《操作系统》

----复习资料

数据科学与大数据技术

编写：小林

日期：2026.06.01

目录

第一章 计算机操作系统概述 1
1.1 计算机系统 1
1.2 计算机操作系统 1
1.3 深入观察操作系统 1
第二章 处理器管理 2
2.1 处理器 2
2.2 中断 2
2.3 进程管理 2
2.4 多线程技术 3
2.5 处理器调度 3
第三章 存储管理 3
3.1 存储管理基础 3
3.2 单连续分区存储管理 4
3.3 页式存储管理 4
3.4 段式存储管理 5
第四章 设备管理 5
4.1 设备管理基础 5
4.2 设备管理软件 5
4.3 独占型外围设备的分配 5
4.4 共享型外围设备的驱动 6
4.5 虚拟设备 6
第五章 文件管理 6
5.1 文件系统概述 6
5.2 文件的组织 6
5.3 文件目录 6
5.4 文件的共享、保护和保密 6
第六章 并发程序设计 7
6.1 并发程序设计 7
6.2 临界区管理 7
6.3 PV 操作（P207，看代码） 7

6.4 管程 7
6.5 进程通信 7
6.6 死锁 7

第一章 计算机操作系统概述

1.1 计算机系统

1. 计算机硬件系统的组成：中央处理器（CPU）、主存储器、外围设备等组件，通过系统总线连接。

2. 按照传输的信息对总线分类：控制线、数据线和地址线。

3. 按照功能对总线分类：内部总线（用于 CPU 芯片内部各元件的连接），系统总线（用于连接 CPU、存储器和各个 I/O 模块），通信总线（用于计算机系统之间的通信）。

4. 中央处理器的三个部分：运算逻辑部件、高速存储部件、控制部件。

5. 计算机系统有寄存器、Cache、主存储器、外存储器等多个层次的存储器。

1.2 计算机操作系统

1. 操作系统是计算机系统中最基础的系统软件，它统一管理软硬件资源，控制程序执行，改善人机界面，合理组织计算机工作流程，为用户使用计算机提供良好的运行环境。

2. 批处理系统

批处理操作系统服务于成为批的一系列作业。作业是把程序、数据连同作业说明书组织起来的任务单位。把批中的作业预先输入作业队伍中，由操作系统按照说明书的要求来调度和控制作业执行，大幅减少人工干预，形成自动转接和连续处理的作业流。其优点有系统资源利用率高、作业吞吐量大、批量集中处理、多道程序运行、用户脱机工作；缺点是作业周转时间延长、不具备交互式计算能力、不利于程序的开发和调试。

3. 分时操作系统（分时和实时的区别）

分时操作系统是目前最常使用的操作系统。其思路如下：多个联机用户同时使用一个计算机系统，在各自终端上进行交互式会话，程序、数据和命令均在会话过程中提供，以问答方式控制程序运行。系统把处理器的时间划分成时间片轮流分配给各个联机终端，若时间片用完则产生时钟中断，将控制权移交至操作系统并重新进行调度。其特点为：分时性、独立性、及时性、交互性。

4. 实时操作系统

实时操作系统是另一类应用范围较大的操作系统，用于满足实时控制和实时信息处理的需要。当外部事件或数据产生时，实时操作系统能够及时对其予以接收并以足够快的速度进行处理，即一方面针对单个实时任务保证在规定时间内控制其生产过程或对控制对象做出快速反应，另一方面能够协调安排所有实时任务的运行。它的主要特点是提供及时响应和高可靠性。

1.3 深入观察操作系统

1. 复用、虚拟和抽象是三种基本的资源管理技术

复用技术可以创造虚拟资源以解决物理资源数量不足的问题，包括空分复用共享和时分复用共享两种基本方法；虚拟技术是对资源进行转化、模拟或整合，把一个或多个物理资源转变成一个或多个逻辑上的对应物；资源抽象是指通过编制软件来屏蔽硬件资源的物理特性和实现细节，简化对硬件资源的操作、控制和使用，即这是一种不考虑物理细节而对资源执行操作的技术。

