

操作系统背诵笔记

简答题

1、批处理操作系统、分时操作系统和实时操作系统各有什么特点？

答：**批处理操作系统**：用户脱机使用计算机，作业是成批处理的，系统内多道程序并发执行，交互能力差。

分时操作系统：可以让多个用户同时使用计算机，人机交互性较强，具有每个用户独立使用计算机的独占性，系统响应及时。

实时操作系统：能对控制对象作出及时反映，可靠性高，响应及时，但是资源利用率低。

2、操作系统具有哪些基本特征？

答：（1）**并发性**。并发性是指两个或两个以上的活动在同一时间间隔内发生。

（2）**共享性**。共享性是指操作系统中的资源可被多个并发执行的进程所使用。

（3）**虚拟性**。虚拟性是指把一个物理上的实体变为若干个逻辑上的对应物。前者是实的，即实际存在的；而后者是虚的，是用户感觉上的东西。

（4）**异步性**。在多道程序环境下，系统允许多个进程并发执行，由于资源有限而进程众多，多数情况下，进程都是按“走走停停”的方式执行，即进程是以人们不可预知的速度向前推进的，这就是异步性。

3、进程是进程实体的运行过程，是系统进行资源分配和调度的一个独立单位。

进程的特征：

（1）**动态性**。进程是程序的一次执行。（最基本的特征）

（2）**并发性**。指多个进程实体，同存于内存中，且能在同一段时间内同时运行。

（3）**独立性**。指进程实体是一个能独立运行、独立获得资源和独立接受调度的基本单位。

（4）**异步性**。指进程按异步方式运行的，即按各自独立的、不可预知的速度向前推进。

（5）**结构性**。指每个进程都配置了一个 PCB，从结构上看，进程实体由程序段、数据段和进程控制块组成。

进程控制块主要包含信息：

（1）进程标识符

①外部标识符 ②内部标识符

（2）处理机状态

①通用寄存器 ②指令计数器 ③程序状态字 PSW ④用户栈指针

（3）进程调度信息

①进程状态 ②进程优先级 ③进程调度所需的其它信息 ④事件

（4）进程控制信息

①程序和数据的地址 ②进程同步和通信机制 ③资源清单 ④链接指针

进程的创建：

引起事件：（1）用户登录（2）作业调度（3）提供服务（4）应用请求

创建过程：（1）申请空白 PCB（2）分配运行所需资源（3）初始化 PCB（标识信息、处理机状态信息、处理机控制信息）（4）进入就绪队列

进程的终止：

引起事件：（1）正常结束（2）异常结束（3）外界干预

终止过程：（1）取出 PCB 读出进程状态（2）若进程处于执行状态，终止执行，置调度标志为真。（3）若有子孙进程，将子孙进程全部终止。（4）归还全部资源（5）将 PCB 从所在队列移除。

进程的阻塞与唤醒：

引起事件：（1）向系统请求共享资源失败（2）等待某种操作完成（3）新数据尚未到达（4）等待新任务到达

阻塞过程：（1）停止执行（2）更改 PCB 状态为阻塞（3）将 PCB 插入阻塞队列（4）重新调度

唤醒过程：（1）将 PCB 移出阻塞队列（2）更改 PCB 状态为就绪（3）将 PCB 插入就绪队列

进程的挂起与激活：

挂起过程：（1）检查进程状态，若处于活动就绪状态，则改为静止就绪；若处于活动阻塞状态，则改为静止阻塞；若处于执行状态，则重新调度（2）为了方便考查进程运行情况，将 PCB 复制到某指定内存区域

激活过程：（1）检查进程状态，若处于静止就绪，则改为活动就绪；若处于静止阻塞，则改为活动阻塞（2）若为抢占调度策略，检查是否要进行重新调度

4、请描述在当前运行进程状态改变时，操作系统进行进程切换的步骤。（重点：有关切换、中断等问题都可以套用）

- 答：（1）保存当前进程上下文环境。
（2）对当前运行进程的 PCB 进行更新，并将其移入适当的队列。
（3）选择其它进程执行。
（4）对选择进程 PCB 进行更新，包括将其状态改为运行。
（5）对存储器管理数据结构进行更新。
（6）恢复被选择进程上次移出时的处理器状态。

5、**临界资源**：同时仅允许一个进程使用的资源。许多物理设备都属于临界资源，如打印机等。

临界区：每个进程中访问临界资源的那段代码。（进入区、临界区、退出区、剩余区）

6、请解释进程同步机制中的让权等待的概念，并说明为什么要采用让权等待。

答：进程同步机制中的让权等待的概念是：当进程不能获得资源时，放弃处理机，避免忙等。采用让权等待，主要是为了更有效地发挥 CPU 的效能，提高系统的吞吐量。

7、试写出 P（S）操作的主要操作步骤。

- 答：（1） $S = S - 1$ （S 为信号量）。
（2）若 $S < 0$ ，阻塞当前进程，将其插入 S 的等待队列，调度另一进程运行。
（3）若 $S \geq 0$ ，当前进程继续运行。

8、试写出 V（S）操作的主要操作步骤。

- 答：（1） $S = S + 1$ （S 为信号量）。
（2）若 $S \leq 0$ ，唤醒第一个等待该资源的进程，当前进程继续运行。
（3）若 $S > 0$ ，当前进程继续运行。

9、阐述对于互斥临界区的管理要求。

答：为实现进程互斥，可利用软件方法，也可在系统中设置专门的同步机制来协调各进程，但所有的同步机制都应遵循下述 4 条规则：

- （1）**空闲让进**。无进程处于临界区时，相应的临界资源处于空闲状态，因而可允许下个请求进入临界区的进程立即进入自己的临界区，以有效地利用临界资源。
- （2）**忙则等待**。已有进程进入自己的临界区时，相应临界资源正被访问，所有其它试图进入临界区的进程必须等待，以保证各进程互斥地访问临界资源。

