

第一章 操作系统引论

简答题

【例 2】什么是多道程序技术？在 OS 中引入该技术，带来了哪些好处？

答：多道程序技术是指在内存中同时存放若干个作业，并使它们共享系统的资源且同时运行的技术。

在 OS 中引入多道程序技术带来了以下好处：

(1) 提高 CPU 的利用率。在引入多道程序设计技术后，由于可同时把若干道程序装入内存，并可使它们交替地执行，这样，当正在运行的程序因 I/O 而暂停执行时，系统可调度另一道程序到 CPU 执行，从而可保持 CPU 处于忙状态，使 CPU 的利用率提高。

(2) 可提高内存和 I/O 设备的利用率。为了能运行较大的作业，通常内存都具有较大的容量，但由于 80% 以上的作业都属于中、小型作业，因此在单道程序的环境下也必定造成内存的浪费。类似地，系统中所配置的多种类型的 I/O 设备，在单道程序环境下，也不能充分利用。如果允许在内存中装入多道程序，并允许它们并发执行，则无疑会大大地提高内存和 I/O 设备的利用率。

(3) 增加系统吞吐量。在保持 CPU、I/O 设备不断忙碌的同时，必然会大幅度地提高系统的吞吐量，从而降低作业加工所需的费用。

【例 3】推动批处理系统和分时系统形成和发展的主要动力是什么？

答：(1) 推动批处理系统形成和发展的主要动力是“不断提高系统资源利用率”和“提高系统吞吐量”。它们主要表现在：脱机输入/输出技术的应用和作业的自动过渡大大地提高了 I/O 的速度、I/O 设备与 CPU 并行工作的程度、减少了主机 CPU 的空闲时间；多道程序设计技术的应用更进一步提高了 CPU、内存和 I/O 设备的利用率和系统的吞吐量。

(2) 推动分时系统形成和发展的主要动力则是“为了更好地满足用户的需要”。主要表现在 CPU 的分时使用缩短了作业的平均周转时间；人机交互能力的提供使用户能方便地直接控制自己的作业；主机的共享，使多个用户（包括远程用户）能同时使用同一台计算机，独立地、互不干扰地处理自己的作业。

【例 7】操作系统具有哪几大特征？它们之间有何关系？

答：操作系统的特征有并发性、资源共享性、虚拟性和异步性。它们的关系如下：

(1) 并发性和资源共享性是操作系统最基本的特征。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) 并发性和资源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，自然不存在资源共享问题；另一方面，若系统不能对资源共享实施有效管理，协调好诸进程对共享资源的访问，也必将影响到程序的并发执行，甚至根本无法并发执行。

(3) 虚拟性以并发性和资源共享性为前提。为了使并发进程能更方便、更有效地共享资源，操作系统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，从而解决众多并发进程对有限的系统资源的争用问题。

(4) 异步性是并发性和资源共享性的必然结果。操作系统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其他进程的制约，不再“一气呵成”，这必然导致异步性特征的产生。

填空题

1.4.2 填空题

1. 设计现代 OS 的主要目标是 ① 和 ②。
2. 单道批处理系统是在解决 ① 和 ② 的矛盾中发展起来的。
3. 在单处理机环境下的多道程序设计具有多道、① 和 ② 的特点。
4. 现代操作系统的两个最基本的特征是 ① 和 ②，除此之外，它还具有 ③ 和 ④ 的特征。
5. 从资源管理的角度看，操作系统具有四大功能：①、②、③ 和 ④；而为了方便用户，操作系统还必须提供 ⑤。
6. 除了传统操作系统中的进程管理、存储器管理、设备管理、文件管理等基本功能外，现代操作系统中还增加了①、② 和 ③ 等功能。
7. 操作系统的基本类型主要有 ①、② 和 ③。
8. 批处理系统的主要优点是 ① 和 ②；主要缺点是 ③ 和 ④。
9. 实现分时系统的关键问题是 ①，为此必须引入 ② 的概念，并采用 ③ 调度算法。
10. 分时系统的基本特征是：①、②、③ 和 ④。
11. 若干事件在同一时间间隔内发生称为 ①；若干事件在同一时刻发生称为 ②。
12. 实时系统可分为 ①、②、多媒体系统和嵌入式系统等类型；民航售票系统属于 ③，而导弹飞行控制系统则属于 ④。
13. 为了使实时系统高度可靠和安全，通常不强求 ①。
14. 当前比较流行的微内核的操作系统结构，是建立在层次化结构的基础上的，而且还采用了 ① 模式和 ② 技术。

答案

二、填空题

1. ① 提高资源利用率；② 方便用户。
2. ① 人机矛盾；② CPU 与 I/O 设备速度不匹配。
3. ① 宏观上同时运行；② 微观上交替运行。
4. ① 并发；② 资源共享；③ 虚拟性，异步性。
5. ① 处理机管理；② 存储器管理；③ 设备管理；④ 文件管理；⑤ 友好的用户接口。
6. ① 系统安全；② 网路；③ 多媒体。
7. ① 批处理系统；② 分时系统；③ 实时系统。
8. ① 资源利用率高；② 系统吞吐量大；③ 无交互作用能力；④ 作业平均周转时间长。
9. ① 人机交互；② 时间片；③ 时间片轮转。
10. ① 多路性；② 独立性；③ 交互性；④ 及时性。
11. ① 并发；② 并行。
12. ① 实时信息处理系统；② 实时控制系统；③ 实时信息处理系统；④ 实时控制系统。
13. ① 资源利用率。
14. ① 客户机/服务器；② 面向对象程序设计。