2. 多道程序设计的优点

多道程序设计允许多个程序同时进入计算机系统的主存，通过竞争处理器资源获得交替执行。它有诸多优点：一是提高 CPU、主存和设备的利用率；二是提高系统的吞吐率，使单位时间内完成的作业数量增加；三是充分发挥系统的并行性，使得设备与设备之间、CPU 与设备之间均可并行工作。

3. 操作系统提供的系统调用按照功能分为六类

进程管理、文件管理、设备管理、存储管理、进程通信、信息维护。

第二章 处理器管理

2.1 处理器

1. 内核态：又称管态，是操作系统管理程序运行时所处的状态，可认为处理器正在运行可信系统软件，此时全部机器指令都被允许在处理器上执行，程序可访问所有主存单元和系统资源，并具有改变处理器状态的能力。

2. 用户态：又称目态，当处理器处于用户态时，它正在运行非可信的应用程序，此时无法执行特权指令，且访问仅限于当前正在处理器上执行程序所在的地址空间。

3. **内核态与用户态通过中断进行切换**。

2.2 中断

1. 所谓中断，是指在程序执行过程中当遇到急需处理的事件时暂时中止现行程序在 CPU 上的运行，转而执行相应的事件处理程序，待处理完成后再返回断点或调度其他程序执行。

2. 中断源：引起中断的事件，即中断请求信号的来源

3. 中断事件的类型：硬件故障中断事件、程序性中断事件、自愿性中断事件、I/O 中断事件、外部中断事件。

4. 中断处理程序：保护处理器现场、分析被中断进程的 PSW 中断码并识别中断源、根据中断源对中断事件进行具体处理、恢复正常操作。

2.3 进程管理

1. 进程是既能描述程序的并发执行，又能共享系统资源的一个基本单位。

2. 进程的三种状态：运行态、就绪态、等待态。

3. 进程控制块（PCB）是进程存在的唯一标识，用来记录和刻画进程状态及环境信息，是进程动态特征的汇集，也是操作系统掌握进程的唯一资料结构和管理进程的主要依据。

4. 在操作系统中，进程物理实体和支持进程运行的环境合称为进程上下文，它由三部分组成：用户级上下文、计算机上下文、系统级上下文。

5. 进程切换实现步骤：保存被中断进程的处理器现场信息；修改被中断进程的进程控制块的有关信息，如进程状态等；把被中断进程的进程控制块加入相关队列；选择占用处理器运行的另一个进程；修改被选中进程的进程控制块的有关信息，如改为就绪态；设置被选中进程的地址空间，恢复存储管理信息；根据被选中进程的上下文信息来恢复处理器现场。

2.4 多线程技术

线程是进程中能够并发执行的实体，是进程的组成部分，也是处理器调度和分配的基本单位。进程包含多个线程，这些线程共享进程所获得的主存空间和资源，可以为完成某一项任务而协同工作。

