,故最大可寻址的空间是 $2^{32}B=4GB$,每个磁盘大小是1KB(等于1 024B),排除存放指针的4B,则存放数据的部分为 $1\ 024-4=1\ 020B$,故文件系统支持文件的最大长度为 $4G\times 1\ 020B=4\ 080GB$ 。

第10章 多处理机操作系统

1. 为什么说通过提高 CPU 时钟频率来提高计算机运算速度的方法已接近极限?

【参考答案】因为CPU所运算的指令或数据及其结果都是以电子信号的方式通过传输介质送入或送出的。因此,在一个时钟周期内,应至少保证信号在传输介质中能完成一个往返的传输。CPU的时钟频率将受限于信号在介质上的传输时间。换言之,电子信号在真空中的传输速度是30cm/ns,而在铜线或光纤中的传输速度大约是20cm/ns。这意味着,对于时钟频率为1GHz的计算机,信号的路径长度不能超过200mm;对于时钟频率为100GHz的计算机,则不能超过2mm;对于时钟频率为1 000GHz的计算机,则不能超过100 μ m。显然,这对缩小元器件体积的要求越来越高。

2. 引入 MPS (multiprocessor system, 多处理机系统) 的原因有哪些?

【参考答案】引入MPS的原因有以下几点。

(1) 解决CPU的时钟频率问题。在早期,人们首先会采用提高CPU时钟频率的方法来提高计算速度。目前,这种通过提高CPU时钟频率来提高计算机计算速度(系统性能)的方法已接近极限。

(2) 增加系统吞吐量。随着系统中处理机数目的增加,系统的处理能力也相应增强。显然,这可使系统在单位时间内完成更多的工作,即增加了系统吞吐量。

(3) 节省投资。在达到相同处理能力的情况下,与 n 台独立的计算机相比,采用具有 n 个处理机的系统,可以更加节省费用。这是因为,此时的 n 个处理机可以做在同一个机箱中,使用同一个电源和共享一部分资源,如外设、内存等。

(4) 提高系统可靠性。MPS通常具有系统重构的功能,即当其中任何一个处理机发生故障时,系统可以进行重构,然后继续运行。

3. 什么是紧密耦合 MPS 和松散耦合 MPS ?

【参考答案】

紧密耦合MPS,通常是通过高速总线或高速交叉开关来实现多个处理机之间的互连的。系统中的所有资源和进程都由OS实施统一的控制和管理。这类系统有两种实现方式:①多处理机共享内存系统和I/O设备,每个处理机都可以对整个存储器进行访问,访问时间一般为10ns~50ns;②将多处理机与多个存储器分别相连,或将内存划分为若干个能被独立访问的存储器模块,每个处理机对应一个存储器或存储器模块,而且每个处理机只能访问其所对应的存储器或存储器模块,以便多个处理机能同时对内存进行访问。

松散耦合MPS,通常是通过通道或通信线路来实现多台计算机之间的互连的。每台计算机都有自己的存储器和I/O设备,并配置了OS来管理本地资源和在本地运行的进程。因此,每台计算机都能独立地工作,必要时可通过通信线路与其他计算机交换信息,以及协调它们之间的工作。但在这种类型的系统中,消息传递的时间一般为10ms~50ms。

4. 何谓 UMA (uniform memory access, 内存统一访问) 多处理机结构? 它又可进一步分为哪几种结构?

【参考答案】在UMA结构的MPS中, 各处理机单元在功能和结构上都是相同的, 在处理上没有主从之分, 即该MPS属于对称多处理 (symmetrical multi-processing, SMP) 系统, 每个处理机均可访问不同模块中的存储器单元, 并且对于每个存储器单元的读/写速度是相同的。根据处理机与存储器模块的连接方式的不同, 可将UMA多处理机结构分为3种: ①使用单总线的SMP结构; ②使用多总线的SMP结构; ③使用单级交叉开关的系统结构。

5. 何谓使用单总线的SMP结构和使用多总线的SMP结构?

【参考答案】本题分步解答如下。

(1) 使用单总线的SMP结构如图1-10-1 (a) 所示。在这种结构的系统中, 多个CPU与一个集中的存储器相连, 所有CPU都通过公用总线访问同一个系统的物理存储器, 每个CPU均可访问不同存储器模块中的单元, 以及与其他CPU进行通信。这种结构的缺点在于可伸缩性有限。系统中所有CPU对存储器的访问, 都需要通过总线进行。可以通过为每个CPU配置一个高速缓存的方法解决此问题, 如图1-10-1 (b) 所示, 这些高速缓存可以很大限度地减少该CPU对总线的访问频率, 极大地减少总线上的数据流量, 以支持更多的CPU。

(2) 使用多总线的SMP结构。在这种结构的系统中, 所有的CPU不仅共享一个高速缓存, 还有一个本地私有的存储器, 如图1-10-1 (c) 所示。各CPU与本地的私有存储器、I/O设备通过本地总线连接, 系统再使用系统总线将不同CPU的本地总线进行连接, 并且将系统中的共享存储器连接在系统总线上。

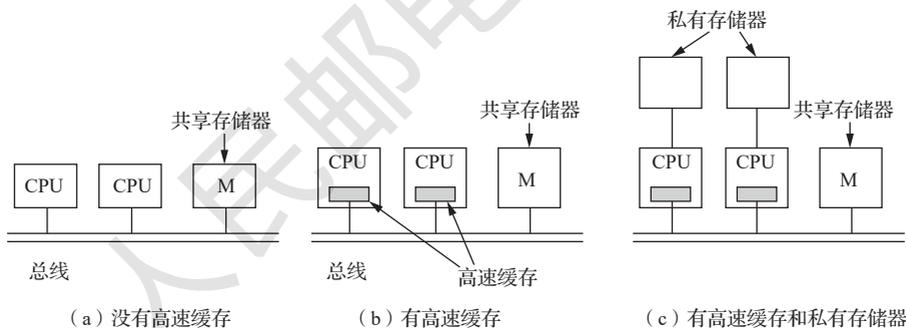


图 1-10-1 基于总线的 SMP 结构

6. 何谓使用单级交叉开关的系统结构和使用多级交换网络的系统结构?

【参考答案】本题分步解答如下。

(1) 使用单级交叉开关的系统结构。在这种结构中, 利用电话交换系统中使用交叉开关的方法, 如图1-10-2所示, 将系统中所有的CPU与存储器结点通过交叉开关阵列相互连接。每个交叉开关均为其中两个结点 (CPU与存储器模块) 之间提供一条专用连接通路, 从而避免了在多个CPU之间因为要访问存储器模块而形成对链路的争夺。

(2) 使用多级交换网络的系统结构。图1-10-3 (a) 是一个最简单的 2×2 交叉开关, 它有两个输入和两个输出。送入任一输入的信息可以交换到任一输出线上。可以将这样的 2×2 交叉开关分级连接起来, 形成多级交叉开关网络, 如图1-10-3 (b) 所示。

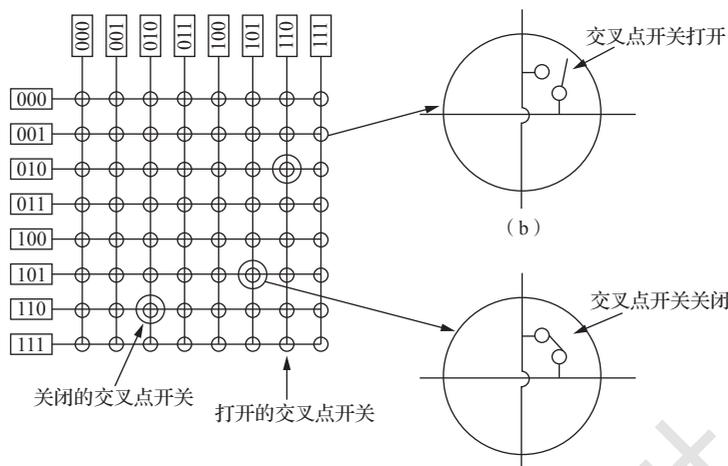


图 1-10-2 使用交叉开关的 UMAMPS

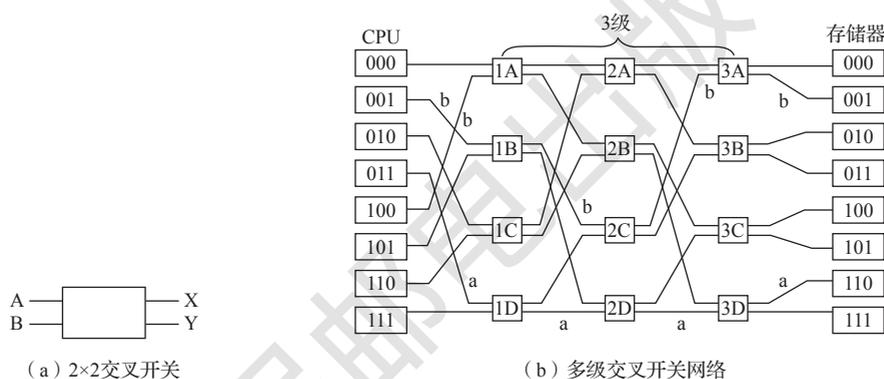


图 1-10-3 使用多级交换网络的 SMP 结构示意图

7. 何谓 NUMA (nonuniform memory access, 非统一内存访问) 结构? 它有何特点?