- (3) **有限等待**。对要求访问临界资源的进程，应保证该进程能在有效时间内进入自己的临界区，以免陷入“死等”状态。
- (4) **让权等待**。当进程不能进入自己的临界区时，应立即释放处理机，以免进程陷入“忙等”。

9、进程之间有哪些基本的通信方式？它们分别有什么特点？适用于哪些场合？

答：进程通信根据交换信息量的多少分为高级通信和低级通信。

低级通信一般只传送一个或几个字节的信息，以达到控制进程执行速度的作用（如PV操作）；高级通信则要传送大量数据，目的是为了交换信息。

高级进程通信方式有很多种，大致可归并为三类：共享存储器、管道通信和消息传递。

共享存储器：在内存中分配一片空间作为共享存储区，需要进行通信的进程把它附加到自己的地址空间中，不需要时则把它取消。

管道通信：它建立连接两个命令的一个打开文件，一个命令向该文件中写入数据，为写者；另一个命令从该文件中读出数据，为读者。

消息传递：它以消息为单位在进程间进行数据交换。

10、列举引起进程调度的时机。

答：（1）执行中的进程执行完毕或因发生某事而不能再继续执行。

（2）执行中的进程因提出I/O请求而暂停执行。

（3）在进程通信或同步过程中执行了某种原语操作，如P操作、block原语、wakeup原语等。

（4）在可剥夺式调度中，有一个比当前进程优先权更高的进程进入就绪队列。

（5）在时间片轮转算法中，时间片用完。

11、进程与线程的主要区别是什么？

答：进程和线程的主要区别如下：

（1）**调度方面**。传统操作系统中，拥有资源和独立调度的基本单位是进程，而在引入线程的操作系统中，线程是独立调度的基本单位，进程是资源拥有的基本单位。在同一进程中，线程的切换不会引起进程切换。在不同的进程中进行线程切换，将会引起进程切换。

（2）**并发性**。不仅进程之间可以并发执行，而且同一进程内的多个线程之间也可以并发执行。

（3）**拥有资源**。进程是拥有资源的基本单位，线程不拥有资源，但线程可以访问其隶属进程的系统资源。

（4）**独立性**。同一进程中的不同线程之间的独立性要比不同进程之间的独立性低得多。

（5）**系统开销**。进程切换时，涉及整个当前进程CPU环境的保存以及新调度进程的CPU环境的设置；而线程切换时，只需保存和设置少量寄存器内存，因此开销小。另外，由于同一进程的多个线程共享进程的地址空间，因此多线程之间的同步与通信非常容易实现。

12、产生死锁的原因：竞争资源和进程的推进顺序不当。

什么是死锁？如何预防死锁？

答：死锁是因进程竞争资源或推进顺序不当而引发的一种胶着状态，即如果一组进程中的每一个进程都在等待仅由该组进程中的其它进程才能引发的事件，那么该组进程就是死锁的，若无外力作用，这种状态将永久保持下去。

死锁的四个必要条件：互斥条件、请求和保持条件、不可抢占条件、循环等待条件。

为了预防死锁，必须破坏死锁的四个必要条件。由于互斥条件一般不能改变，因此可以采取破坏四个必要条件的后三个。

13、简述解决死锁问题的三种主要方法。（一定要注意甄别）

答：解决死锁的三种主要方法有：

- （1）**死锁的预防**。系统按预定的策略为进程分配资源，这些分配策略能使死锁的四个必要条件之一不成立，从而使系统不产生死锁。
- （2）**死锁的避免**。系统动态地测试资源分配情况，仅当能确保系统安全时才给进程分配资源。
- （3）**死锁的检测**。对资源的申请和分配不加限制，只要有剩余的资源就应把资源分配给申请者，操作系统要定时判断系统是否出现了死锁，当有死锁发生时设法解除死锁。

14、在哲学家就餐问题中，如果将先拿起左边筷子的哲学家称为左撇子，而将先拿起右边筷子的哲学家称为右撇子。在同时存在左撇子和右撇子的前提下，我们安排哲学家随意就座。请问是否可能产生死锁，为什么？（重点：反证思维一定要有）

答：不可能产生死锁，它破坏了产生死锁的必要条件——“循环等待”。

如果存在所有左边的哲学家等待右边哲学家放下筷子的循环等待，则每个哲学家肯定已获得左边的筷子，但还没有获得右边的筷子，这与存在右撇子的情况不符。同理，亦不存在相反的循环等待链，而且，因不相邻的哲学家之间不存在竞争资源关系，所以也不可能存在五个以下哲学家的循环等待链。

15、在存储器管理中，什么是重定位？为什么要引入重定位技术？

答：重定位就是将作业地址空间中的逻辑地址转换为主存中的物理地址，其实质为地址变换。因为源程序经过编译、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（逻辑地址）。而在装入内存时，其分配到的内存的起始地址通常不是 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不同。为使程序能够正确运行，必须进行重定位。

16、为什么要进行内存管理？

答：在单道批处理系统阶段，一个系统一个时间段内只执行一个程序，内存的分配极其简单，仅分配给当前运行进程即可。而引入了多道程序的并发执行之后，进程之间共享的不仅仅是处理机，还有主存储器。如果不对内存进行管理，容易导致内存数据的混乱，以至于限制进程的并发执行，所以为了更好地支持多道程序并发执行，必须要进行内存管理。