第二章 进程的描述与控制

简答题

【例 6】 如何保证诸进程互斥地访问临界资源？

答：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称作进入区的代码，以检查是否有其他进程已进入临界区使用临界资源，若有，则进程必须等待；否则，允许进程进入临界区，同时设置标志表示有进程正在临界区内。同样，在临界区后必须增加一段称作退出区的代码，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、退出区具体可用多种同步机制实现，如锁、信号量机制等。

【例 7】 何谓“忙等”？它有什么缺点？

答：所谓“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，它仍占有 CPU，并通过不断地执行循环测试指令来等待该事件的完成。

“忙等”的主要缺点是浪费 CPU 的时间，另外，它还可能引起预料不到的后果。例如考虑某个采取高优先权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 的优先权较高，B 的优先权较低，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，陷入“死等”状态。

【例 26】 试从调度性、并发性、拥有资源、独立性、系统开销以及对多处理机的支持等方面，对进程和线程进行比较。

答：进程和线程之间在调度性、并发性、拥有资源、独立性、系统开销及对多处理机的支持方面的比较如下。

(1) 调度性。在传统的操作系统中，拥有资源的基本单位、独立调度和分派的基本单位都是进程。而在引入线程的 OS 中，则把线程作为调度和分派的基本单位，进程只是拥有资源的基本单位，而不再是调度和分派的基本单位。

(2) 并发性。在引入线程的 OS 中，不仅进程间可以并发执行，而且在一个进程内的多个线程间，也可以并发执行，因而比传统的 OS 具有更好的并发性。

(3) 拥有资源。在这两种 OS 中，拥有资源的基本单位都是进程。线程除了一点在运行中必不可少的资源(如线程控制块、程序计数器、一组寄存器值和堆栈)外，本身基本不拥有系统资源，但它可共享其隶属进程的资源。

(4) 独立性。每个进程都能独立地申请资源和独立地运行；但同一进程的多个线程则共享进程的内存地址空间和其他资源，它们之间的独立性比进程之间的独立性要低。

(5) 开销。由于创建或撤消进程时，系统都要为之分配和回收资源，如内存空间等。进程切换时所保存和设置的现场信息也要明显地多于线程，因此，OS 在创建、撤消和切换进程时所付出的开销显著地大于线程。另外，由于隶属于同一个进程的多个线程共享同一地址空间和打开文件，从而使它们之间的同步和通信的实现也变得更为容易。

(6) 支持多处理机系统。传统的进程，只能运行在一个处理机上；多线程的进程，则可以将进程中的多个线程分配到多个处理机上，从而获得更好的并发执行效果。

【例 27】 什么是内核支持线程和用户级线程？并对它们进行比较。

答：内核支持线程是在内核支持下实现的，即每个线程的线程控制块设置在内核中，

所有对线程的操作(如创建、撤消和切换等)，都是通过系统功能调用由内核中的相应处理程序完成。而用户级线程仅存在于用户空间中，即每个线程的控制块设置在用户空间中，所有对线程的操作也在用户空间中完成，而无需内核的帮助。

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，而内核支持线程则不然，它需要得到 OS 内核的支持。

(2) 处理器的分配。在多处理机环境下，对纯粹的用户级线程来说，内核一次只为一个进程分配一个处理器，即进程无法享用多处理机带来的好处；而在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器上并行运行，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，内核的调度方式和算法与进程的调度十分相似，只不过调度的单位是线程；而对只设置了用户级线程的系统，内核调度的单位则仍为进程，当一进程得到 CPU 时，隶属于该进程的多个线程可通过用户态的线程调度分享内核分配给进程的 CPU 时间。因此，在条件相同的情况下，内核支持的线程通常比用户级线程得到更多的 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诸线程之间，由于不需陷入内核，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，被阻塞的将是整个进程；而当一个内核支持线程执行这些系统调用时，内核只阻塞这个线程，但仍可调度其所属进程的其他线程执行。

填空题

2.4.2 填空题

1. 在单用户单任务环境下，用户独占全机，此时机内资源的状态，只能由运行程序的操作加以改变，此时的程序执行具有 ① 性和 ② 性特征。

2. 并发进程之间的相互制约，是由于它们 ① 和 ② 而产生的，因而导致程序在并发执行时具有 ③ 特征。

3. 程序并发执行与顺序执行时相比产生了一些新特征，分别是 ①、② 和 ③。

4. 引入进程的目的是 ①，而引入线程的目的是 ②。

5. 进程由 ①、② 和 ③ 组成，其中 ④ 是进程存在的唯一标志。

6. 进程最基本的特征是 ① 和 ②，除此之外，它还有 ③、和 ④ 特征。

7. 由于进程的实质是程序的一次执行，故进程有 ① 的基本特征，该特征还表现在进程由 ② 而产生，由 ③ 而执行，由 ④ 而消亡，即进程具有一定的生命期。

8. 引入进程带来的好处 ① 和 ②。

9. 当前正在执行的进程由于时间片用完而暂停执行时，该进程应转变为 ① 状态；若因发生某种事件而不能继续执行时，应转为 ② 状态；若应终端用户的请求而暂停执行时，它应转为 ③ 状态。