2.5 处理器调度

1. 高级调度：又称作业调度、长程调度，从外村拿到主存；从输入完成的一批作业中按照预定的调度策略挑选若干作业进入主存，为其分配所需资源并创建对应作业的用户进程。

2. 中级调度：又称平衡负载调度、中程调度，完成外村和主存中的进程对换工作。

3. 低级调度：又称线程（进程）调度、短程调度，主要功能是根据某种原则决定就绪队列中的哪个进程/线程获得处理器，并将处理器分配给它使用。

4. 计算机系统的主要性能指标

1) 资源利用率

CPU 利用率等于 CPU 有效工作时间除以 CPU 总运行时间。

2) 吞吐率：单位时间内 CPU 处理作业的个数。

3) 公平性：确保每个进程都能获得合理的 CPU 份额和其他资源份额，不会出现“饥饿”现象。

4) 响应时间：从交互式进程提交一个请求（命令）直到获得响应之间的时间间隔。

5) 作业周转时间：批处理用户从向系统提交作业开始，到作业完成为止的时间间隔。

5. 优先数调度算法

1) 先来先服务（FCFS）：按照作业进入系统的时间顺序排序，常用于批处理系统的高级调度和分时系统的终端连接。

2) 最短作业优先（SJF）：优先处理计算时间短的作业有利于提高系统的吞吐率和平均周转率，但该原则对计算时间长的作业（进程）不公平，甚至会出现“饿死”现象。

3) 最短剩余时间优先（SRTF）：从已有作业队列中选择剩余完成时间最短的作业进行，类似 SJF，其不公平特征比 SJF 更加突出。

4) 最高响应比优先（HRRF）：响应比是等待时间与计算时间之比，这一算法能够改善 SJF 和 SRTF 对计算时间长的作业/进程极端不公平的问题。

6. 时间片轮转调度算法：它是一种进程调度算法，基于先来先服务调度算法，但是每个进程占有 CPU 运行必须受到时间片的限制，一旦进程运行时间超过了该时间片就得让出 CPU。

第三章 存储管理

3.1 存储管理基础

1. 逻辑地址与物理地址

目标代码程序存放于磁盘文件中，由于程序在主存中的位置不可预知，程序地址空间中的地址总是相对某个基准（通常为 0）开始连续编号，成为逻辑地址或相对地址。

CPU 执行指令和处理数据时面向的实际主存地址，称为物理地址或绝对地址。

2. 主存管理的四个模式：单连续分区存储管理、页式存储管理、段式存储管理、段页式存储管理。

3. 地址转换

地址转换又称为重定位，指把逻辑地址转换为物理地址的工作，分为静态地址重定位和动态地址重定位两种。

静态地址重定位由装入程序来实现代码模块的加载和地址转换，将程序与数据装入分配给进程的主存指定区域，并将其中的所有逻辑地址修改为主存物理地址。

动态地址重定位是主流的地址转换技术，装入程序只实现代码模块的加载，将程序与数据装入分配给进程的主存指定区域，对程序中的逻辑地址不做任何修改。CPU 在执行程序指令时，通过硬件的地址转换机制实现逻辑地址到物理地址的动态转换。

4. 主存扩充包括对换技术与虚拟技术两种方式

3.2 单连续分区存储管理

1. 单连续分区存储管理指每个进程占用物理上完全连续的存储空间。它可以细分为三种：单用户连续分区存储管理、固定分区存储管理和可变分区存储管理。

2. 常用的可变分区主存分配算法

1) 最先适配算法（first-fit）：该算法中未分配区表按起始地址从小到大链接，顺链接找到第一个能满足长度要求的空闲区，分割此分区，一部分分配给进程，另一部仍为空闲区（若有）。

2) 下次适配算法（next-fit）：该算法总是从未分配区的上次扫描结束处顺序查找未分配区表，直至找到第一个能满足长度要求的空闲区为止。

3) 最优适配算法（best-fit）：该算法中未分配区表按照分区长度从小到大链接，顺链找到第一个能够满足长度要求的空闲区，分割此分区，一部分分配给进程，另一部分仍为空闲区（若有）。

4) 最坏适配算法（worst-fit）：该算法中未分配区表按照分区长度从大到小链接，如第一个空闲分区能满足长度要求，分割此分区，一部分分配给进程，另一部分仍为空。

3.3 页式存储管理

1. 页式存储管理的逻辑地址由两部分组成：页号和单元号，其物理地址也由两部分组成：页框号和单元号。

2. P100 计算平均存取时间

3. **页式虚拟存储管理的基本思想**：当装入一个进程时，把它的全部页面装入虚拟存储器，但执行时先将部分页面装入实际主存，然后根据执行行为动态调用不在主存中的页面，同时进行必要的页面调出。