17、程序的装入和链接：静态装入和可重定位装入、静态链接、动态链接、运行时动态链接。（容易搞混淆，建议考试前多看几遍）

编译：由编译程序将用户源代码编译成若干个目标模块。

链接：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，形成一个完整的装入模块。

装入：由装入程序将装入模块装入内存。

绝对装入方式：用户程序经编译后，将产生绝对地址的目标代码。装入模块被装入内存后，由于程序中的相对地址与实际内存地址完全相同，故不必对程序和数据的地址进行修改。

可重定位装入方式：用户程序编译形成的若干个目标模块，它们的起始地址从 0 开始，程序中的其它地址也都是相对于起始地址计算的。故要根据内存的具体情况将装入模块装入到内存的适当位置，并在进程装入时一次完成对目标程序中指令和数据地址的修改，即地址变换。

动态运行时的装入方式：装入程序把装入模块装入内存后，并不立即把装入模块中的逻辑地址转换为物理地址，而是等到程序真正要执行时才进行，此方式通常需要重定位寄存器支持。

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。

装入时动态链接：将用户源程序编译后所得的一组目标模块，在装入内存时，采用边装入边链接的链接方式。

运行时动态链接：对某些目标模块的链接，是在程序执行中需要该目标模块时，才对它进行链接。其优点是便于修改和更新，便于实现对目标模块的共享。

18、分区存储管理：分区方式（单一连续分区、固定分区、可变式分区）、分区分配算法（首次适应算法、循环首次适应算法、最佳适应法、最坏适应法等）。

单一连续分配：分为系统区、用户区。

优点：简单、无外部碎片，可采用覆盖技术，不需要额外的技术支持。

缺点：只能用于单用户、单任务的操作系统中，有内部碎片，存储器利用率极低。

固定分区：将用户内存空间划分为若干个固定大小区域，每个分区只装入一道作业。

优点：可用于多道程序系统最简单的存储分配，无外部碎片。

缺点：不能实现多进程共享一个主存区，利用率较低，有内部碎片。

可变式分区（动态分区）：在进程装入内存时，根据进程的大小动态地建立分区，并使分区的大小正好适合进程的需要。

首次适应算法：空闲分区以地址递增的次序链接。分配内存时顺序查找，找到大小能满足要求的第一个空闲分区。

优点：优先利用内存低地址部分的空间，从而保留了高地址部分的大的空闲分区，无内部碎片。

缺点：由于低地址部分不断被划分，致使低地址留下许多难以利用的很小的空间，即外部碎片，而每次查找又都是从低地址部分开始的，这无疑增加了查找可用空间分区的开销。

循环首次适应算法（邻近适应、下次适应）：空闲分区以地址递增次序链接。分配时从上次查找结束的位置开始继续循环查找，找到大小能满足要求的第一个空闲分区。

优点：使空闲分区的分布更加均匀，减少了查找空闲分区的开销。

缺点：导致缺乏大的空闲分区。

最佳适应算法：空闲分区以容量递增的次序链接，找到第一个能满足要求的空闲分区。

优点：总能分配给作业最恰当的分区，并保留大的分区。

缺点：导致产生很多难以利用的碎片空间，即外部碎片。

最差适应算法（最大适应）：空闲分区以容量递减的次序链接，找到第一个能满足要求的空闲分区，也就是挑选出最大的分区。

优点：使分给作业后剩下的空闲分区比较大，足以装入其它作业。

缺点：由于最大的空闲分区总是因首先分配而被划分，当有大作业到来时，其存储空间的申请会得不到满足。

19、在内存管理中，内部碎片和外部碎片各指的是什么？在固定式分区管理、可变式分区分配、页式虚拟存储系统、段式虚拟存储管理中，各会存在何种碎片？为什么？

答：**内部碎片：**指已经被分配出去但不能被利用的内存空间。

外部碎片：指还没有被分配出去，但因其太小而无法分配给申请内存空间的新进程的内存空闲区域。

固定分区管理：存在内部碎片。因为固定分区管理的每一个分区大小固定且分配给指定的进程，所以整个分区都属于一个进程，但如果进程比分区小，就会留下无法利用的内存空间。

可变式分区分配：存在外部碎片。因为每个分区大小不固定，所以不可能分配给一个进程多余的空闲区域，但是由于进程需要一段连续的空闲区域，而内存中小的空闲区域可能满足不了进程的需求，就成了无法分配出去的外部碎片。

页式虚拟存储系统：存在内部碎片。因为页的大小是固定的，每个进程的大小不会都刚好是页大小的整数倍，所以进程的最后一页往往都会存在一些已经分配给该进程但无法利用的空间。

段式虚拟存储管理：存在外部碎片。因为一个段的分配需要连续的区域，而段长又不固定，所以内存中会存在一些小的空闲区域不能分配给任何一段，就成了无法分配出去的外部碎片。

20、解释页式存储管理中为什么要设置页表和快表。

答：页式存储管理首先把主存储器分成大小相等的物理块，作为主存分配的物理单位，同时要求程序逻辑地址也分成与块大小一致的页面，这样就可以把作业信息按页面存放在块中。进行存储分配时，根据作业大小，确定其页面数，在装入主存时给它分配相应数目的物理块。这些物理块可以不相邻，为了在作业执行过程中准确地查找逻辑地址与绝对地址的对应关系，系统为每个作业建立一张页表，指出逻辑地址中的页号与物理块中的块号的对应关系。

页表一般存放在主存储器中，当要按给定的逻辑地址进行读/写时，必须两次访问主存，延长了指令的执行周期，降低了执行速度，为了提高存取速度，系统设置了一个小容量的高速缓冲存储器，利用高速缓冲存储器存放页表的一部分，这部分页表即“快表”，利用快表可以一次访问主存完成读/写，大大缩短地址转换时间，从而提高查找速度和执行指令速度。

21、阐述基本分页存储管理和请求分页存储管理的异同之处。（非常重要！）

答：在基本分页存储管理系统中，系统将每个程序按固定的大小分成若干页，每页对应一个物理块号，程序的所有页面都被装入到内存当中。

在请求分页存储管理系统中，程序仍然被系统分成若干页，但并不是所有的页面都被装入到系统中，而仅仅装入程序运行所必须的页面。当需要某一个页面时，再请求从外部调入，如果没有空闲的空间，则利用置换技术进行页面的淘汰和置换。