10. 用户为阻止进程继续运行, 应利用 ① 原语, 若进程正在执行, 应转变为 ② 状态; 以后, 若用户要恢复其运行, 应利用 ③ 原语, 此时进程应转变为 ④ 状态。

11. 系统中共有 5 个用户进程, 且当前 CPU 在用户态下执行, 则最多可有 ① 个用户进程处于就绪状态, 最多可有 ② 个用户进程处于阻塞状态; 若当前在核心态下执行, 则最多可有 ③ 个用户进程处于就绪状态, 最多可有 ④ 个用户进程处于阻塞状态。

12. 为了防止 OS 本身及关键数据(如 PCB 等), 遭受到应用程序有意或无意的破坏, 通常也将处理机的执行状态分成①和 ②两种状态。

13. 进程同步主要是对多个相关进程在 ①上进行协调。

14. 同步机制应遵循的准则有是 ①、②、③和 ④。

15. 在记录型信号量机制中, $S.value > 0$ 时的值表示 ①; 每次 wait 操作意味着 ②, 因此应将 $S.value$ ③, 当 $S.value$ ④时, 进程应阻塞。

16. 在记录型信号量机制中, 每次 signal 操作意味着 ①, 因此应将 $S.value$ ②, 当 $S.value \leq 0$ 时, 表示 ③, 此时应 ④。

17. 在利用信号量实现进程互斥时, 应将 ①置于 ②和 ③之间。

18. 在每个进程中访问 ①的那段代码称为临界区。为实现对它的共享, 应保证进程 ②地进入自己的临界区, 为此在每个进程的临界区前应设置 ③, 临界区后应设置 ④。

19. 利用共享的文件进行进程通信的方式被称作 ①, 除此之外, 进程通信的类型还有 ②、③和 ④三种类型。

20. 客户机—服务器系统通信机制主要的实现方法有是 ①、②和 ③三种。

21. 为实现消息缓冲队列通信, 应在 PCB 中增加 ①、②、③三个数据项。

22. 引入线程概念后, 操作系统以 ①作为资源分配的基本单位, 以 ②作为 CPU 调度和分派的基本单位。

23. 在采用用户级线程的系统中, OS 进行 CPU 调度的对象是 ①; 在采用内核支持的线程的系统中, CPU 调度的对象是 ②。

24. 线程之所以能减少并发执行的开销是因为 ①。

答案

二、填空题

1. ① 封闭性; ② 可再现性。
2. ① 共享资源; ② 相互合作; ③ 间断性或异步性。
3. ① 间断性; ② 失去封闭性; ③ 不可再现性。
4. ① 使程序能正确地并发执行, 以提高资源利用率和系统吞吐量;
② 减少并发执行的开销, 提高程序执行的并发程度。
5. ① 进程控制块(PCB); ② 程序段; ③ 数据段; ④ PCB。
6. ① 动态性; ② 并发性; ③ 独立特征; ④ 异步性。
7. ① 动态性; ② 创建; ③ 调度; ④ 撤销(终止)。
8. ① 提高资源利用率; ② 增加系统吞吐量。
9. ① 就绪; ② 阻塞; ③ 静止就绪。
10. ① 挂起; ② 静止就绪; ③ 激活; ④ 活动就绪。
11. ① 4; ② 4; ③ 5; ④ 5。
12. ① 用户态; ② 系统态。
13. ① 执行次序。
14. ① 空闲让进; ② 忙则等待; ③ 有限等待; ④ 让权等待。
15. ① 可用的临界资源数量; ② 申请一个临界资源; ③ 减 1; ④ 小于 0。
16. ① 释放一个临界资源; ② 加 1; ③ 仍有请求该资源的进程被阻塞;
④ 唤醒相应阻塞队列中的首进程。
17. ① 临界区; ② wait 操作; ③ signal 操作。
18. ① 临界资源; ② 互斥; ③ 进入区; ④ 退出区。
19. ① 管道通信; ② 共享存储器; ③ 消息系统; ④ 客户机-服务器系统。
20. ① 套接字; ② 远程过程调用; ③ 远程方法调用。
21. ① 消息队列首指针 mq; ② 消息队列互斥信号量 mutex;
③ 消息队列资源信号量 Sm。
22. ① 进程; ② 线程。
23. ① 进程; ② 线程。
24. ① 线程基本不拥有资源。

第三章 处理机调度与死锁

简答题

【例 12】 不安全状态是否必然导致系统进入死锁状态？

答：不安全状态不一定导致系统进入死锁状态。因为，安全性检查中使用的向量 Max 是进程执行前提供的，而在实际运行过程中，一进程需要的最大资源量可能小于 Max ，如：一进程对应的程序中有一段进行错误处理的代码，其中需要 n 个 A 种资源，若该进程在运行过程中没有碰到相应错误而不需调用该段错误处理代码，则它实际上将完全不会请求这 n 个 A 种资源。