【参考答案】①NUMA也称为非一致存储访问。在NUMA结构的MPS中,系统访问时间会随存储字位置的不同而变化,系统中的公共存储器和分布在所有处理机上的本地存储器共同构成了系统的全局地址空间,其可被所有的处理机访问。②NUMA结构的特点是:所有共享存储器在物理上是分布式的,在逻辑上是连续的,所有这些存储器的集合就是全局地址空间;系统中的每个CPU都可以访问整个系统的内存,但访问时所使用的指令有所不同。

8. 在 NUMA 结构中,为什么要为每个 CPU 配置高速缓冲? CC-NUMA 和 NC-NUMA 分别代表什么?

【参考答案】①对于NUMA多处理机结构,其为每个CPU配备各自的高速缓存的目的是减少CPU对远程内存的访问次数。②将每个CPU配备有各自的高速缓存的结构称为CC-NUMA。将每个CPU没有配备各自的高速缓存的结构称为NC-NUMA。

9. 试说明多处理机 OS 的特征。

【参考答案】多处理机OS是在单机多道程序系统的基础上发展起来的,它们之间有许多相似之处,但也存在着较大的差异。归纳来说,多处理机OS具有以下几方面的新特征。

(1) 并行性。单机多道程序系统的主要目标是,为用户建立多个虚拟处理机以及模拟多处理机环境,使程序能并发执行,从而改善资源利用率并提高系统吞吐量。而在MPS中,由于存在多个实处理机,可使多个进程并行执行,因此,多处理机OS的主要目标应是进一步增强程序执行的并行性,以获得更高的系统吞吐量并提高系统的运算速度。

(2) 分布性。在单处理机系统中,所有的任务都是在同一个处理机上执行的,所有的文件和资源也都处于OS的统一管理之下。然而对于MPS而言,无论其结构如何,在任务、资源和对它们的控制等方面,都呈现出一定的分布性。这种情况,在松散耦合MPS中表现得尤其明显。

(3) 处理机之间的通信和同步性。在MPS中,不仅在同一个处理机上并发执行的各进程之间,由于资源共享和相互合作的需要,须实现同步和通信,而且在不同处理机上运行的不同进程之间,也需要同步和通信,这对于提高程序执行的并行性、改善系统性能至关重要。

(4) 可重构性。为提高系统的可靠性,在MPS中,应使OS具有这样的能力:当系统中某个处理机或存储器模块等资源发生故障时,系统能够自动切除故障资源、换上备份资源,并对系统进行重构,以保证其能继续工作。

10. 试比较单处理机 OS 和多处理机 OS 中的进程管理。

【参考答案】单处理机OS和多处理机OS中的进程管理在下列几个方面存在不同。

(1) 进程同步方面。在单机多道程序系统中,各进程只能交替执行,不会发生两个进程在同一时刻访问系统中同一个共享资源的情况。然而,在多机环境下,由于多个进程在不同的处理机上是并行执行的,因而可能出现多个进程对某个共享资源进行同时访问的情况。可见,多处理机OS中,不仅需要解决程序并发执行时引发的同步问题,而且需要解决多个不同的处理机程序并行执行时所引发的同步问题。因此,对于这两类进程同步问题的解决机制,除了采用通常的锁、信号量和管程外,还应采用新的同步机制和互斥算法。

(2) 进程通信方面。在单机环境中,所有进程都采用共享同一存储器的方式,驻留在同一台机器中。这样,进程间通信的主要方式是共享存储器方式和直接通信方式。但在多机环境中,相互合作的进程可能运行在不同的处理机上,它们之间的通信涉及处理机间的通信,特别是在松散耦合MPS中,进程甚至在不同的机器上,其间的通信还需要较长的通信信道或者需要经过网络。因此,在MPS中,进程通信的实现广泛地采用了间接通信方式。

(3) 进程调度方面。在单机环境中,进程调度只是简单地按照一定的算法,从就绪队列中选择一个进程并使之执行。为平衡I/O负载,可在调度时适当地进行I/O任务和计算任务的搭配,以提高系统的资源利用率。但在MPS中,发挥多处理机最大效能的关键在于增强程序执行的并行性。因此,在进程调度时,主要应考虑如何实现负载的平衡。在调度任务并为其分配处理机时,一方面必须了解每个处理机的能力,以便把适合的任务分配给它;另一方面也要确切地了解作业中各任务间的关系,即哪些任务必须顺序执行,哪些任务可以并行执行。

11. 试比较单处理机 OS 和多处理机 OS 中的内存管理。

【参考答案】在多处理机环境中,通常每个处理机都有属于自己的局部(本地)存储器,也有可供多个处理机共享的(系统)存储器。每个处理机访问局部存储器与访问系统存储器或其他处理机的局部存储器(统称远地存储器)相比,所花费的时间也可能是不同的。因此,在MPS中,存储器系统的结构十分复杂,这致使对存储器系统的管理也变得非常复杂,除了需要具有单机多道程序系统中的地址转换机构和虚拟存储器功能外,还应增强/增加下列功能/机制。

(1) 地址转换机构。该机构不仅用于将虚拟地址转换成物理地址, 还应能确定所访问的是局部存储器还是远地存储器。事实上, 在目前很多支持多处理机的OS中, 对整个存储器系统已经采用连续的地址方式进行描述, 即一个处理机无须专门去识别所要访问的存储器模块的具体位置。

(2) 访问冲突仲裁机构。当多个处理机上的进程同时竞争访问某个存储器模块时, 该机构能够按照一定的规则, 决定哪个处理机上的进程可立即访问, 哪个或哪些处理机上的进程应等待。

(3) 数据一致性机制。当共享内存中的某个数据在多个处理机的局部存储器中出现时, OS应保证这些数据的一致性。

12. 何谓中心同步实体、集中式同步机构和非集中式同步机构?

【参考答案】为实现进程之间的同步, 系统中必须有相应的同步实体, 如硬件锁、信号量以及进程等。如果该同步实体满足下述2个条件, 则称之为中心同步实体: ①具有唯一的名字, 并且为彼此必须同步的所有进程所知道; ②在任何时刻, 这些进程中的任何一个进程都可以访问该同步实体。基于中心同步实体所构成的所有同步机构被称为集中式同步机构; 相应地, 其他同步机构则称为非集中式同步机构。

13. 集中式同步算法具有哪些特征和缺点?

【参考答案】本题分步解答如下。

(1) 集中式同步算法具有2个特征: ①当多个进程需要同时访问共享资源或进行通信时, 仅由中心控制节点做出判定, 选择一个进程执行; ②判定所需要的全部信息都集中在中心控制节点。

(2) 集中式同步算法的缺点在于: ①可靠性差, 由于中心控制节点的故障会对系统造成灾难性的影响, 有的系统允许中心控制节点进行浮动, 即当其出现故障时, 系统立即选择一个新的节点作为中心控制节点; ②易形成瓶颈, 大量的资源共享和进程通信都是通过中心控制节点进行管理的, 这很容易使中心控制节点成为整个系统的瓶颈, 严重影响系统的响应速度和吞吐量。

14. 一个完全分布式同步算法应具有哪些特征?

【参考答案】一个完全分布式同步算法应具有以下特征: ①所有节点均具有相同的信息; ②所有节点仅基于本地信息做出判定; ③为了做出最后的判定, 所有节点担负相同的职责; ④为了做出最后的判定, 所有节点都要付出同样的工作量; ⑤通常一个节点发生故障, 不会导致整个系统崩溃。大多数同步算法都无法同时具备这5个特征, 因此完全分布式同步算法的实际应用很少。

15. 如何利用自旋锁实现对总线的互斥访问? 它与信号量的主要区别是什么?

【参考答案】利用自旋锁实现对总线互斥访问的方法是: 在总线上设置一个自旋锁, 该锁最多只能被一个内核进程持有。当一个内核进程需要使用总线对某个存储单元进行读/写访问时, 先请求自旋锁, 以获得对总线的使用权。如果该锁被占用, 那么这个进程就会一直进行“旋转”, 循环测试锁的状态, 直到自旋锁重新可用。如果锁未被占用, 则请求该锁的内核进程便能立刻得到它, 并继续执行, 直到完成对指定存储单元的读/写操作后, 释放该锁。可见, 自旋锁可以在任何时刻防止多个内核进程同时进入临界区, 因此其可有效避免多处理机上并发运行的内核进程对总线资源的竞争。

自旋锁与信号量的主要区别在于：自旋锁可避免调用进程阻塞。由于自旋锁使用者一般保持锁的时间非常短，调用进程可用“旋转”来取代进程切换。由于进程切换需要一定开销，并且会使高速缓存失效，直接影响系统的性能，因此将自旋锁应用于对总线资源的竞争，其效率要远高于信号量机制，且在多处理机环境中应用非常方便。

16. 为什么要引入读-复制-更新锁？它对读者和写者分别有何影响？

【参考答案】读/写自旋锁允许多个读进程同时读，但只要有一个写进程在写，便会禁止所有读进程去读，使读者进入阻塞状态。如果写的时间非常长，则将严重影响多个读进程的工作。解决上述问题的方法是改变写进程对文件（共享数据结构）进行修改（写）的方式。此即，当某写进程要往某文件中写入数据时，它先读该文件，将文件的内容复制到一个副本上，以后只对副本上的内容进行修改。修改完成后，在适当的时候再将修改完成后的文件全部写回去。

对读者和写者的影响。①读者不会被阻塞。读者在访问被读-复制-更新锁保护的共享数据时不会被阻塞，这一方面极大地提高了读进程的运行效率，另一方面也使读者所在的CPU不会发生上下文切换，减少了处理机的开销。②无须为共享文件（数据）设置同步机构。在采用读-复制-更新锁时，允许多个读者和一个或多个写者同时访问被保护的数据，而无须为共享数据设置同步机构，因此读者没有同步开销，也无须考虑死锁等问题。但是写者的同步开销却比较大，需要复制被修改的数据结构，延迟数据结构的释放，还必须使用某种锁机制，以与其他写者的修改操作同步执行。因此，如果读操作较多，而写操作较少，那么读-复制-更新锁就利大于弊；反之，则对读者的性能提高可能不足以弥补给写者所带来的损失，此时还是应当采用读/写自旋锁。