22、在分页存储管理系统中，页表的主要作用是什么？现代大多数计算机系统都支持非常大的逻辑地址空间，这给页表设计带来了什么样的新问题，应如何解决。

答：页表的主要作用是记录进程的每个页面与对应的页框信息。非常大的逻辑地址空间会导致进程的页表非常大，难以装入连续的地址空间。为解决这个问题，可引入多级页表机制，即对页表进行分页，实现页表的离散存储，对离散分配的页表再建立页表。在此基础上，还可以引入虚拟存储技术，即将部分页表调入内存从而提高内存利用率。

22、局部性理论。（重点！）

局部性原理：大多数程序执行时，在一个较短的时间内仅使用程序代码的一部分，相应地，程序所访问的存储空间也局限于某个区域。

时间局部性：一条指令的一次执行和下次执行，一个数据的一次访问和下次访问，都集中在一个较短的时间内。

空间局部性：当前指令和邻近的几条指令，当前访问的数据和邻近的数据，都集中在一个较小的区域内。

页面分配策略：固定分配局部置换、可变分配全局置换、可变分配局部置换。

抖动（颠簸）：刚被淘汰的页面，过后不久又要访问，并且调入不久以后又调出，如此反复，使得系统把大部分时间用在了页面的调入调出上，而几乎不能完成任何有效的工作。

Belady 异常：FIFO 置换算法的缺页率可能会随着所分配的物理块数增加而增加的异常现象。

23、操作系统中什么是虚拟存储器？为什么要引入虚拟存储技术？

答：在具有层次结构存储器的计算机系统中，自动实现部分装入和部分替换功能，能从逻辑上为用户提供一个比物理存储容量大得多，可寻址的“主存储器”，叫做虚拟存储器。虚拟存储区的容量与物理内存无关，而受限于计算机的地址结构和可用磁盘容量。

计算机操作系统引入和使用虚拟存储技术的主要目的是提高系统的内存利用率和系统吞吐量。

24、影响缺页中断率有哪几个主要因素？

答：影响缺页中断率的因素有四个：

- (1) **页面大小。**页面划分大，则缺页率较低；反之，缺页率较高。
- (2) **进程所分配物理块的数目。**所分配的物理块数目越多，缺页率越低；反之则越高。
- (3) **页面置换算法。**算法的优劣对缺页中断率影响很大，但不可能找到一种最佳算法。
- (4) **程序固有特性。**程序本身的编制方法对缺页中断率有影响，根据程序执行的局部性原理，程序编制的局部化程度越高，相应执行时间的缺页程度越低。以数组为例，如果每一行元素存放在一页中，则按行处理各元素缺页中断率低；反之，按列处理各元素，则缺页中断率高。

25、虚拟存储器有哪些特征？简述之。

答：(1) **离散性。**指程序在内存中离散存储。

(2) **多次性。**指无需在作业运行时一次性地全部装入内存，而是允许被分成多次调入内存运行。

(3) **对换性。**指无需在作业运行时一直常驻内存，而是允许在作业的运行过程中，进行换进和换出。

(4) **虚拟性。**指从逻辑上扩充内存的容量，使用户所看到的内存容量，远大于实际的内存容量。

26、在虚拟分页存储管理方案中，对于一个处于运行状态的进程，当 CPU 读取下一条指令时，发生缺页中断。操作系统要执行哪些操作以获得所需要部分的指令？

答：在页表中发现所要访问的页不在内存中，则产生缺页中断。操作系统接到此中断信号后，就调出缺页中断处理程序，根据页表中给出的外存地址，将该页调入内存，使作业继续运行下去。若内存中有空闲块，则分配一页，将新调入页装入内存，并修改页表中相应页表项驻留位及相应的内存块号。若此时内存中没有空闲块，则要淘汰某页。若该页在内存期间被修改过，则要将其写回至外存。

27、覆盖技术与虚拟存储技术有何本质不同？交换技术与虚拟存储技术中使用的调入/调出技术有何相同与不同之处？

答：覆盖技术与虚拟存储技术最本质的不同在于覆盖程序段的最大长度要受内存容量大小的限制，而虚拟存储器中程序的最大长度不受内存容量限制，只受计算机地址结

构的限制。另外，覆盖技术中的覆盖段由程序员设计，且要求覆盖段中各个覆盖具有相对独立性，不存在直接联系或相互交叉访问；而虚拟存储技术对用户的程序段之间没有这种要求。

交换技术就是把暂时不用的某个程序及数据从内存移到外存中去，以便腾出必要的内存空间，或把指定的程序或数据从外存读到内存中的一种内存扩充技术。交换技术与虚存中使用的调入/调出技术的主要相同点是：都要在内存与外存之间交换信息。交换技术与虚存中使用的调入/调出技术的主要区别是：交换技术调入/调出整个进程，因此一个进程的大小要受内存容量大小的限制；而虚存中使用的调入/调出技术在内存和外存之间来回传递的是页面或分段，而不是整个进程，从而使得进程的地址映射具有了更大的灵活性，且允许进程的大小比可用的内存空间大。

28、为什么要在设备管理中引入缓冲技术？操作系统如何实现缓冲技术？

答：引入缓冲的主要原因有：

- (1) 缓和 CPU 与 I/O 设备间速度不匹配的矛盾。
- (2) 减少对 CPU 的中断频率，放宽对中断响应时间的限制。
- (3) 解决数据粒度不匹配的问题。
- (4) 提高 CPU 和 I/O 设备之间的并行性。