【例 15】 死锁检测程序的运行频率较高或较低，各有什么优缺点？

答：死锁的检测可非常频繁地在每次资源请求时进行，其优点是：可以尽早地检测到死锁及其所涉及的进程，并有可能找到引起系统死锁的那个(或那几个)进程。其缺点是频繁的检测会耗费相当多的 CPU 时间，增加系统的开销。相反，每隔较长时间或当 CPU 利用率下降到较低程度时进行死锁的检测，则可以降低运行死锁检测程序的开销，但在检测到死锁时可能涉及到很多进程，也难以找到引起死锁的那个进程，从而难以从死锁状态恢复过来。

【例 16】 解除死锁，在选择撤消进程或被抢占资源的进程时，可考虑哪些因素？

答：解除死锁，在选择撤消进程或被抢占资源的进程时，可考虑下列因素：

- (1) 优先权；
- (2) 进程已执行的时间；
- (3) 估计的剩余执行时间；
- (4) 已产生的输出量；
- (5) 已获得的资源量和资源类型；
- (6) 还需要的资源量；
- (7) 进程的类型(批处理型或交互型)；
- (8) 需要被撤消的进程数等。

填空题

1. 高级调度又称作 ① 调度，其主要功能是 ②；低级调度又称作 ③ 调度，其主要功能是 ④。

2. 作业调度必须做 ① 和 ② 两个决定。

3. 进程调度的主要任务是 ①、② 和 ③，进程调度的方式主要有 ④ 和 ⑤ 两种方式。

4. 在抢占调度方式中，抢占的原则主要有：①、② 和 ③。

5. 在设计进程调度程序时，应考虑 ①、② 和 ③ 三个问题。

6. 为了使作业的平均周转时间最短, 应该选择 ① 调度算法; 为了使当前执行的进程总是优先权最高的进程, 则应选择 ② 调度算法; 而分时系统则常采用 ③ 调度算法。

7. 分时系统中, 时间片选得太小会造成 ① 的现象, 因此, 时间片的大小一般选择为 ②。

8. 在采用动态优先权时, 为了避免一个低优先权的进程处于饥饿状态, 可以 ①; 而为了避免一个高优先权的长作业长期垄断 CPU, 则可以 ②。

9. 高响应比优先调度算法综合考虑了作业的 ① 和 ②, 因此会兼顾到长、短作业。

10. 死锁产生的主要原因是 ① 和 ②。

11. 死锁产生的必要条件是 ①、②、③ 和 ④。

12. 通过破坏死锁产生的四个必要条件可进行死锁的预防, 其中 ① 条件一般是不允许破坏的, 一次性分配所有资源破坏的是其中的 ② 条件, 资源的有序分配破坏的是其中的 ③ 条件。

13. 避免死锁, 允许进程动态地申请资源, 但系统在进行分配时应先计算资源分配的 ①。若此次分配不会导致系统进入 ②, 便将资源分配给它, 否则便让进程 ③。

14. 解决死锁问题的方法有预防、避免、检测并解除等, 一次性分配所有的资源采用的是其中的 ① 方法, 银行家算法采用的是其中的 ② 方法。

15. 根据死锁定理, 一个状态为死锁状态的充分条件是当且仅当该状态的资源分配图是 ① 时。

16. ① 和 ② 是解除死锁的两种常用方法。

答案

1. ① 作业调度;
② 按照一定的算法从外存的后备队列中选若干作业进入内存, 并为它们创建进程;
③ 进程调度; ④ 按一定算法从就绪队列中选一个进程投入执行。
2. ① 接纳多少个作业; ② 接纳哪些作业。
3. ① 保存 CPU 现场; ② 按某种算法选择一个就绪进程;
③ 把 CPU 分配给新进程; ④ 抢占调度; ⑤ 非抢占调度。
4. ① 时间片原则; ② 短作业优先; ③ 优先权原则。
5. ① 引起调度的因素; ② 调度算法的选择; ③ 就绪队列的组织。
6. ① 短作业(进程)优先; ② 立即抢占的高优先权优先; ③ 时间片轮转。
7. ① 系统开销增大; ② 略大于一次典型的交互所需要的时间。
8. ① 随着进程等待时间的增加而提高其优先权;
② 随着进程运行时间的增加而降低其优先权。
9. ① 运行时间; ② 等待时间。
10. ① 竞争资源; ② 进程推进顺序非法。
11. ① 互斥条件; ② 请求与保持条件; ③ 不剥夺条件; ④ 环路等待条件。
12. ① 互斥; ② 请求与保持; ③ 环路等待。

- 13. ① 安全性；② 不安全状态；③ 等待。
- 14. ① 预防死锁；② 避免死锁。
- 15. ① 不可完全简化。
- 16. ① 撤消进程；② 剥夺资源。