17. 何谓二进制指数补偿算法？它所存在的主要问题是什么？

【参考答案】①二进制指数补偿算法是指：为每一个CPU对锁进行测试的TSL指令设置一个指令延迟执行时间，使该指令下次在该延迟时间设定的时间后执行，该延迟时间是按照一个TSL指令执行周期的二进制指数方式增加的。例如，当一个CPU发出TSL指令对锁进行第一次测试而发现锁不空闲时，便推迟第二次测试指令的执行时间，等到 2^1 个指令执行周期后，如果第二次测试仍未成功，则将第三次测试指令的执行时间推迟到 2^2 个指令执行周期后……如果第 $n-1$ 次测试仍未成功，则将第 n 次的测试推迟到 2^{n-1} 个指令执行周期后，依次类推，直至次数达到设定的最大值。当锁释放时，可能首先会由延迟时间最小的CPU获得该锁。②该算法存在的主要问题是：当锁被释放时，可能由于各CPU测试指令的延迟时间未到，没有一个CPU会及时对锁进行测试，即不能及时发现锁空闲，进而造成浪费。

18. 时间戳定序机构和事件计数的作用分别是什么？

【参考答案】本题分步解答如下。

（1）时间戳定序机构。对该机构最基本的要求是，在系统中应具有唯一的、由单一物理时钟驱动的物理时钟体系，以确保各处理机时钟间的严格同步。其基本功能是：①对所有特殊事件，如资源请求、通信等，均加印时间戳；②对每一种特殊事件，只能使用唯一的时间戳；③根据事件上的时间戳，定义所有事件的全序。利用时间戳定序机构，再配以相应的算法，可实现不同处理机的进程同步。实际上，许多集中式和分布式同步方式，都将时间戳定序机构作为同步机构的基础。

（2）事件计数。在这种同步机构中，利用定序器为所有特定事件排序。定序器的初值为0，且为非减少的，对其仅能施加ticket操作。当一个事件发生时，系统便会为之分配一个称为编

号（或标号）V的序号，然后使ticket自动加1，一系列的ticket操作形成了一个非负的、增加的整数序列，然后把打上标号的事件送至等待服务队列排队。与此同时，系统将所有已服务事件的编号保留，并形成称为事件计数E的栈。实际上，E是保存已出现的某特定类型事件编号计数的对象（Object），其初值为0，当前值是栈顶的编号。

19. 什么是任务流时间和调度流时间？举例说明。

【参考答案】本题分步解答如下。

（1）任务流时间。把完成任务所需要的时间定义为任务流时间。如图1-10-4所示，图中有3个处理机 $P_1 \sim P_3$ 和5个任务 $T_1 \sim T_5$ ，调度从时间0开始，共运行了7个时间单位，在处理机 P_1 上运行任务 T_1 和 T_2 ，分别需要5个和1.5个时间单位；在处理机 P_2 上运行任务 T_1 和 T_2 ，分别用了5个和2个时间单位；在处理机 P_3 上运行任务 T_3 、 T_4 和 T_5 ，它们每个都需要2个时间单位。因此，完成任务 T_1 共需要 $5+2=7$ 个时间单位，而完成任务 T_2 共需要 $1.5+5=6.5$ 个时间单位。

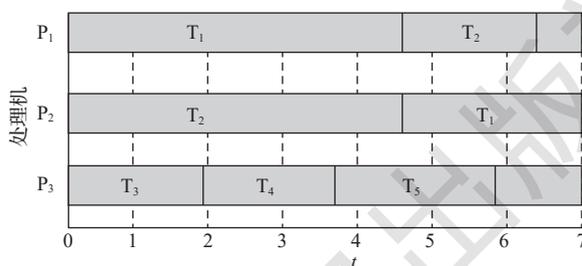


图 1-10-4 任务流和调度流示意

（2）调度流时间。在MPS中，任务可以被分配到多个处理机上运行。一个调度流时间是系统中所有处理机上的任务流时间的总和。在图1-10-4所示的例子中，3个处理机上的调度流时间 $=T_1$ 流 $+T_2$ 流 $+T_3$ 流 $+T_4$ 流 $+T_5$ 流 $=7+6.5+2+2+2=19.5$ 个时间单位。

20. 试比较多处理机 OS 中的静态分配方式和动态分配方式。

【参考答案】①静态分配方式是指一个进程从开始执行直至完成，都被固定地分配到一个处理机上去执行。此时，须为每一处理机设置一专用的就绪队列，该队列中的各进程先后都是被分配到该处理机上执行的。在进程阻塞后再次就绪时，其仍会被挂在这个就绪队列中，因而下次它仍在此处理机上执行。这种方式与单机环境下的进程调度一样。其优点是进程调度的开销小；缺点是会使各处理机的忙闲不均。换言之，系统中有些处理机的就绪队列可能很快就变成了空队列，即处理机处于空闲状态，而有些处理机则一直忙碌。②动态分配方式是指为了防止系统中的多个处理机忙闲不均，在系统中仅设置一个公共的就绪队列，系统中的所有就绪进程都被放在该队列中。分配进程时，可将进程分配到任何一个处理机上。这样，对一个进程的整个运行过程而言，在每次被调度执行时，进程都会被随机地分配到当时是空闲的某一处理机上去执行。

21. 何谓自调度方式？该方式有何优缺点？

【参考答案】本题分步解答如下。

（1）自调度方式，是由单处理机调度方式演变而来的，即在系统中设置有一个公共的进程或线程就绪队列，所有的处理机在空闲时都可到该队列中取得一进程（或线程）加以运行。在自调度方式中，可采用在单机环境下所用的调度算法，如FCFS调度算法、最高优先级优先调度算法等。

(2) 优点。①方便移植。系统中的公共就绪队列可按照单处理机系统中所采用的各种方式进行组织,其调度算法也可沿用单处理机系统中所用的算法,因此容易将单机环境下的调度机制移植到MPS上,故它仍然是当前MPS中较常用的调度方式。②处理机利用率高。只要公共就绪队列不空,就不会出现处理机空闲的情况,也不会发生处理机忙闲不均的现象。

(3) 缺点。①瓶颈问题。整个系统只设置一个就绪队列供多个处理机共享,这些处理机必须互斥地访问该队列,这容易形成系统瓶颈。②低效性。当线程阻塞后再重新就绪时,它只能进入这唯一的就绪队列,但却很少有可能仍在阻塞前的处理机上运行。如果在每个处理机上都配有高速缓存,则这时在其中保留的该线程的数据将失效,而在该线程新获得的处理机上又须重新建立这些数据的复制。一个线程在其整个生命周期中可能要多次更换处理机,这使高速缓存的使用效率变得很低。③线程切换频繁。通常,一个应用程序中的多个线程都是相互合作的,但在采用自调度方式时,这些线程很难同时获得处理机而同时运行,这会使某些线程因合作线程未获得处理机运行而阻塞,进而被切换下来。

22. 何谓成组调度方式?按进程平均分配处理机时间和按线程平均分配处理机时间这两种方法,哪个更有效?

【参考答案】为了解决在自调度方式中线程被频繁切换的问题,提出了成组调度方式。该方式将1个进程中的1组线程分配到1组处理机上去执行。在成组调度时,可考虑按进程平均分配处理机时间或按线程平均分配处理机时间。

(1) 面向所有应用程序平均分配处理机时间。假定系统中有 N 个处理机和 M 个应用程序,每个应用程序中至多含有 N 个线程,则每个应用程序至多可有 $1/M$ 的时间去占有 N 个处理机。例如,有4个处理机与2个应用程序,其中,应用程序A中有4个线程,应用程序B中有1个线程。这样,每个应用程序可占用4个处理机 $1/2$ 的时间。图1-10-5(a)所示为此时处理机的分配情况,由图可看出,使用这种分配方式,在应用程序A运行时,4个处理机都在忙碌;而在应用程序B运行时,则只有1个处理机忙碌,其他3个空闲。因此,将有 $3/8$ 的处理机时间(即37.5%)被浪费。

| | 应用程序A | 应用程序B |
|------|-------|-------|
| 处理机1 | 线程1 | 线程1 |
| 处理机2 | 线程2 | 空闲 |
| 处理机3 | 线程3 | 空闲 |
| 处理机4 | 线程4 | 空闲 |

1/2 1/2

(a) 处理机时间浪费37.5%

| | 应用程序A | 应用程序B |
|------|-------|-------|
| 处理机1 | 线程1 | 线程1 |
| 处理机2 | 线程2 | 空闲 |
| 处理机3 | 线程3 | 空闲 |
| 处理机4 | 线程4 | 空闲 |

4/5 1/5

(b) 处理机时间浪费15%

图1-10-5 两种分配处理机时间的方法

(2) 面向所有线程平均分配处理机时间。由于应用程序A中有4个线程,应用程序B中只有1个线程,因此,为应用程序A分配 $4/5$ 的时间,而只为应用程序B分配 $1/5$ 的时间,如图1-10-5(b)所示。此时,将只有15%的处理机时间被浪费。可见,按线程平均分配处理机时间的方法更有效。

23. 试说明采用专用处理机分配方式的理由。

【参考答案】首先,在具有数十个乃至数百个处理机的高度并行的系统中,每个处理机的投资费用在整个系统中只占很小一部分。对系统的性能和效率来说,单个处理机的利用率已远不像在单机系统中那么重要。其次,在一个应用程序的整个运行过程中,由于每个进程或线程