根据 I/O 控制方式，缓冲的实现方法有：

- (1) 采用专门硬件缓冲器。
- (2) 在内存划出一个具有 n 个单元的专用缓冲区，以便存放输入输出的数据。内存缓冲区又称为软件缓冲。

29、阐述计算机系统中缓冲的作用和分类。

答：为了缓和 CPU 和外设之间的矛盾，操作系统引入了单缓冲、双缓冲、循环缓冲以及缓冲池。

单缓冲：在 CPU 和外设之间设置了一个缓冲区，当有数据交换时，先把数据发往缓冲区，再冲缓冲区中读数据。

双缓冲：具有两个缓冲，当一个进程正在往一个缓冲区读数据的时候，操作系统可能正在读或写另外一个缓冲区。

循环缓冲：具有多个缓冲区的组合，它更加能够缓和 CPU 和外设之间速度的不匹配。

缓冲池：既可用于输入又可用于输出的公用缓冲池，在池中设置了多个可供若干个进程共享的缓冲区。

30、什么是 DMA 方式？它与中断方式的主要区别是什么？

答：DMA 是直接存储器存取。DMA 传输将数据从一个地址空间复制到另外一个地址空间。

当 CPU 初始化这个传输动作，传输动作本身是由 DMA 控制器来实行和完成。在实现 DMA 传输时，是由 DMA 控制器直接掌管总线，因此，存在着一个总线控制权转移问题。即 DMA 传输前，CPU 要把总线控制权交给 DMA 控制器，而在结束 DMA 传输后，DMA 控制器应立即把总线控制权再交回给 CPU。

它与中断方式的主要区别在于，DMA 方式只需要 CPU 在开始和完成传输时进行干预，其它时候不需要 CPU 干预。中断驱动方式在每个数据需要传输时中断 CPU，而 DMA 方式则是在所要求传送的一批数据全部传送结束时才中断 CPU。中断驱动方式数据传送是在中断处理时由 CPU 控制完成的，而 DMA 方式则是在 DMA 控制器的控制下完成的。

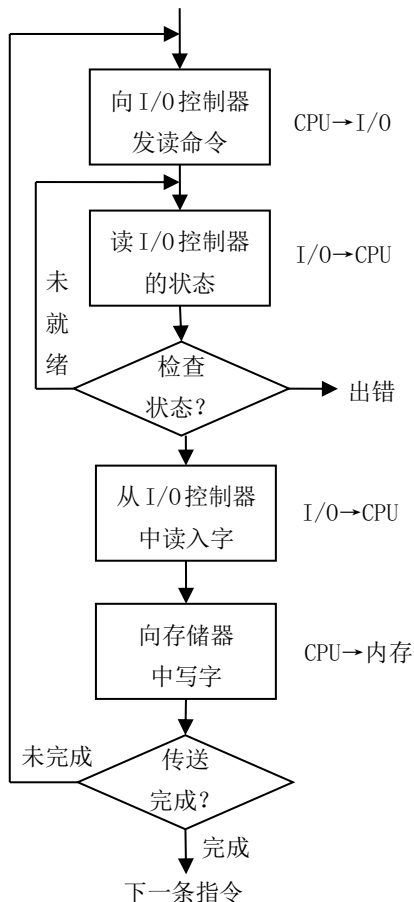
31、DMA 控制方式与通道控制方式有什么不同？（一个 DMA 对应一个设备，而一个通道可以对应多个设备）

答：在 DMA 控制方式中，DMA 控制器控制设备和主存之间成批地进程数据交流，而不用 CPU 干预。这样不但减轻了 CPU 的负担，而且提高了 I/O 数据传送速度。这种控制方式应用于块设备的数据传输。

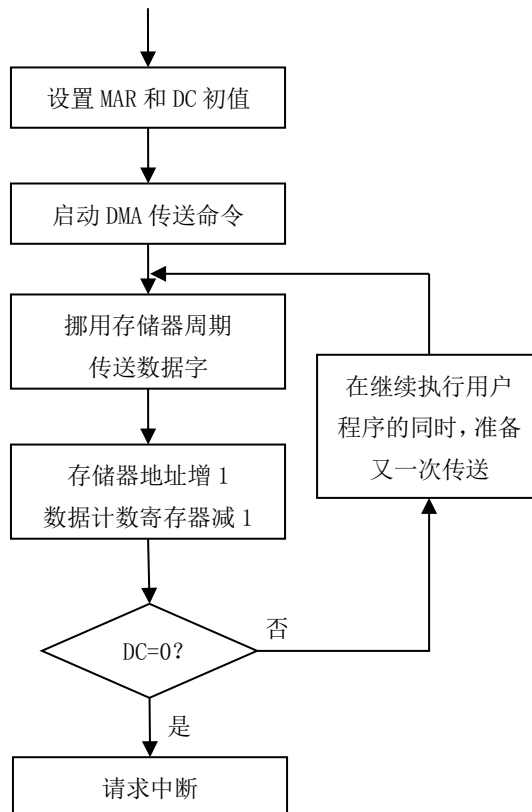
通道控制方式与 DMA 控制方式类似，也是一种以内存为中心，实现设备与内存直接交换数据的控制方式。在通道控制方式中，CPU 只需发出启动指令，指出通道相应的操作和 I/O 设备，该指令就可以启动通道并使通道从内存中调出相应的通道程序执行。与 DMA 相比，通道方式所需的 CPU 干预更少，并且可以做到一个通道控制多台设备，从而进一步减轻了 CPU 负担。

32、以从 I/O 设备读入数据为例，请用流程图方式说明程序 I/O、DMA 传输控制的处理过程。

答：程序 I/O 方式：



DMA 方式：



33、阐述什么是 SPOOLING 技术。

答：SPOOLING 技术是**同时联机外围操作技术**的简称。它是关于慢速字符设备如何与计算机主机进行数据交换的一种技术，通常又称假脱机技术。在多道程序环境下，利用多道程序中的一道或者两道来模拟脱机输入/输出中的外围控制机的功能，以达到“脱机”输入/输出的目的。利用这种技术可把独占设备转变成共享的虚拟设备，从而提高独占设备的利用率和进程推进速度。