第四章 存储器管理

简答题

【例 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。问“良好的存储器环境”应包含哪几个方面？

答：存储器管理是为多道程序的并发运行提供良好的存储器环境。它包括以下内容：

- (1) 能让每道程序“各得其所”，并在不受干扰的环境中运行；还可以使用户从存储空间分配、保护等繁琐事务中解脱出来。
- (2) 向用户提供更大的存储空间，使更多的作业能同时投入运行；或使更大的作业能在较小的内存空间中运行。
- (3) 为用户对信息的访问、保护、共享以及动态链接等方面提供方便。
- (4) 良好的存储器环境，还应包括能使存储器有较高的利用率。

【例 2】 在什么情况下需要进行重定位？为什么要引入动态重定位？

答：源程序经过编译、链接产生的装入模块一般总是从 0 开始编址的，其中的地址都是相对于起始地址的相对地址。在将它装入内存时，其分配到的内存空间的起始地址通常不为 0，因此指令和数据的实际物理地址与装入模块中的相对地址是不一致的，此时，为了使程序能够正确执行，必须将相对地址转换成物理地址，即进行重定位。

进程在运行过程中经常要在内存中移动位置(如对换、紧凑时)，引入动态重定位的目的就是为了满足程序的这种需要。动态重定位的实现需要一定的硬件支持，重定位的过程是由硬件地址变换机构在程序执行每条指令时自动完成的。

【例 5】提高内存利用率的途径主要有哪些？

答：内存利用率不高，主要表现为以下四种形式：

(1) 内存中存在着大量的、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，从而也就降低了内存的利用率。

(4) 内存中存在着重复的拷贝。

针对上述问题，可分别采用下述方法提高内存的利用率：

(1) 改连续分配方式为离散分配方式，以减少内存中的零头。

(2) 增加对换机制，将那些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，当程序在运行中需要调用某段程序时，才将该段程序由外存装入内存。这样，可以避免装入一些本次运行中不用的程序。

(4) 引入虚拟存储器机制，使更多的作业能装入内存，并使 CPU 更加忙碌。引入虚拟存储器机制，还可以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拷贝。



填空题

4.4.2 填空题

1. 使每道程序能在内存中“各得其所”是通过 ① 功能实现的；保证每道程序在不受干扰的环境下运行，是通过 ② 功能实现的；为缓和内存紧张的情况而将内存中暂时不能运行的进程调至外存，是 ③ 功能实现的；能让较大的用户程序在较小的内存空间中运行，是通过 ④ 功能实现的。
2. 程序装入的方式有 ① 、 ② 和 ③ 三种方式。
3. 程序的链接方式有 ① 、 ② 和 ③ 三种方式。
4. 把作业装入内存中随即进行地址变换的方式称为 ① ；而在作业执行期间，当访问到指令和数据时才进行地址变换的方式称为 ② 。
5. 地址变换机构的基本任务是将 ① 中的 ② 变换为 ③ 中的 ④ 。
6. 通常，用户程序使用 ① 地址，处理机执行程序时则必须用 ② 地址。
7. 在首次适应算法中，空闲分区以 ① 的次序拉链；在最佳适应算法中，空闲分区以 ② 的次序拉链。
8. 在连续分配方式中可通过 ① 来减少内存零头，它必须得到 ② 技术的支持。
9. 在伙伴系统中，令 $\text{buddy}_k(x)$ 表示大小为 2^k 、起始地址为 x 的块的伙伴的地址，则 $\text{buddy}_k(x)$ 的通用表达式为 ① 。
10. 实现进程对换应具备 ① 、 ② 和 ③ 三方面的功能。
11. 分页系统中若页面较小，虽有利于 ① ，但会引起 ② ；而页面较大，虽可减少 ③ ，但会引起 ④ 。
12. 分页系统中，页表的作用是实现 ① 到 ② 的转换。

13. 在分页系统中为实现地址变换而设置了页表寄存器，其中存放了处于①状态进程的②和③；而其他进程的上述信息则被保存在④中。
14. 引入分段主要是满足用户的需要，具体包括①、②、③、④等方面。
15. 在页表中最基本的数据项是①；而在段表中则是②和③。
16. 把逻辑地址分成页号和页内地址是由①进行的，故分页系统的作业地址空间是②维的；把逻辑地址分成段号和段内地址是由③进行的，故分段系统的作业地址空间是④维的。
17. 在段页式系统中(无快表)，为获得一条指令或数据，都需三次访问内存。第一次从内存中取得①；第二次从内存中取得②；第三次从内存中取得③。

答案

1. ① 内存分配；② 内存保护；③ 对换；④ 内存扩充(或虚拟存储器)。
2. ① 绝对装入方式；② 可重定位装入方式；③ 动态运行时装入方式。
3. ① 静态链接；② 装入时动态链接；③ 运行时动态链接。
4. ① 静态重定位；② 动态重定位。
5. ① 地址空间；② 逻辑地址；③ 内存空间；④ 物理地址。
6. ① 逻辑；② 物理。
7. ① 地址递增；② 空闲区大小递增。
8. ① 紧凑；② 动态重定位。
9. ① $x + 2^k - [(x / 2^k) \% 2] * 2^{k+1}$ (其中“%2”表示除以2然后取余数)。
10. ① 对换空间的管理；② 进程换入；③ 进程换出。
11. ① 减少块内碎片；② 页表太长；③ 页表长度；④ 块内碎片增大。
12. ① 页号；② 物理块号。
13. ① 执行；② 页表长度；③ 页表始址；④ 它们的PCB。
14. ① 便于编程；② 分段共享；③ 分段保护；④ 动态链接。
15. ① 物理块号；② 段的内存基址；③ 段长。
16. ① 机器硬件；② 一；③ 程序员；④ 二。
17. ① 页表起始地址；② 块号；③ 指令或数据。