专用一个处理机，因此可以完全避免进程或线程频繁切换，这加速了程序的运行。

24. 在动态调度方式中，调度的主要责任是什么？在调度时应遵循哪些原则？

【参考答案】动态调度方式允许进程在执行期间动态地改变其线程的数目。这样，OS和应用程序能够共同地进行调度决策。OS负责把处理机分配给作业，而每个作业负责将分配到的处理机再分配给自己的某一部分可运行任务。在该方式中，OS的调度责任主要限于处理机的分配，并遵循空闲则分配、新作业绝对优先、保持等待、释放即分配等原则。

第11章 虚拟化和云计算

1. 何谓虚拟化？为什么要引入虚拟化？

【参考答案】①虚拟化是指把实体计算机的物理资源抽象成逻辑资源，基于这些逻辑资源构建与实体计算机架构类似、功能等价的逻辑计算机，这些逻辑计算机称为虚拟机（virtual machine, VM）。②虚拟化的基本思想是分离软硬件以产生更好的系统性能。引入虚拟化的目的是在多用户之间共享资源并提高资源利用率和应用程序灵活度。

2. 请简述虚拟化的发展过程。

【参考答案】该问题分步解答如下。

（1）早期的VM/370。IBM公司开发的一个分时系统VM/370，其核心为虚拟机管理程序，又称为VMM（virtual machine monitor，虚拟机监视器），它在裸机上运行并且具备多道程序功能。该系统向上层提供若干个虚拟机，它们仅仅是裸机硬件的精确复制品。这个复制品包含内核态/用户态、I/O功能、中断及其他真实硬件所应具备的全部功能。

（2）新时代虚拟机。20世纪90年代，为了解决在Intel 80x86 CPU上运行多个Windows XP的需求，VMware公司采用了一种新的虚拟化技术，并开发了可运行Windows XP的应用程序。该应用程序可以运行一个或多个Windows XP或其他用于Intel 80x86的客户机OS，而每个客户机OS都可运行自己的应用程序。原物理机上的Windows XP为主机OS，而VMware应用程序则为VMM。客户机OS安装在一个虚拟盘上，该盘实际只是主机OS文件系统中的一个大文件。

（3）Java虚拟机。在SUN公司发明Java程序设计语言时，也同时发明了称为JVM（Java virtual machine）的虚拟机。引入JVM后，Java语言在不同平台上运行时不再需要重新编译。

（4）虚拟化发展。在笔记本和台式计算机上，VMM允许用户安装多个OS来用于研究工作，或运行为客户机OS而编写的应用程序。为多个客户机OS编写应用程序的公司可以采用虚拟化技术，在单个物理服务器上运行多个OS，以便开发、测试和调试应用程序。在数据中心，虚拟化技术常常用于运行和管理计算环境。

3. 实现虚拟化的主要技术有哪些？请简要说明。

【参考答案】根据实现原理的不同，虚拟化技术可分为全虚拟化和半虚拟化这两大类。此外，利用对虚拟化提供支持的硬件特性实现虚拟化的技术被统称为硬件辅助虚拟化，其在全虚拟化和半虚拟化方案中都有体现。下面分别介绍这3种技术。

（1）全虚拟化，是一种利用纯软件实现的方法。VMM为虚拟机模拟出硬件环境，接收虚拟机的硬件请求，并将其转发到真正的硬件上。

(2) 半虚拟化, 又称为“协同虚拟化”, 通过修改供虚拟机使用的硬件抽象来避开硬件中存在的虚拟化漏洞, 并在客户机OS中加入虚拟化指令, 使客户机OS可以请求VMM帮助访问硬件。

(3) 硬件辅助虚拟化, 是通过虚拟化开发出新的硬件特性来简化虚拟化技术的。例如VT-x和AMD-V, 两者都针对特权指令为CPU添加了一个执行模式, 即使VMM运行在一个新增的根模式下。特权和敏感指令调用都会自动陷入虚拟化层, 而不再需要翻译或半虚拟化。虚拟机的状态保存在虚拟机控制结构或虚拟机控制块中。

4. 举例说明硬件辅助虚拟化与全虚拟化的结合。

【参考答案】全虚拟化的实现与计算机硬件结构具有强相关性。早期的ARM处理器在设计硬件结构时缺乏对虚拟化技术的支持, 纯软件的全虚拟化技术实现复杂, 且运行效率低下。为了解决这一问题, 硬件辅助技术应运而生。基于ARM v8.4架构的鲲鹏处理器加入了异常级别的设计、指令集扩展及专用寄存器等面向虚拟化技术的硬件支持。硬件辅助降低了VMM实现的复杂度, 以往依赖纯软件进行的复杂操作可直接使用专用指令让硬件自动执行, 这使虚拟机的运行效率及稳定性得到了提升。

5. 何谓 VMM? 它主要有哪些类型?

【参考答案】VMM是虚拟化技术中的虚拟化层, 用来实现对计算机硬件资源的模拟、隔离和共享。从软件架构的角度看, VMM主要有两种类型: Type 1 VMM(裸金属架构VMM)和Type 2 VMM(寄居架构VMM)。①Type 1 VMM, 直接运行在硬件之上, 管理底层硬件, 并监管上层虚拟机。②Type 2 VMM, 是PC上常用的类型, 这类VMM作为一个应用程序运行在主机OS上, 不会直接管理硬件资源, 而是会利用主机OS与硬件进行交互。

6. 何谓 CPU 虚拟化? 如何实现 CPU 虚拟化?

【参考答案】CPU虚拟化是将实际存在的硬件CPU虚拟成逻辑上的CPU。各虚拟CPU之间相互隔离, 并且都能像硬件CPU一样正确执行指令以及处理中断和异常。

要想实现CPU虚拟化, 则要对指令执行、中断和异常、上下文切换等进行模拟。它们分别描述如下。

(1) 指令执行。为了保证在VMM对CPU完全控制的同时, 虚拟机能够正确、快速地运行, 在指令执行方面须保证虚拟机能受限制地执行。具体实现思想是: 普通CPU指令直接执行, 以保证计算性能; 敏感指令由VMM模拟执行, 以保证安全。

(2) 中断和异常。为了使虚拟机能够正常响应中断和异常, 可采用VMM模拟方式为虚拟机提供与硬件环境一致的中断和异常触发条件与处理过程, 以使虚拟机觉察不到其所处环境为虚拟环境。

(3) 上下文切换。虚拟机之间采用时分复用的方式共享CPU, 因此虚拟机之间的切换需要解决上下文切换问题。VMM可控制虚拟机进行上下文切换: 首先保存当前运行虚拟机的上下文, 然后恢复即将运行虚拟机的上下文。

7. 试比较解释执行技术、扫描与修补技术以及二进制翻译技术的优缺点。

【参考答案】本题分步解答如下。

(1) 解释执行技术是指将虚拟机所须执行的每一条指令都经VMM实时解释执行。VMM把指令解释成能够在主机上运行的一个函数, 而这个函数可以模拟出虚拟机希望的执行效果。该技术的优点是所有指令都在VMM的监控之下; 缺点是效率低, 原本CPU在一个机器周期内就可

以执行完成的普通指令，变成了复杂且耗时的内存读/写操作。

(2) 扫描与修补技术是指扫描虚拟机所须执行的代码，保留普通指令并修补敏感指令。修补是指将指令中的敏感指令替换成一个外跳转，在跳转到VMM空间后，再执行可以模拟敏感指令效果的安全代码块，然后跳回虚拟机继续执行下一条指令。相比解释执行技术，该技术的效率提高了许多，但每次需要执行敏感指令时都需要跳转，这导致代码的局部性较差，限制了运行效率的进一步提高。

(3) 二进制翻译技术是指在启动虚拟机时，VMM就会预先将后续可能会用到的代码进行翻译并存储在缓冲区中。翻译时，保留普通指令、替换敏感指令，最后形成一段可直接按顺序执行的代码。该技术的优点是缓冲区中翻译好的代码局部性较高，直接执行速度更快；缺点是占用内存空间较大。

8. VMM 控制虚拟机进行上下文切换的主要实现方式有哪些？

【参考答案】VMM控制虚拟机进行上下文切换的主要实现方式有两种：软件切换和基于硬件支持的切换。①在软件切换中，VMM会维护一块内存，用于保存各虚拟机的上下文。虚拟机进行切换时，先将当前CPU所有寄存器的值存入内存空间，再从内存中读取即将运行的虚拟机在挂起前所保存的CPU寄存器值，并将其装入对应的寄存器中。②当硬件虚拟化技术出现后，硬件提供了便于虚拟机进行上下文切换的专用数据结构和指令，使得上下文切换的过程更加高效。以Intel-VT为例，其设计了虚拟机控制结构VMCS，用于保存虚拟机中所有vCPU的各种状态参数和操作策略。VMCS其实是物理内存中的一段有特定格式的内存空间，可通过专用指令对其进行控制。

9. 试说明内存虚拟化中两级内存映射的过程。

【参考答案】虚拟内存的虚拟化包括共享RAM中的物理内存以及为虚拟机动态分配内存。这就需要客户机OS和VMM维护从虚拟内存到物理内存的映射和从物理内存到机器内存的映射。客户机OS仍然负责从虚拟内存到虚拟机的物理内存的映射，但是它不能直接访问机器内存。VMM负责将物理内存映射到机器内存上。图1-11-1给出了两级内存映射的过程。