34、为什么要引入 SPOOLING 系统？SPOOLING 系统可带来哪些好处？（比较难理解，但是必须看 10 遍以上掌握其中的精髓）

答：所有字符设备都是独占设备并属于慢速设备，因此，当一个进程在某台字符设备上交换数据时，往往要等待较长时间，并且在此进程未释放该设备之前，其它进程不能同时访问该设备，从而使这类设备成为系统的“瓶颈”。另一方面，分配到字符设备的进程，在整个运行期间也并非一直使用设备，利用率较低，从而降低了

整个系统的性能。SPOOLING 技术正是针对上述问题提出的一种技术。SPOOLING 技术的核心思想是利用一台可共享、高速、大容量的块设备来模拟独占设备的操作，使一台独占设备变成多台可并行使用的虚拟设备。SPOOLING 系统主要由输入井和输出井、输入缓冲区和输出缓冲区以及输入进程和输出进程 3 部分组成。在 SPOOLING 系统中，输入进程将用户要求的数据从输入设备送到输入井，当需要输入数据时，CPU 直接从输入井将数据读入内存；输出进程把用户要输出的数据先从内存送到输出井，等输出设备空闲时再将输出井中的数据输出到设备上。

SPOOLING 系统可带来如下好处：

- (1) 提高了 I/O 速度
- (2) 将独占设备改造成共享设备
- (3) 实现了虚拟设备功能

35、简述 SPOOLING 技术是如何模拟脱机外围设备操作的。

答：预输入程序模拟控制输入的外围机，缓输出程序模拟控制输出的外围机，输出井和输入井模拟脱机外围设备操作的两个磁盘。

36、按资源分配管理技术，输入输出设备类型可分为哪几类？

答：按资源分配管理的特点，输入输出设备可分为独享设备、共享设备和虚拟设备三类。

独享设备：不能共享的设备，一段时间只能由一个作业独占。如打印机、读卡机、磁带机等。所有字符型输入输出设备原则上都应是独享设备。

共享设备：可由若干作业同时共享的设备，如磁盘机等。共享分配技术保证多个进程可以同时方便地直接存取一台共享设备，共享提高了设备的利用率。块设备都是共享设备。

虚拟设备：利用某种技术把独享设备改造成多台同类型独享设备或共享设备。虚拟分配技术就是利用共享设备去虚拟独享设备，从而使独享设备成为可共享的、快速 I/O 的设备。实现虚拟分配的最有名的技术是 SPOOLing 技术，即假脱机技术。

37、为什么要引入设备独立性？如何实现设备独立性？（重点是逻辑设备表）

答：引入设备独立性可使应用程序独立于具体的物理设备。此时，用户用逻辑设备名来申请使用某类物理设备，当系统中有多台该类型的设备时，系统可以将其中的一台分配给请求进程，而不必局限于某一台指定的设备，这样可以显著改善资源的利用率及可适应性。独立性还可以使用户程序独立于设备的类型，如进行输出时，既可用显示终端，也可以用打印机。有了这种适应性，就可以很方便地进行输入/输出重定向。

为了实现设备独立性，必须在设备驱动程序之上设置一层设备独立性软件，用来执行所有 I/O 设备的公用操作，并向用户层软件提供统一接口。关键是系统中必须设置一张逻辑设备表（LUT）用来进行逻辑设备到物理设备的映射，其中每个表目中包含逻辑设备名、物理设备名和设备驱动程序入口地址；当应用程序用逻辑设备名请求分配 I/O 设备时，系统必须为它分配相应的物理设备，并在 LUT 中建立一个表目，以后进程利用该逻辑设备名请求 I/O 操作时，便可从 LUT 中得到物理设备名和驱动程序入口地址。

38、文件系统。

文件：以计算机硬盘为载体存储在计算机上的信息集合。

文件系统：操作系统中负责管理和存储文件信息的软件机构。

文件系统组成：与文件管理有关的软件、被管理的文件以及实施文件管理所需的数据结构。

文件分为：无结构文件（流式文件）、有结构文件（记录式文件）

目录：用于标识系统中的文件及其物理地址，供检索时使用的一种数据结构。

39、什么是顺序文件、索引文件、索引顺序文件和 HASH 文件？试说明它们的优点和缺点。

（超级重点！！逻辑结构）

答：**顺序文件**：指由一系列记录按照某种顺序排列所形成的文件。

优点：当需要对记录进行批量存取时，它的存取效率最高。

缺点：当文件较大时，记录的检索效率较低。

记录的增加和删除比较困难。

索引文件：索引结构为一个逻辑文件的信息建立一个索引表，索引表中的表目存放文件记录的长度和所在逻辑文件的起始位置，而逻辑文件中不再保存记录的长度信息。索引表本身是一个定长文件，每个逻辑块可以是变长的，索引表和逻辑文件两者构成了索引文件。

优点：可以随机访问，易于文件的增删。

缺点：增加了存储空间开销，索引表查找策略对文件系统效率影响很大。

索引顺序文件：索引顺序文件是顺序文件和索引文件两种形式的结合。索引顺序文件将顺序文件中的所有记录分为若干个组，为顺序文件建立一张索引表，并为每组中的第一个记录在索引表建立一个索引项，其中含有该记录的关键字和指向该记录的指针。索引表中索引项按照关键字顺序排列，索引顺序文件的逻辑文件（主文件）是一个顺序文件，每个分组内部的关键字不必有序排列，但是组与组之间的关键字是有序排列的。

优点：提高了存取的速度。

缺点：需要配置索引表而增加了存储开销。

HASH 文件：通过 HASH 函数建立关键字和相应记录物理地址之间的对应关系，可以直接通过关键字的值找到记录的物理地址，即关键字的值决定了记录的物理地址，这种结构的文件称为 HASH 文件。