第五章 虚拟存储器

简答题

【例3】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？

答：因请求调页时，只要作业的部分页在内存，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时，则产生缺页中断，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，而相应的页都不在内存时，就将产生多次缺页中断。

填空题

1. 在请求调页系统中，地址变换过程可能会因为 ①、② 和 ③ 等原因而产生中断。
2. 虚拟存储器的基本特征是 ① 和 ②，因而决定了实现虚拟存储器的关键技术是 ③ 和 ④。
3. 实现虚拟存储器，除了需要有一定容量的内存和相当容量的外存外，还需要有 ①、② 和 ③ 的硬件支持。
4. 为实现请求分页管理，应在纯分页的页表基础上增加 ①、②、③ 和 ④ 等数据项。
5. 在请求调页系统中要采用多种置换算法，其中 OPT 是 ① 置换算法，LRU 是 ② 置换算法，NUR 是 ③ 置换算法，而 LFU 则是 ④ 置换算法，PBA 是 ⑤ 算法。
6. VAX/VMS 操作系统采用页面缓冲算法：它采用 ① 算法选择淘汰页，如果淘汰页未被修改，则将它所在的物理块插到 ② 链表中，否则便将其插入 ③ 链表中，它的主要优点是大大减少 ④ 次数。
7. 在请求调页系统中，调页的策略有 ① 和 ② 两种方式。
8. 在请求调页系统中，反复进行页面换进和换出的现象称为 ①，它产生的原因主要是 ②。
9. 分页系统的内存保护通常有 ① 和 ② 两种措施。
10. 分段系统中的越界检查是通过 ① 中存放的 ② 和逻辑地址中的 ③ 的比较，以及段表项中的 ④ 和逻辑地址中的 ⑤ 的比较来实现的。
11. 为实现段的共享，系统中应设置一张 ①，每个被共享的段占其中的一个表项，其中应包含了被共享段的段名、②、③ 和 ④ 等数据项；另外，还在该表项中记录了共享该段的 ⑤ 的情况。
12. 在分段系统中常用的存储保护措施有 ①、②、③ 三种方式。

13. 在采用环保护机制时，一个程序可以访问驻留在 ① 环中的数据；可以调用驻留在 ② 环中的服务。

14. Intel x86/Pentium 系列 CPU 可采用 ① 和 ② 两种工作模式。

15. Intel x86/Pentium 的分段机制，每个进程用于地址映射的段表也叫做 ①；另外，当进程运行在特权级别为 0 的核心态下时，它必须使用 ② 来进行地址映射。

16. Intel x86/Pentium 的分页机制，采用 ① 级分页模式，其外层页表也叫做 ②。

答案

二、填空题

1. ① 逻辑地址越界；② 缺页；③ 访问权限错误。

2. ① 多次性；② 对换性；③ 请求调页(段)；④ 页(段)置换。

3. ① 页表机制；② 地址变换机构；③ 缺页中断机构。

4. ① 状态位；② 访问字段；③ 修改位；④ 外存地址。

5. ① 最佳；② 最近最久未用；③ 最近未用；④ 最少使用；⑤ 页面缓冲。

6. ① FIFO；② 空闲页面；③ 修改页面；④ 换进/换出而读写磁盘。

7. ① 预调页；② 请求调页。

8. ① 抖动；② 置换算法选用不当。

9. ① 越界检查；② 存取控制。

10. ① 段表寄存器；② 段表长度；③ 段号；④ 段长；⑤ 段内地址。

11. ① 共享段表；② 共享进程计数；③ 段在内存的起始地址；
④ 段长；⑤ 每个进程。

12. ① 越界检查；② 存取控制权限检查；③ 环保护机构。

13. ① 相同环或较低特权；② 相同环或较高特权。

14. ① 实模式；② 保护模式。

15. ① 局部描述符表 LDT；② 全局描述符表 GDT。

16. ① 两；② 页目录。

第六章 输入输出系统

简答题

【例 2】请说明什么是中断，并给出 CPU 响应中断并进行中断处理的一般过程。

答：中断是指计算机在执行期间，系统内部或外部设备发生了某一急需处理的事件，使得 CPU 暂时停止当前正在执行的程序而转去执行相应的事件处理程序，待处理完毕后又返回原来被中断处，继续执行被中断的程序的程序的过程。

CPU 响应中断并进行中断处理的一般过程如下：

(1) 保存现场。CPU 收到中断信号后，通常由硬件自动将处理机状态字 PSW 和程序计数器 PC 中的内容，保存到中断保留区(栈)中。

(2) 转中断处理程序进行中断处理。由硬件分析中断的原因，并从相应的中断向量中获得中断处理程序的入口地址，装入 CPU 的程序计数器中，从而使处理机转向相应的中断处理程序。