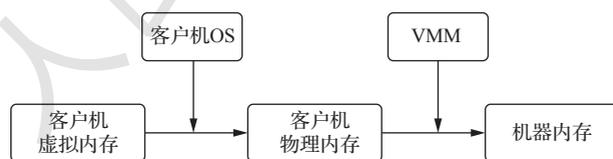


图 1-11-1 两级内存映射过程

10. 请简述全设备模拟、半虚拟化和直接 I/O 虚拟化这 3 种设备虚拟化技术。

【参考答案】本题分步解答如下。

(1) 全设备模拟可以在软件中复制一个设备的所有功能或总线结构。该软件作为虚拟设备处于VMM中，客户机OS的I/O访问请求会陷入VMM中，并与I/O设备进行交互。单一的硬件设备可以由多个同时运行的虚拟机共享。全设备模拟软件的运行速度会显著慢于其所模拟的硬件。

(2) 半虚拟化由前端驱动和后端驱动组成。前端驱动负责管理客户机OS的I/O请求，后端驱动负责管理真实的I/O设备并复用不同虚拟机的I/O数据。与全设备模拟相比，半虚拟化的方法可以获得更好的设备性能，但也会产生更高的CPU开销。

(3) 直接I/O虚拟化是让虚拟机直接访问硬件设备。它能获得近乎访问本地硬件的性能, 并且CPU开销不高。然而, 该技术目前主要应用在大规模主机的网络方面, 对于商业硬件设备而言仍有许多挑战。例如, 当一个物理设备被回收以备后续再用时, 它可能被设置为一个未知状态, 这可能会导致系统工作不正常, 甚至让整个系统崩溃。

11. 实现多核虚拟化的主要困难有哪些?

【参考答案】多核处理机通过在一个单一芯片上集成多个处理机核而具有更高的性能, 但多核虚拟化对计算机体系结构工程师、编译器编写者、系统设计者以及程序员等都提出了许多新的挑战。主要有2个困难: ①应用程序编写者必须完全并行地使用所有处理机核; ②软件必须明确地为处理机核分配任务。

12. 何谓云计算? 为什么要引入云计算?

【参考答案】①云计算是分布式计算、网格计算、并行计算、效用计算、网络存储技术、虚拟化、负载均衡等传统计算机技术和网络技术发展融合的产物, 旨在通过网络把多个成本相对较低的计算实体整合成一个具有强大计算能力的完美系统, 并借助SaaS(软件即服务)、PaaS(平台即服务)、IaaS(基础设施即服务)等先进的商业模式把这种强大的计算能力分布到终端用户手中。②云计算的核心理念是通过不断提高“云”的处理能力来减少用户终端的处理负担, 最终使用户终端简化成一个单纯的I/O设备, 并能按需享受“云”的强大计算能力。美国国家标准与技术研究院给出的云计算定义为: 云计算是一种按使用量付费模式, 这种模式提供可用的、便捷的、按需的网络访问, 用户进入可配置的计算资源共享池(含网络、服务器、存储设备、应用软件以及服务等)后, 其中的资源可被快速提供, 用户只须投入很少的管理工作或服务供应商进行很少的交互即可。

13. 试对广义的云计算和狭义的云计算进行比较。

【参考答案】①狭义的云计算是指IT(information technology, 信息技术)基础设施的交付和使用模式, 即通过网络以按需、易扩展的方式获得所需的资源。提供资源的网络被称为“云”。“云”中的资源在使用者看来是可以无限扩展的, 并且可以随时获取、按需使用、按使用收费。这种特性经常被称为像使用水和电一样使用IT基础设施。②广义的云计算是指服务的交付和使用模式, 即通过网络以按需、易扩展的方式获得所需的服务。这种服务可以与IT、软件、互联网等相关, 也可以是任意其他形式的服务。

14. 云计算的基本特征有哪些?

【参考答案】美国国家标准与技术研究院对云计算的定义中指出了其5个特征。①按需自助服务: 消费者无须同服务供应商交互就可以在需要时得到自助的计算资源, 如服务器时间、网络存储空间等(资源的自助服务)。②无处不在的网络访问: 用户可借助不同的客户端, 通过标准的应用进行网络访问以获取可用服务。③共享资源池: 根据消费者的需求来动态划分或释放不同的物理资源和虚拟资源。④快速弹性: 云计算具有快速、弹性地提供和释放资源的能力; 用户可在任何时间以任何量化方式对其进行购买。⑤服务可计量: 云计算能以计量的方法对服务类型进行自动控制、对资源进行优化使用。

15. 什么是静态迁移? 请简述它的处理过程。

【参考答案】静态迁移是指在虚拟机关机或暂停的情况下, 将相关文件从一台物理机迁移到另一台物理机上。在虚拟机关机的情况下, 只需要简单地迁移虚拟机镜像和相应的配置文件到另一台物理机上即可。如果需要保存虚拟机迁移之前的状态, 则须在迁移之前将虚拟机暂

停，然后复制其状态至目的主机，最后在目的主机上重建虚拟机状态，并恢复执行。这种方式的迁移过程需要显式地停止虚拟机的运行。静态迁移的步骤为：①复制虚拟机的镜像文件和配置文件；②将镜像文件和配置文件复制到目标虚拟机相应的目录中；③激活虚拟机配置文件；④开启虚拟机电源，启动迁移后的虚拟机。

16. 什么是动态迁移？它的处理步骤有哪些？

【参考答案】动态迁移也叫作在线迁移（分为手动和自动两种），是指在虚拟集群中，当一个虚拟机失效时，在不停止虚拟机的前提下，将故障转移至另一台主机的虚拟机上的过程。这一过程实现的前提是这两个虚拟机要运行相同的客户机OS。在线迁移虚拟机具体包括以下6个步骤。

步骤0和步骤1：预迁移与预定。确定要迁移的虚拟机和目标主机。尽管用户可以手动将一台虚拟机迁移到一台合适的物理主机上，但在大部分情况下，迁移是因负载均衡和服务器合并等策略而自动发起的。

步骤2：迭代预复制。由于虚拟机的整个执行状态都存储在内存中，因此向目标节点发送虚拟机的内存数据可以确保虚拟机提供服务的连续性。第一轮会传输所有的内存数据，后续的传输会不断地迭代复制刚更新过的数据，该过程重复进行，直至脏的内存页足够少。在该步骤中，尽管一直在迭代复制内存数据，但并不中断程序的运行。

步骤3：挂起并复制。在最后一轮传输内存数据时，挂起正在被迁移的虚拟机。其他非内存数据（如CPU和网络的状态等）也被同时发送。在该步骤中，虚拟机停止且其他应用不再运行。这一段的不可用时间称为迁移的停机时间。应尽量缩短停机时间，以使用户无法察觉。

步骤4和步骤5：提交与重新激活。在复制了所有需要的数据之后，在目标主机上，虚拟机重新装载其整体，恢复在其中执行的程序，并继续为其提供服务。然后，网络连接被重新定向至新虚拟机，此时，对源主机的依赖被清除。最后，从源主机中移除原虚拟机。至此，整个迁移过程结束。

17. 请简述物理集群和虚拟集群的区别。

【参考答案】一组通过物理网络互连的物理机器称为物理集群。安装在一个或多个物理集群的分布式服务器上的多个客户虚拟机称为虚拟集群。在逻辑上，处于一个虚拟集群的客户虚拟机，通过一个跨越了多个物理网络的虚拟网络互联在一起。一个虚拟集群可以由多台物理机器构成，也可以由一个可运行在多个物理集群上的虚拟机构成。在虚拟集群中，虚拟机客户系统与物理主机系统并存，并且虚拟机运行在物理主机之上。

18. 请举例说明虚拟机的在线迁移过程。

【参考答案】图1-11-2所示为从主机A向主机B在线迁移虚拟机的过程。在迁移中，将虚拟机的状态文件从存储区域复制到物理主机上。

19. 虚拟机在线迁移方案的设计目标是什么？

【参考答案】当虚拟机运行在线服务时，在线虚拟机迁移方案的设计目标是 minimized 停机时间、网络带宽消耗及总迁移时间这3个指标。另外，在迁移过程中，还需要确保不会因资源（如CPU、网络带宽等）竞争而中断运行在同一物理主机上的其他活跃服务。

20. 软件授权的目的是什么？

【参考答案】软件授权是软件保护概念的延伸和发展。其目标对象同时涵盖了开发商和最终用户。其目的是在保护软件不被盗版的同时，为开发商创造更方便、更灵活的销售模式。

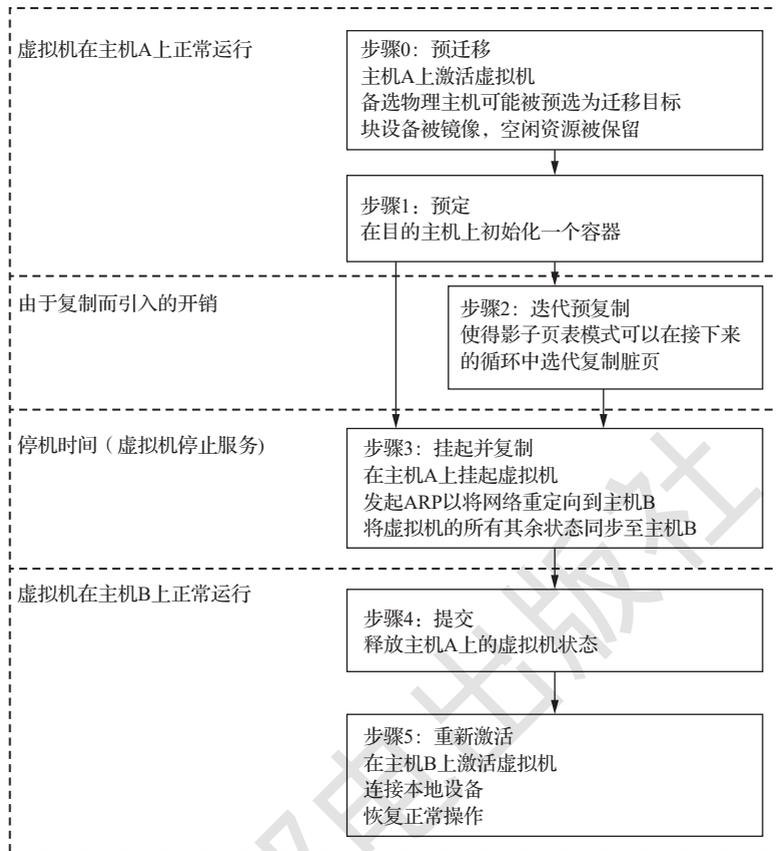


图 1-11-2 在线迁移虚拟机的过程

21. 运行在虚拟机上的软件如何被授权？举例说明。

【参考答案】大部分软件是基于某个处理机而授权的。在某些情况下，软件商会在许可证（license）中加入明确的条款，禁止在虚拟机或未授权的虚拟机中使用该软件。

以微软公司的Windows Server 2012 R2为例，它有2种基本的版本：标准版和数据中心。除了在针对不同的虚拟机时许可证不同外，它们各自的特点几乎相同。这意味着用户选择的版本取决于虚拟环境，而不是所需要的功能可用性。