优点：有很高的存取速度。

缺点：因不同关键词的 HASH 函数值相同而引起冲突。

40、文件的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三种，请分别简述它们的优缺点。**（超级重点！！物理结构）**

答：**连续结构**：在创建文件时需要给文件分配一组连续的盘块。

优点：顺序访问容易、速度快，当文件是定长时可以根据文件起始地址及记录长度进行随机访问。

缺点：文件存储要求连续的存储空间，会产生碎片，也不利于文件的动态扩充。

链接结构：可为文件分配多个不连续的盘块，再通过每个盘块上的链接指针，将同属于一个文件的多个离散的盘块链接成一个链表，由此所形成的物理文件称为链接文件。

优点：消除了磁盘的外部碎片，提高了外存的利用率。对插入、删除和修改记录都非常容易。能适应文件的动态增长，无需事先知道文件的大小。

缺点：只能按照文件的指针链顺序访问，查找效率低，指针信息存放消耗外存空间。

索引结构：为文件的每个分区单独建立一张索引表，该索引表记录了分配给该文件

的所有块号。

优点：直接访问和顺序访问的速度都比较快，易于文件的增删。

缺点：索引表增加存储空间开销，索引表的查找策略对文件系统效率影响较大。

41、链接结构。

链接分配分为：隐式链接和显式链接

隐式链接：用于链接物理块的指针隐式地放在每个物理块中。

显式链接：用于链接物理块的指针显式存放在内存的一张链接表中，每个磁盘设置一张链接表，也称文件分配表（FAT）

42、目录管理

目录管理的要求：（1）实现“按名存取”

（2）提高对目录的检索速度

（3）允许文件共享

（4）允许文件重名

单级目录：在整个文件系统中只建立一张目录表，每个文件占一个目录项。

实现了“按名存取”，但查找速度慢，不允许重名，不便于实现文件共享。

二级目录：将文件目录分成主文件目录和用户文件目录，系统为每个用户建立一个单独的文件用户目录。

解决了文件重名问题，并可以获得较高的查找速度，但二级目录缺乏灵活性，不能对文件分类，特别是当用户需要在某些任务上进行合作和访问其它文件时会产生很多问题。

多级目录：将两级目录结构的层次关系加以推广，就形成了多级目录结构，即树形目录结构。便于对文件分类，层次结构清晰，也能够更有效地进行文件的管理和保护；但在查找一个文件时，需要按照路径名逐级访问中间节点而增加了磁盘访问次数，进而影响了查询速度。

43、简述文件的检索过程。

答：一般文件的检索过程如下：

（1）由用户提供的路径和文件名，在目录文件中找到该文件目录。

（2）由文件目录（文件控制块）中所描述的信息查找到该文件存储的物理位置。

（3）根据文件存放的物理结构及其相应的存取方式，访问该文件。

44、文件存储空间的管理技术。

空闲表：空闲表法属于连续分配方式，为每个文件分配一块连续的存储空间，系统为外存上的所有空闲区建立一张空闲盘块表，表项包括第一个空闲块号、物理块号和空闲块数目等。

空闲链表：（1）空闲块链表是将文件存储设备上的所有空闲块链接在一起，形成一条空闲块链，并设置一个头指针指向空闲块链的第一个物理块。

（2）空闲盘区链表是将磁盘上的所有空闲盘区链接在一起，形成一条空闲盘区链，在每个盘区上除了含有用于指示下一个空闲盘区的指针外，还应有能指明本盘区大小的信息。

45、简述利用位示图进行文件存储空间管理的思想。这种方法的优缺点是什么？

答：若磁盘块空闲，则用“0”表示，若磁盘块已分配，则用“1”表示，从而得到一张位示图表，反映了所有磁盘块的信息。

优点：很容易找到一个或一组相邻接的空闲块。

缺点：整个磁盘的位示图文件比较大，在磁盘空闲块较少时，搜索空闲块要花费一些时间。

填空题

1. **分时系统特征**：多路性、独立性、及时性、交互性。**实时系统特征**：多路性、独立性、及时性、交互性、可靠性。
1. 操作系统的基本特征有并发性、共享性、虚拟性和异步性。
2. 分布式操作系统的特征有：分布性、并行性、共享性、透明性和健壮性。
3. 在分时操作系统中，改善响应时间的方法是：重入码和虚拟存储器。
在批处理兼分时的系统中，往往由分时系统控制的作业称为前台作业，而由批处理系统控制的作业称为后台作业。
4. 内核的基本功能是中断处理、进程管理、资源的基本操作。
5. 用户与操作系统的接口是：命令接口和程序接口。
6. 作业的控制方式有：脱机作业控制和联机作业控制。
7. 系统调用有五大类，它们是：进程控制类、文件操作类、设备管理类、信息维护和通信系统。
8. 采用客户/服务器模式构造一个操作系统的基本思想是，把操作系统划分为若干进程，其中的每一个各实现单独的一种服务。
9. 中断处理程序的主要工作是：保护现场、分析中断原因和处理发生的中断事件，在大多数情况下，中断处理程序往往简单处理完前两个工作后，就把具体的处理交给其它程序模块去做。
10. UNIX 的 shell 有两层含义，一是指由 shell 命令组成的shell 命令语言；二是指该命令的解释程序。
11. 操作系统提供给应用程序的接口是系统调用。
12. 要达到并发进程执行结果的可再现性，可采用Bernstein 条件实现。
13. 并发执行程序有三个特征，它们是：间断性、失去封闭性和不可再现性。
14. 若有两个进程 P1 和 P2 能并发执行，则 Bernstein 条件为： $R(P1) \cap W(P2) \cup R(P2) \cap W(P1) \cup W(P1) \cap W(P2) = \{ \}$ 。
15. 当一个进程独占处理器顺序执行时，具有三个特性：顺序性、封闭性和可再现性。
16. 进程可以并发执行，若干个并发执行的进程交替占用处理器，而进程各种状态的转换不是事先预定的，也不是完全由操作系统来确定的，而是在硬件和操作系统的相互配合下完成的，起主要作用的是中断系统。
17. 操作系统中，进程可以分为系统进程和用户进程两类。
18. 从结构上说，任一进程均由三部分组成，这三个组成部分分别是进程控制块、程序段和数据段。
19. 一个操作中的所有动作，要么全做，要么全不做，这叫做原子操作。
20. AND 机制能实现系统的安全性，并能防止死锁出现。
21. 采用信号量机制能保证程序的偏序执行。
22. 引起进程相互制约的两类原因是：互斥使用资源和进程间相互合作。
23. 互斥与同步解决方法有软件方法、中断方法、Test-and-Set 指令、Swap 指令和信号量方法。
24. 内存管理的主要功能有内存的分配和回收、地址映射、存储保护和扩充内存。
25. 利用硬件中断装置使得计算机操作系统可以控制各个程序的执行，为用户提供各种服务。主存储器是以字节为单位进行编址的。