4. 页面调度算法

1) 最佳置换页面调度算法（OPT）：首先淘汰以后不会再访问的页

2) 先进先出页面调度算法（FIFO）：总是淘汰最先调入主存的那一页

3) 最近最少使用页面调度算法（LRU）：淘汰最近一段时间较久未被使用的那一页

4) 最不常用页面调度算法（LFU）：淘汰最近一段时间内访问次数比较少的页面

5) （不考）时钟页面调度算法（clock）：采用循环队列机制构造页面队列，形成了一个类似于时钟表面的环形表，队列指针相当于钟表上的指针，指向可能要淘汰的页面。

3.4 段式存储管理

段式程序设计可以更好地体现模块化程序设计的思想，应用程序由若干程序段（模块）组成。段式存储管理把进程的逻辑地址空间分成多段，提供（段号:段内位移）形式的二维逻辑地址。

段式存储管理可以实现虚拟存储管理，基本思路是：进程在装入主存时，其所有分段都存放在虚存中，进程运行时先把当前需要的一段或几段装入主存，在执行过程中访问到不在主存的段时再把它动态装入。

第四章 设备管理

4.1 设备管理基础

1. 设备控制器应具有的功能：接收和识别 CPU 或通道发出的命令；实现数据的交换；发现和记录设备及其自身的状态信息，供 CPU 处理时使用；当连接多台设备时，设备控制器还需要识别设备的地址等。

2. 可以将 I/O 控制方式分为四类，轮询方式、终端方式、DMA 方式和通道方式，其中 DMA 方式是高速设备不需要 CPU 接入，直接和主存交换信息。通道方式是速度最快的。

4.2 设备管理软件

1. I/O 软件的总体设计目标：高效率和通用性

2. I/O 软件设计需考虑的问题：设备无关性、出错处理、同步/异步传输、缓冲技术。

3. I/O 缓冲技术的目的：为了解决 CPU 于设备之间速度不匹配的问题，协调逻辑记录与物理记录大小的不一致，提高 CPU 和设备的并行性，减少 I/O 操作对 CPU 的中断次数以及放宽对 CPU 中断响应时间的要求。

4. 常用的缓冲技术：单缓冲、双缓冲和循环缓冲

4.3 独占型外围设备的分配

设备独立性：通常用户不指定具体的物理设备，而指定逻辑设备，使得用户作业和物理设备之间的分配关系分离开来，再通过操作系统的设备管理功能建立逻辑设备和物理设备之间的对应关系设备的这种特性。

设备独立性的优点：用户应用程序与具体物理的外围设备无关；易于处理输入/输出设备的故障；提高了外围设备分配的灵活性。

4.4 共享型外围设备的驱动

1. 磁盘的地址编制方式：（柱面号，磁头号，扇区号），（盘面号，磁道号，扇区号）。

2. 移臂调度最常用的算法为扫描算法，三种扫描算法：

1) 单向扫描算法：移动臂总是向一个方向扫描，顺序处理 I/O 要求，但是再归途中不提供服务。

2) 双向扫描算法：移动臂首先向一个方向移动，将所有访问请求处理完毕后再反向继续处理，如此循环往复。

3) 电梯调度算法：基于双向扫描算法，移动到一个方向最远的请求后反向。

4.5 虚拟设备

1. 虚拟设备是用一类物理设备模拟另一类物理设备的技术，可以将独占型设备变为共享型设备，使用共享型外围存储模拟独占型设备。

2. **SPOOLing 技术**是一种用共享型磁盘设备模拟独占型物理设备的技术，也是一种速度匹配技术，它的核心思想是预输入和缓输出。

第五章 文件管理

5.1 文件系统概述

1. 文件系统是操作系统中负责存取和管理信息的模块，它用统一的方式管理用户和系统信息的存储、检索、更新、共享和保护，并为用户提供一整套方便有效的文件使用和操作方法。