（1）标准版的许可证只允许物理主机运行两个虚拟机，这两个虚拟机不能使用虚拟机自动激活技术。

（2）数据中心与许可证在一台物理主机上，此时物理主机可以运行虚拟机的数量不受限制，且这些虚拟机可以容易地使用虚拟机自动激活技术。

虚拟机不会直接授权。然而，如上所述，标准版的许可证允许一个许可的物理主机上运行达到2个OS的虚拟实例，数据中心的许可证允许主机上运行任意数量的OS的虚拟实例。

22. 云计算中授权的主要功能是什么？

【参考答案】云计算中的授权主要用于身份认证与访问管理，它是确定用户或系统身份并授予其权限的过程，被用于确定用户或系统是否具有执行某些操作所需的权限。在数字服务方面，授权是认证的下一步骤。授权管理是为了有效管理根据机构策略制定实体可访问资源权利的过程而进行的活动。

第12章 保护和安全

1. 实现“安全环境”的主要目标是什么？

【参考答案】实现“安全环境”的主要目标有数据机密性、数据完整性和系统可用性。

①数据机密性：指将机密的数据置于保密状态，仅允许被授权用户访问系统中的信息，以避免数据暴露。②数据完整性：指对数据或资源的可信赖程度，包括数据（信息内容）的完整性和数据来源的完整性，通常用于表述防止不当或未经授权的修改。③系统可用性：指能保证计算机中的资源供授权用户随时访问，系统不会拒绝服务。

2. 系统安全性的复杂性表现在哪几个方面？

【参考答案】系统安全性的复杂性表现在以下几个方面。

(1) 多面性：在较大规模的系统中，通常存在多个风险点，在这些风险点处又都包含物理安全、逻辑安全及安全管理这3方面的内容。其中任一方面出现问题，都可能引发安全事故。

(2) 动态性：由于IT不断发展和攻击者的手段层出不穷，系统的安全问题呈现出了动态性。

(3) 层次性：系统安全是一个涉及诸多方面且相当复杂的问题，因此需要采用系统工程的方法对其加以解决——利用多个层次的安全功能来覆盖系统安全的各个方面。

(4) 适度性：根据实际需要提供适度的安全目标并加以实现。

3. 对系统安全性的威胁可分为哪几类？分别介绍它们。

【参考答案】对系统安全性的威胁主要包括：假冒用户身份、数据截取、拒绝服务、修改信息、伪造信息、否认操作、中断传输、通信量分析。

(1) 假冒用户身份：也称为身份攻击，指用户身份被非法窃取，亦即攻击者伪装成一个合法用户，利用安全体制所允许的操作去破坏系统安全；在网络环境下，假冒者又可分为发方假冒和收方假冒两种。

(2) 数据截取：未经核准的用户可能会通过非正当途径截取网络中的文件和数据，由此造成网络信息的泄露。截取方式可以是直接从电话线上窃听，也可以是利用计算机和相应的软件来截取信息。

(3) 拒绝服务：指未经主管部门的许可，拒绝接受一些核准用户对网络资源进行访问。

(4) 修改信息：未经核准的用户不仅可能从系统中截取信息，还可能修改数据包中的信息。

(5) 伪造信息：未经核准的用户可将一些经过精心编造的虚假信息送入计算机，或者在某些文件中增加一些虚假记录，这同样会威胁系统中数据的完整性。

(6) 否认操作：这种类型的威胁又称为抵赖，是指某用户不承认自己曾经做过的事情。

(7) 中断传输：指系统中因某资源被破坏而造成信息传输中断，这将威胁系统的可用性。

(8) 通信量分析：攻击者可以通过窃听手段窃取在线路中传输的信息，考察数据包中的协议控制信息，了解通信者的身份与地址等；通过研究数据包的长度和通信频度，攻击者可以了解到所交换数据的性质。

4. 可信计算机系统评价准则将计算机系统的安全程度分为哪几个等级？

【参考答案】国际标准化组织将“IT安全评价通用准则（简称CC）”作为国际标准，CC为相互独立的机构对相应的IT安全产品进行评价提供了可比性。该标准将计算机系统的安全程度划分为8个等级： D_1 、 C_1 、 C_2 、 B_1 、 B_2 、 B_3 、 A_1 、 A_2 。其中， D_1 为安全度最低级， C_1 又称为自由安全保护级， C_2 又称为受控存取控制级，在后面的各级中，逐渐加强了对用户访问权限的限制。

5. 什么是易位法和置换法？试举例说明置换法。

【参考答案】易位法和置换法是两种最基本的加密方法。①易位法是按照一定规则，重新安排明文中的比特或字符顺序以形成密文，而字符本身保持不变。按易位单位的不同，易位法又可分成比特易位和字符易位两种。前者的实现方法简单易行，并且可用硬件实现，主要用于数字通信中；后者则是利用密钥对明文进行易位后形成密文，例如，将明文Please transfer转换为密文AFIISKSOSELAWAIA的加密过程。②置换法是按照一定规则，用一个字符去置换另一个字符以形成密文，例如，How are you? 中的每个字符用字母表中其后面第一个字母代替就是Ipx bsf zpv?

6. 试比较对称加密算法和非对称加密算法。

【参考答案】可以从表1-12-1所示的几个方面比较对称加密算法和非对称加密算法。

表1-12-1 对称加密算法和非对称加密算法的比较

| 比较内容 | 对称加密算法 | 非对称加密算法 |
|--------------|--|--|
| 密钥的保密性 | 密钥必须保密，故密钥的传递需要可靠的通道 | 公开密钥无须保密，可通过一般的通信环境传递；而私有密钥则无须传递 |
| 密钥的量 | 密钥量大，难以管理，如 N 个用户之间相互保密地传送数据，需要 $N(N-1)/2$ 个密钥 | 密钥量大大减少， N 个用户之间相互保密地传送数据，只需要 $2N$ 个密钥 |
| 加密速度 | 加密速度快，故适合大量数据的加密 | 加密速度慢，只适合少量数据的加密 |
| 互不相识的用户之间的通信 | 由于通信双方需要共享保密的密钥，故无法满足互不相识的用户之间的保密通信 | 由于公开密钥是公开的，故可以满足互不相识的用户之间的保密通信 |
| 数字签名 | 由于密钥至少是两人共享，故难以解决数字签名的验证问题 | 可以完成数字签名 |

7. 试说明保密数据签名的加密和解密方式。

【参考答案】保密数据签名的加密和解密方式如下：

- （1）发送者A用自己的私有密钥 K_{da} 对明文 P 进行加密，得到密文 $D_{K_{da}}(P)$ ；
- （2）发送者A再用接收者B的公钥 K_{eb} 对 $D_{K_{da}}(P)$ 进行加密，得到 $E_{K_{eb}}[D_{K_{da}}(P)]$ 后传送至接收者B；
- （3）接收者B收到 $E_{K_{eb}}[D_{K_{da}}(P)]$ 后，先用私钥 K_{db} 对其进行解密，得到 $D_{K_{db}}\{E_{K_{eb}}[D_{K_{da}}(P)]\}=D_{K_{da}}(P)$ ；
- （4）接收者B再用发送者A的公钥 K_{ea} 对 $D_{K_{da}}(P)$ 进行解密，得到 $E_{K_{ea}}[D_{K_{da}}(P)]=P$ 。

8. 数字证书的作用是什么？举例说明数字证书的申请、发放和使用过程。

【参考答案】CA（certification authority，认证机构）为公开密钥发放一份公开密钥证明书，又把该公开密钥证明书称为数字证书，用于证明通信请求者的身份。其申请、发放和使用过程如下：

（1）用户A在使用数字证书之前，应先向CA申请数字证书，此时A应向CA提供身份证明和希望使用的公钥A；

（2）CA在收到用户A发来的申请报告后，若决定接受其申请，则发给A一份数字证书，此证书中包括公钥A和CA发证者的签名等信息，然后利用CA的私钥对所有的这些信息进行加密，即对CA进行数字签名；