(3) 中断返回。中断处理完成后通过中断返回指令，将保存在中断栈中的被中断进程的现场信息取出，并装入到相应的寄存器中，从而使处理机返回到被中断程序的断点执行。

填空题

1. 对打印机的 I/O 控制方式常采用 ①，对磁盘的 I/O 控制方式常采用 ②。
2. DMA 是指允许 ① 和 ② 之间直接交换数据的设备。在 DMA 中必须设置地址寄存器，用于存放 ③；还必须设置 ④ 寄存器用来暂存交换的数据。
3. 设备控制器是 ① 和 ② 之间的接口，它接受来自 ③ 的 I/O 命令，并用于控制 ④ 的工作。
4. 缓冲池中的每个缓冲区由 ① 和 ② 两部分组成。
5. I/O 软件通常被组织成 ①、②、③ 和 ④ 四个层次。
6. 驱动程序与 ① 紧密相关，如果计算机中连有 3 个同种类型的彩色终端和 2 个同种类型的黑白终端，可以为它们配置 ② 个设备驱动程序。
7. 为实现设备分配，系统中应配置 ① 和 ② 的数据结构；为实现控制器和通道的分配，系统中还应配置 ③ 和 ④ 的数据结构。
8. 除了设备的独立性外，在设备分配时还要考虑 ①、② 和 ③ 三种因素。
9. 为实现设备独立性，在系统中必须设置 ① 表，通常它包括 ②、③ 和 ④ 三项。
10. SPOOLing 系统是由磁盘中的 ① 和 ②，内存中的 ③ 和 ④，⑤ 和 ⑥ 以及井管理程序构成的。
11. 实现后台打印时，SPOOLing 系统中的输出进程，只为请求 I/O 的进程做两件事：
(1) 为之在输出井中申请一 ①，并将 ② 送入其中；(2) 为用户进程申请一张 ③，并将 ④ 填入表中，再将该表排在 ⑤ 队列中。

12. 磁盘的访问时间由①、②和③三部分组成,其中所占比重比较大的是④,故磁盘调度的目标为⑤。

13. 在磁盘调度中,选择优先为离当前磁头最近的磁道上的请求服务的算法为①算法,这种算法的缺点是会产生②现象;选择优先为当前磁头移动方向上、离当前磁头最近的磁道上的请求服务的算法为③算法。

答案

二、填空题

1. ① 中断驱动方式; ② DMA 控制方式。
2. ① I/O 设备; ② 内存; ③ 内存地址; ④ 数据缓冲。
3. ① CPU; ② I/O 设备; ③ CPU; ④ I/O 设备。
4. ① 缓冲首部; ② 缓冲体。
5. ① 用户层软件; ② 设备独立性软件; ③ 设备驱动程序; ④ I/O 中断处理程序。
6. ① I/O 设备的特性; ② 2。
7. ① 设备控制表; ② 系统设备表; ③ 控制器控制表; ④ 通道控制表。
8. ① 设备的固有属性; ② 设备的分配算法; ③ 设备分配中的安全性。
9. ① 逻辑设备表 LUT; ② 逻辑设备名; ③ 物理设备名; ④ 设备驱动程序入口地址。
10. ① 输入井; ② 输出井; ③ 输入缓冲区; ④ 输出缓冲区; ⑤ 输入进程;
⑥ 输出进程。
11. ① 空闲缓冲区; ② 要打印的数据; ③ 用户打印请求表; ④ 用户的打印要求;
⑤ 假脱机文件。
12. ① 寻道时间; ② 旋转延迟时间; ③ 数据传输时间; ④ 寻道时间;
⑤ 使磁盘的平均寻道时间最短。
13. ① 最短寻道时间优先(SSTF); ② 饥饿; ③ 电梯调度(SCAN)。

第七章 文件管理

简答题

【例 3】 目前广泛采用的目录结构是哪种？它有什么优点？

答：目前广泛采用的目录结构是多级树形目录结构。它具有以下优点：

(1) 能有效地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；但对有 i 级的树形目录，在目录中每检索一指定的文件，最多可能要检索到近 $i \times \sqrt{N}$ 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，使用与其他用户文件相同的名字；而且，同一个用户的不同分目录中的文件也允许重名。

(3) 便于实现文件共享。在树形目录中，用户可通过路径名来共享他人的文件；也可将一共享文件链接到自己的目录下，从而使文件的共享变得更为方便，其实现方式也非常简单，系统只需在用户的目录文件中增设一目录项，填上用户赋予该共享文件的新文件名，以及该共享文件的唯一标识符(或索引结点编号)即可。

(4) 能更有效地进行文件的管理和保护。在多级目录中，用户可按文件的不同性质，将它们存放到不同的目录子树中，还可以给不同的目录赋予不同的存取权限，因此，能更有效地对文件进行管理和保护。