2. 文件按用途可以划分为系统文件、库文件和用户文件。

5.2 文件的组织

1. 常见的文件存储介质：磁带、光盘和磁盘

2. 卷是存储介质的物理单位；

3. 块是存储介质上连续信息所组成的一个区域，也称为物理记录。块在主存储器和辅助存储器进行信息交换时是一个标准的物理单位，即每次都以一个物理块或整数个物理块进行信息交换。

4. 4 种常见的文件物理结构：顺序文件、连接文件、直接文件、索引文件

5. 逻辑文件，又称文件的逻辑结构。它是独立于物理环境的，用户概念中的抽象信息组织方式；用户能观察到的，并加以处理的数据集合。

6. 文件的逻辑结构的两种实现形式：基于字节流的流式文件，基于记录的记录式文件。

5.3 文件目录

1. 文件目录是实现文件的“按名存取”的关键数据结构。

2. 文件目录需要永久保存，因此也组织成文件存放在磁盘上，称目录文件。

5.4 文件的共享、保护和保密

1. 文件的保密：隐藏文件目录、设置口令、使用密码

2. 文件的存取方法：顺序存取、直接存取、索引存取

第六章 并发程序设计

6.1 并发程序设计

1. 程序设计具有的特性：程序执行的顺序性、计算环境的封闭性、计算结果的确定性、计算过程的可再现性。
2. 并发程序设计的特性：并发性、共享性、交互性

6.2 临界区管理

1. 临界区调度的原则：一次性至多只有一个进程进入临界区内执行；如果已有进程在临界区中，试图进入此临界区的其他进程应等待；进入临界区内的进程应在有限时间内退出，以便让等待队列中等待的进程进入。

6.3 PV 操作（P207，看书，看代码）

P(s)：将信号量 s 的 value 值减 1，若结果小于 0，则执行 P 操作的进程被阻塞，进入信号量 s 的 list 所指队列中；若结果大于等于 0，则执行 P 操作的进程继续执行。

V(s)：将信号量 s 的 value 值加 1，若结果不大于 0，则执行 V 操作的进程从信号量 s 的 list 所指队列中释放一个进程，使其转换为就绪态，自己则继续执行；若结果大于 0，则执行 V 操作的进程继续执行。

6.4 管程

管程具有的属性：共享性、安全性、互斥性

6.5 进程通信

1. 进程通信可以分为进程直接通信和进程间接通信
2. 进程直接通信：在该方式下，企图发送或接收消息的每个进程必须指出新建发给谁或从谁那里接受信件。
3. 进程间接通信：采用该方式时，进程间发送或接收信件通过一个共享的数据结构（信箱）来进行，每个信箱有一个唯一的标识符。当两个以上的进程共享一个信箱时，它们之间通过共享的信箱通信。

6.6 死锁

1. **死锁的定义**：如果在一个进程集合中，每个进程都在等待只能由该集合中的另一个进程才能引发的事件，则称一组进程或系统此时发生了死锁。
2. **产生死锁的必要条件**：互斥条件、占有和等待条件、不剥夺条件、循环等待条件。上述条件中，前三个条件是死锁存在的必要条件，但不是充分条件，第四个条件是前三个条件同时存在时产生的结果，所以这些条件并不完全独立。
3. 从四个方面解决死锁问题：死锁的预防、避免死锁、检测死锁、解除死锁。
4. 产生死锁的根本原因：可共享资源不足，另一个基本原因是进程的推进顺序不当。
5. 死锁的避免：当不能防止死锁的产生时，如果能掌握并发进程中与每个进程有关的资源申请情况，仍然可以避免死锁的产生。

6. 银行家算法：这是一种资源分配算法，检查申请者对各类资源的最大需求量，如果系统现存的各类资源可以满足它的最大需求量时，就满足当前的申请。