（3）用户A在向用户B发送报文信息时，由用户A用私钥对报文进行加密（即进行数字签名），然后连同已加密的数字证书一起发给B；

（4）为了能对所收到的数字证书进行解密，用户B须向CA申请获得CA的公钥B；CA收到用户B的申请后，可决定将公钥B发给用户B；

（5）用户B利用CA的公钥B对数字证书加以解密，以确认该数字证书系原件，并从数字证书中获得公钥A，然后确认该公钥A系用户A的密钥；

（6）用户B再利用公钥A对用户A发来的加密报文进行解密，进而得到用户A发来的报文的真实明文。

9. 可利用哪几种方式来确定用户身份的真实性？

【参考答案】由于身份认证是通过验证被认证对象的一个或多个参数的真实性和有效性来确定被认证对象是否名符其实的，因此，在被认证对象与要验证的那些参数之间，应存在严格的对应关系。身份认证目前主要依据3方面的信息来实现。①所知：基于个人所知道或所掌握的知识（如某系统的登录名、口令、密码等）进行身份认证。②所有：基于个人所具有的东西（如身份证、信用卡、钥匙等）进行身份认证。③个人特征：基于个人所具有的特征，特别是生理特征（如指纹、声音等）进行身份认证。

10. 基于口令机制的认证技术通常应满足哪些要求？

【参考答案】为了防止攻击者猜中口令，这种机制通常应满足以下几点要求。

（1）口令长度适中：口令通常是由一串字母和数字组成的。口令如果太短，则很容易被攻击者猜中。因此建议口令长度不少于7个字符，而且在口令中应包含大写字母和小写字母以及数字，最好还能引入特殊符号。

（2）自动断开连接：为了给攻击者猜中口令增加难度，在口令机制中还应引入自动断开连接的功能，即只允许用户输入有限次数的不正确口令，通常规定为3~5次。如果用户输入不正确口令的次数超过了规定的次数，系统便会自动断开该用户所在终端的连接。

（3）隐蔽回送显示：在用户输入口令时，登录程序不应将该口令回送到屏幕上显示，以防止被就近的其他用户发现。

（4）记录和报告：该功能用于记录所有用户登录系统和退出系统的时间，也用于记录和报告攻击者非法猜测口令的企图，以及所发生的与安全性有关的其他不轨行为，这样便能及时发现有人在对系统的安全性进行攻击。

11. 在口令机制中，应如何保证口令文件的安全性？

【参考答案】为了保证口令文件的安全性，可采取2种保护措施。

(1) 采用加密技术。首先,对口令文件中的口令部分必须采用加密技术加密后再进行保存。当验证口令时,系统须用同样的算法对用户输入的口令进行编码,并将编码后的口令与存储在口令文件中的已加密的口令进行比较。为了防止密钥泄露,对口令进行加密可采用单向加密的方式,即只能将口令的明文转换成密文,而不能将密文解密成明文。

(2) 采用强制性的保护措施。为了防止攻击者通过利用加密程序、采用字典攻击等方式来破译口令,还可采用强制性的保护措施,如仅允许OS本身访问口令文件,或者进一步强制为仅允许需要访问该表的系统模块访问口令文件。

12. 试给出一种对加密口令进行验证的方法。

【参考答案】当某用户输入一个口令时,系统利用函数 $f(x)$ 对该口令进行加密,然后将加密后的口令与存储在口令文件中的已加密的口令进行比较,若两者匹配,则认为该用户是合法用户。对攻击者而言,即使其能获取口令文件中已加密的口令,也无法对它们进行译码,因而不会影响系统的安全性。图1-12-1所示为对加密口令进行验证的方法。

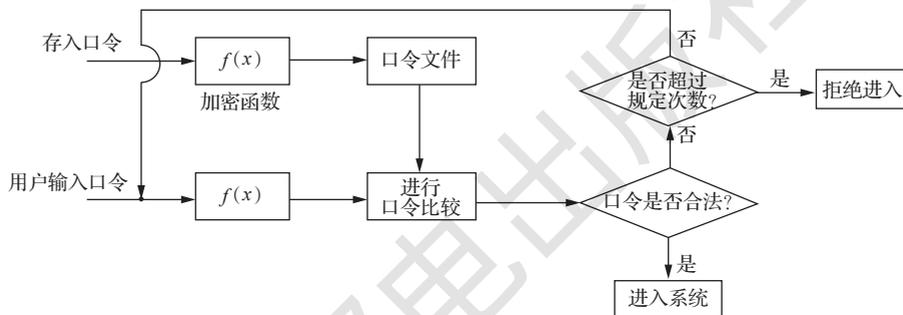


图1-12-1 对加密口令进行验证的方法

13. 基于物理标志的认证技术又可细分为哪几种?

【参考答案】基于物理标志的认证技术主要有基于磁卡的认证技术和基于IC卡(integrated circuit card, 集成电路卡)的认证技术这2种。

(1) 基于磁卡的认证技术。磁卡是基于磁性原理来记录数据的,如果在磁条上记录的是用户名、用户密码、账号和金额,则该卡便是金融卡或银行卡;如果在磁条上记录的是有关用户的信息,则该卡便可作为证明用户身份的物理标志。

(2) 基于IC卡的认证技术。在IC卡中可装入CPU和存储器芯片,使该卡具有一定的智能。IC卡中的CPU用于对内部数据进行访问和与外部数据进行交换,还可利用较复杂的加密算法对数据进行处理,这使IC卡比磁卡具有更强的防伪性和保密性。

14. 智能卡可分为哪几种类型?它们是否都可用于基于用户持有物的认证技术中?

【参考答案】智能卡分为存储器卡、微处理器卡和密码卡。①存储器卡:没有微处理器芯片,不具有安全性,只能作为储值卡,用来存储少量金额的现金与信息。②微处理器卡:增加了一个微处理器,增强了安全性,被广泛用作信用卡。用户可以在商场把信用卡插入读卡机后,授权进行一定数额的转账,然后信用卡会将一段加密后的信息发送到商场,商场再将该信息转发到银行,最后从用户在该银行中的账户中扣除所须付出的金额。③密码卡:增加了加密运算协处理器和RAM,由于它支持非对称加密算法,而且所支持的密钥长度可达1024位,因而极大地增强了IC卡的安全性。因此密码卡完全可以作为用户的数字身份证明。

15. 被选用于身份识别的生理标志应具备哪几个条件？请列举几种常用的生理标志。

【参考答案】被选用的生理标志应具备以下条件：①足够的可变性，系统可根据它来区别成千上万的不同用户；②保持稳定，通常不会发生变化；③不易被伪装。常用的生理标志有指纹、眼纹、声音、人脸等。

16. 对生物识别系统的要求有哪些？一个生物识别系统通常由哪几部分组成？

【参考答案】对生物识别系统的要求有：①识别系统的性能必须满足要求，包括应具有很强的抗欺骗和防伪造能力，而且还应能防范攻击者设置陷阱；②能被用户接受，完成一次识别的时间不应太长，一般不应超过1~2s；出错率应足够低，这一点随应用场合的不同而异；对于用在极为重要的场合中的识别系统，要求绝对不能出错，可靠性和可维护性也要好；③系统成本适当；系统成本包括系统本身的成本、运营期间所需的费用和系统维护（包含消耗性材料等）的费用。

生物识别系统通常由注册和识别两部分组成。①注册部分：配置有一张注册表，每个注册用户注册表中都有一个记录；记录中至少有两项，其中一项用于存放用户姓名，另一项用于存放用户的主要特征（用户的生物特征被数字化后形成用户样本，从中可提取出用户的主要特征）。该记录通常存放在中心数据库中，供多个生物识别系统共享，但也可存放在用户的身份智能卡中。②识别部分：第1步是要求用户输入用户登录名，这样可使系统尽快找到该用户在系统中的记录；第2步是对用户输入的生物特征进行识别，即对用户的生物特征与用户记录中的样本信息特征进行比较，若相同，则允许用户登录，否则，拒绝用户登录。

17. 早期通常采用的内部攻击方式有哪几种？

【参考答案】早期通常采用的内部攻击方式有下列6种。①窃取尚未清除的有用信息。在许多OS中，在进程结束而归还资源时，有的资源中可能还留存着非常有用的信息，但系统并未清除它们。攻击者为了窃取这些信息，会请求调用许多内存页面和大量的磁盘空间或磁带，以读取其中的有用信息。②通过非法的系统调用搅乱系统。攻击者尝试利用非法的系统调用，或者在合法的系统调用中使用非法参数（还可能使用虽是合法、但不合理的参数），达到搅乱系统的目的。③使系统自己封杀校验口令程序。攻击者为了逃避校验口令验证，在登录过程中可通过按DEL或者BREAK键等令系统封杀掉校验口令程序，此时用户无须输入口令便可成功登录。④尝试许多在明文规定中不允许做的操作。攻击者专门去执行系统不允许做的操作，企图破坏系统的正常运行。⑤在OS中增添陷阱门。攻击者要求某个系统程序员在OS中增添陷阱门，使攻击者可以绕过口令验证而进入系统。⑥骗取口令。攻击者可以伪装成一个忘记了口令的用户，请求系统管理员帮其查出某个用户的口令。

18. 何谓逻辑炸弹？较常用的引发逻辑炸弹爆炸的条件有哪些？

【参考答案】“逻辑炸弹”是指在特定逻辑条件被满足时实施破坏的计算机程序，该程序被执行后会造成计算机数据丢失、计算机不能从硬盘或者软盘进行引导，甚至会使整个系统瘫痪，并出现物理损坏的虚假现象。

常用的引发逻辑炸弹爆炸的条件有：①时间引发，即规定在一年中或一个星期中的某个特定的日期触发爆炸；②事件引发，即当所设置的事件发生时引发爆炸，比如发现了所寻找的某些文件；③计数器引发，即计数值达到所设置的值时引发爆炸。