填空题

1. 文件管理应具有 ①、②、③ 和 ④ 等功能。
2. 文件按逻辑结构可分成 ① 和 ② 两种类型，现代操作系统普遍采用的是其中的 ③ 结构。
3. 记录式文件，把数据的组织分成 ①、② 和 ③ 三级。
4. 数据项是用来描述一个实体的 ①；记录是用来描述一个实体的 ②；文件用于描述 ③ 的某方面的属性。
5. 一个文件系统模型由最低层 ①、中间层 ② 和最高层 ③ 三个层次组成。
6. 对文件的访问有 ① 和 ② 两种方式。
7. 从文件管理的角度来看，文件是由 ① 和文件体两部分组成的；而在具体实现时，前者的信息通常以 ② 或 ③ 的方式存放在文件存储器上。
8. 文件目录的最主要功能是实现 ①，故目录项的内容至少应包含 ② 和 ③。
9. 对目录管理的要求，首先是能实现 ①，其次是提高对目录的 ②，同时应允许多个用户 ③，以及允许 ④，以便不同用户能按自己的习惯对文件命名。

10. 在采用树形目录结构的文件系统中, 树的结点分为三类: 根结点表示根目录, 枝结点表示 ①, 叶结点表示 ②。

11. 在利用线性检索法对树形目录进行检索时, 系统首先读入 ①, 将它与 ② 文件中的各目录项中的文件名进行比较。若匹配, 便可得到 ③。

12. ① 是指避免文件拥有者或其他用户因有意或无意的错误操作使文件受到破坏; ② 是指允许多个用户共同使用同一个文件。

13. 引入索引结点后, 一个文件在磁盘中占有的资源包括 ①、② 和 ③ 三部分。

14. 文件在使用前必须先执行 ① 操作, 其主要功能是把文件的 ② 从外存复制到内存中, 并在 ③ 和 ④ 之间建立一条通路, 再返回给用户一个 ⑤。

答案

二、填空题

1. ① 文件存储空间的管理; ② 目录管理; ③ 文件的读/写管理;
④ 文件的共享和保护。
2. ① 字符流式; ② 记录式; ③ 字符流式。
3. ① 数据项; ② 记录; ③ 文件。
4. ① 某个属性; ② 某方面(相对完整)的属性; ③ 一个实体集(即群体)。
5. ① 对象及其属性; ② 对对象进行操纵和管理的软件集合; ③ 用户接口。
6. ① 顺序访问; ② 随即访问。
7. ① 文件控制块 FCB; ② 目录项; ③ 磁盘索引结点。
8. ① 按名存取(文件名到物理地址的转换); ② 文件名; ③ 文件的物理地址。
9. ① 按名存取; ② 检索速度; ③ 共享文件; ④ 文件重名。
10. ① 子目录文件; ② 数据文件。
11. ① 路径名的第一个分量名; ② 根目录/当前工作目录; ③ FCB/索引结点指针。
12. ① 文件保护; ② 文件共享。
13. ① 一个磁盘索引结点; ② 一个(或多个)目录项;
③ 若干个存放文件内容的磁盘块。
14. ① 打开; ② FCB/索引结点; ③ 用户; ④ 指定文件; ⑤ 文件描述符。

第八章 磁盘管理

填空题

1. 文件的物理结构主要有 ①、② 和 ③ 三种类型，其中顺序访问效率最高的是 ④，随机访问效率最高的是 ⑤。
2. 可将顺序文件中的内容装入到 ① 的多个盘块中，此时，文件 FCB 的地址部分给出的是文件的 ②，为了访问到文件的所有内容，FCB 中还必须有 ③ 信息。
3. 可将链接式文件中的文件内容装入到 ① 的多个盘块中，并通过 ② 将它们构成一个队列，③ 链接文件具有较高的检索速度。
4. 对字符流式文件，可将索引文件中的文件内容装入到 ① 的多个盘块中，并为每个文件建立一张 ② 表，其中每个表项中含有 ③ 和 ④。
5. UNIX System V 将分配给文件的前十个数据盘块的地址登记在 ① 中，而所有后续数据块的地址则登记在 ② 盘块中；再将这些登记数据块地址的首个盘块的块号登记在 ③ 中，其他块的块号则登记在 ④ 盘块中。
6. 在利用空闲链表来管理外存空间时，可有两种方式：一种以 ① 为单位拉成一条链；另一种以 ② 为单位拉成一条链。
7. 在成组链接法中，将每一组的 ① 和该组的 ② 记入前一组的 ③ 盘块中；再将第一组的上述信息记入 ④ 中，从而将各组盘块链接起来。

答案

二、填空题

1. ① 顺序结构；② 链接结构；③ 索引结构；④ 顺序结构；⑤ 索引结构。
2. ① 连续；② 首个物理块的块号；③ 文件长度。
3. ① 离散；② 链接指针；③ 显式。
4. ① 离散；② 索引；③ 逻辑块号；④ 对应的物理块块号。
5. ① 索引结点的直接地址项；② 一次间址；③ 索引结点的一次间址项；④ 二次间址。
6. ① 空闲盘块；② 空闲盘区。
7. ① 盘块数；② 所有的盘块号；③ 最后一个；④ 超级块的空闲盘块号栈。