19. 何谓陷阱门和特洛伊木马？举例说明。

【参考答案】本题分步解答如下。

(1) 陷阱门, 是一段代码, 可提供进入一个程序的隐蔽入口点。程序员通过陷阱门可跳过正常的验证过程。例如: 正常的登录程序代码如图1-12-2(a)所示, 该程序while循环中的最后两句的含意是, 仅当输入的用户名和口令都正确时, 用户才能登录成功。但如果将该程序while循环中的最后一条语句改成图1-12-2(b)所示的登录程序代码, 此时最后两句的含意变为, 当输入的用户名和口令都正确时, 或者使用的登录名为“zzzzz”而无论用什么口令时, 都能成功登录。

| | |
|--|--|
| <pre>while(TRUE){ printf("login:"); get_string(name); disable_echoing(); printf("password:"); get_string(password); enable_echoing(); v=check_validity(name, password); if(v) break; } excute_shell(name);</pre> | <pre>while(TRUE){ printf("login:"); get_string(name); disable_echoing(); printf("password:"); get_string(password); enable_echoing(); v=check_validity(name, password); if(v strcmp(name,"zzzzz")==0)break; } excute_shell(name);</pre> |
|--|--|

(a) 正常的登录程序代码

(b) 插入陷阱门后的登录程序代码

图 1-12-2 陷阱门实例

(2) 特洛伊木马, 是一个嵌入有用程序中的、隐蔽的、危害安全的恶意程序。当该程序执行时会引发隐蔽代码执行, 进而产生难以预料的后果。例如: 编写特洛伊木马的人将其隐藏在一个新游戏程序中, 并将该游戏程序发送给某计算机系统的操作员。操作员在玩新游戏时, 隐藏在后台运行的特洛伊木马就能继承操作员的高特权, 访问口令文件。再如: 在文本编辑程序中隐藏的特洛伊木马, 会把用户正在前台编辑的文件悄悄地复制到预先设定的某个地方, 以便以后能访问它; 但这并不会过分影响用户所进行的文本编辑工作, 用户也很难发现自己的文件已被复制。

20. 何谓移动代码? 为什么说在应用程序中包含了移动代码就可能不安全?

【参考答案】如果一个程序在运行时能在不同机器之间来回迁移, 那么该程序就被称为移动代码。如果一个用户程序中包含了移动代码, 那么当为该用户程序建立进程后, 该移动代码就将占用该进程的内存空间, 并作为合法用户的一部分运行, 此时其拥有用户的访问权限。这样显然不能保证系统安全, 因为别有用心的人完全可以借助移动代码进入其他系统, 以合法用户的身份进行窃取和破坏。

21. 什么叫缓冲区溢出? 攻击者是如何利用缓冲区溢出进行攻击的?

【参考答案】①如果程序员没有对某个数据结构进行边界检查, 而编译器也不对它进行边界检查, 那么就可以向该数据结构填充超过其本身容量的数据, 溢出的数据将覆盖在合法数据上, 这种情况叫作缓冲区溢出。②攻击者可以利用缓冲区溢出进行攻击, 例如, 在C语言中进行函数调用时, 会首先将函数调用的参数和函数的返回地址依次压入堆栈的栈顶; 如果函数体内还定义了一个局部变量(如一个数组), 则还将在堆栈栈顶为局部变量分配空间。攻击者可向该数组写入超过其容量的大量数据, 其中溢出的数据将覆盖掉保存在堆栈中的函数返回地址, 此时函数再返回时, 就将根据栈中的内容跳转到一个随机的地方, 从而引起程序崩溃; 如果攻

击者将一段恶意代码以及该恶意代码的起始地址写到该数组中，并且使恶意代码的起始地址刚好覆盖到函数的返回地址上，那么函数返回后便会执行该恶意代码，进而通过程序崩溃拒绝服务，或者利用恶意代码实施攻击。

22. 病毒和蠕虫有何异同？

【参考答案】病毒和蠕虫都是利用程序来威胁计算机系统安全的，它们都具有传染性、隐蔽性和破坏性，但病毒是一段代码，需要寄生在另一个合法程序中，而蠕虫则是一个独立的程序；病毒通过宿主程序的运行而感染给更多其他的程序，蠕虫则通过自我复制而传播到其他计算机中。

23. 计算机病毒的特征是什么？它与一般的程序有何区别？

【参考答案】计算机病毒与一般的程序有着明显的区别，其主要特征介绍如下。

(1) 寄生性：病毒通常不是一个独立的程序，而是会寄生在某个文件中或是磁盘的系统区中；寄生在文件中的病毒称为文件型病毒，而侵入磁盘系统区的病毒则称为系统型病毒；还有一种综合型病毒，它既能寄生在文件中，又能侵入磁盘系统区。

(2) 传染性：病毒在运行过程中将进行自我复制，并将复制品放置在其他文件中或磁盘的某个系统区中；文件被感染后便含有该病毒的一个克隆体，而这个克隆体也同样会传染给其他文件，如此不断地传染，就会使病毒迅速蔓延开来。

(3) 隐蔽性：为了逃避反病毒软件的检测，病毒设计者会通过伪装、隐藏、改变状态等手段将病毒隐藏起来，使病毒能在系统中长期生存。

(4) 破坏性：主要表现在占用系统空间、占用处理机时间、对系统中的文件造成破坏、使机器运行产生异常等方面。

计算机病毒与一般程序的区别：病毒通常不是独立的程序，它具有自我复制和迅速传播的特性，而且还会想方设法隐藏自己，其存在的基本目标就是搞破坏。

24. 什么是文件型病毒？试说明文件型病毒对文件的感染方式。

【参考答案】文件型病毒是指采用寄生方式附着在正常程序里，在病毒发作时不影响原来程序的正常运行，以致不能被用户及时发现而长期潜伏下来的病毒。受感染的程序在被执行时，病毒将寻找其他可执行文件继续传播。

文件型病毒使文件受感染的方式可分为两种。①主动攻击型感染：当病毒程序在执行时，它将不断地对磁盘上的文件进行检查，当发现被检查文件尚未被感染时，就去感染它，进而使它带有病毒。②执行时感染：在病毒环境中，每当一个未被感染的程序在被执行时，如果它是病毒所期待的文件类型，且磁盘没有写保护，则该程序就会被病毒感染；病毒在感染文件时，通常是有针对性的。

25. 病毒设计者可以采取哪几种隐藏方式来让病毒逃避检测？

【参考答案】病毒设计者通常采取3种隐藏方式来让病毒逃避检测。

(1) 伪装。为了逃避检测，病毒将把自己伪装起来，使被感染过的文件与原有文件一样。伪装方式主要有两种。①压缩伪装：通过压缩技术，使感染上病毒的文件长度与原文件的长度一致，以逃避检测。使用该方法时，在病毒中应包含压缩程序和解压缩程序。②修改日期和时间伪装：通过修改感染上病毒的文件修改日期和时间，使之与原文件相同。

(2) 隐藏。为了逃避反病毒软件的检测，病毒程序的设计者常把病毒隐藏在一个不易被检查到的地方。①隐藏于目录或注册表空间。目录的末端是一个隐藏病毒的好地方。②隐藏于程

序的页内碎片里。一个程序段和数据段可能被装入若干个页面中。通常在最后一页都会有页内碎片，因此，在系统中就可能存在许多页内碎片，病毒就可以隐藏在这些碎片中。当病毒隐藏在多个碎片中时，可由指针将这些碎片链接起来。③更改用于磁盘分配的数据结构。在这种方法中，病毒可以为真正的引导记录扇区和病毒自身重新分配磁盘空间，然后再更改用于磁盘分配的数据结构的内容，以使病毒合法地占据存储空间，这样既不会被发现，也不会被覆盖。④更改坏扇区列表。病毒可以更改真正的引导记录扇区和磁盘的任意空闲扇区，然后将这些扇区作为坏扇区，再相应地修改磁盘的坏扇区列表。

(3) 改变状态。为了逃避反病毒软件的检测，病毒设计者又设计了多状态病毒。常见的产生多状态病毒的方法有：①插入多余的指令；②对病毒进行加密。

26. 试说明基于病毒数据库的病毒检测方法。

【参考答案】基于病毒数据库的病毒检测方法介绍如下。①建立病毒数据库。为了建立病毒数据库，首先应当采集病毒样本，为此，设计了一个称为诱饵文件的程序，它能让病毒感染，但不会执行任何操作。用它来获取病毒的完整代码，然后将病毒的完整代码输入病毒数据库中，如此一来，病毒数据库中所收集的病毒样本的种类就会越来越多，利用它去检测病毒的成功率也就会越来越高。②扫描硬盘上的可执行文件。将反病毒软件安装到计算机上之后，便可对硬盘上的可执行文件进行扫描，检查硬盘上的所有可执行文件，看是否有与病毒数据库中的病毒样本相同的，如果有，则将其清除。

27. 如何利用数字签名验证信息内容？

【参考答案】要利用数字签名验证信息内容和确定签名人的身份，可采用以下方法。

(1) 发送方使用Hash函数（或称作散列函数）为自己的任意长度的消息生成一个简短的、固定长度的消息摘要（也称作信息标记）。

(2) 发送方使用自己的私有密钥对消息摘要进行加密，将加密结果作为自己的数字签名，并将其与消息原文一起发送给接收方。

(3) 接收方收到消息后，使用发送方的公开密钥对其中的数字签名进行解密，从而得到发送方消息摘要的副本。

(4) 接收方使用相同的Hash函数来计算接收到的消息摘要。

(5) 比较两个摘要，如果两者完全相同，则可确认数字签名是正确的，即发送方的消息在发送的过程中没有发生改变。