26. 主存储器有多种管理方式,对不同的管理方式有不同的实现保护的方法,在每个程序占据主存连续空间的存储方式中,硬件设置两个寄存器:基址寄存器和限长寄存器,用来限定用户程序执行时可以访问的主存空间范围。
27. 在存储管理方案中,可用上、下限寄存器实现存储保护的是分区式存储管理。
28. 早期个人计算机的存储管理一般采用单用户连续存储管理方式。
29. 由固定分区发展为分页存储管理方式的主要推动力是提高内存利用率;由分页系统发展为分段系统,进而又发展为段页式系统的主要动力分别是:满足用户需要和既满足用户需要,又提高内存利用率。
30. I/O 控制方式有:程序直接控制方式、中断驱动方式、DMA 方式和通道控制方式。
31. 通道的类型有:字节多路通道、数组选择通道和数组多路通道。
32. 在 I/O 系统中引入缓冲的原因有:减少中断次数、提高并行性和缓解 CPU 与 I/O 速度不匹配的矛盾。
33. 通道程序解决了 I/O 操作的独立性和各部件工作的并行性,采用通道技术后,能实现 CPU 与通道的并行操作。
34. 由操作系统启动外围设备不仅可保证安全地使用外围设备,正确地传送信息,而且可减少用户为启动外围设备而必须了解外围设备特性及组织启动等工作,大大方便了用户。实现设备的独立性可采用逻辑设备表。
35. 设备管理的基本任务是:完成用户提出的 I/O 请求,提高 I/O 速度及改善 I/O 设备的利用率。
36. 主存储器与外围设备之间的信息传送操作称为DMA。
37. 在对打印机进行 I/O 控制时,通常采用中断驱动方式,对硬盘的 I/O 控制采用DMA方式。
38. 执行一次磁盘输入输出操作所花费的时间包括寻道时间、旋转延迟时间和传输时间。
39. 磁盘的驱动调度包括移臂调度和旋转调度部分,其目的是降低读写磁盘的总时间。
40. 操作系统利用共享设备来模拟独占设备的工作,为用户提供虚拟设备服务,实现虚拟设备必须要有一定的硬件和软件条件为基础,操作系统实现虚拟设备的功能模块是在计算机控制下通过联机的外围设备同时操作来实现其功能的,因此也把它称为SPooling。
41. 采用 SPooling 技术的计算机系统中,操作员只要启动预输入程序工作,就可以把作业存放到输入井中等待处理。
42. 为解决 HASH 冲突,可采用的方式有:二维表、加位移常量和溢出技术。
43. 文件系统中设置打开 (OPEN) 操作的目的是减少通道压力和提高访问效率。
44. 文件在文件存储空间的组织方式,成为文件的物理结构。
45. UNIX 系统,使文件描述信息单独形成一个数据结构叫索引结点。在文件目录中的每个目录项,仅由文件名和指向该文件 i 结点的指针构成。
46. 文件系统的主要目标是提高存储空间的利用率和减少存取时间。
47. 按照组织方式分类文件,可以将文件分为逻辑文件和物理文件。
48. 对于请求分页管理系统中,提取页面的策略有:请求调页策略和预调页策略。

知识点补充

1、简述计算机系统的中断机制及其作用。

答：中断机制包括硬件的中断装置和操作系统的中断处理服务程序。中断装置由一些特定的寄存器和控制线路组成，中央处理器和外围设备等识别到的事件保存在特定的寄存器中，中央处理器每执行完一条指令，均由中断装置判别是否有事件发生。若无事件发生，CPU 继续执行；若有事件发生，则中断装置中断原占有 CPU 的程序的执行，让操作系统的处理事件服务程序占用 CPU，对出现的事件进行处理，事件处理完后，再让原来的程序继续占用 CPU 执行。

2、选择进程调度算法的准则是什么？

答：由于各种调度算法都有自己的特性，因此，很难评价哪种算法是最好的。一般说来，选择算法时可以考虑如下一些原则：

- ① 处理器利用率；
- ② 吞吐量；
- ③ 等待时间；
- ④ 响应时间。

在选择调度算法前，应考虑好采用的准则，当确定准则后，通过对各种算法的评估，从中选择出最合适的算法。

3、作业在系统中有哪几种状态？

答：一个作业进入系统到运行结束，一般要经历进入、后备、运行和完成四个阶段，相应地，作业亦有进入、后备、运行和完成四种状态。

- ① 进入状态：作业的信息从输入设备上预输入到输入井，此时称为作业处于进入状态。
- ② 后备状态：当作业的全部信息都已输入，且由操作系统将其存放在输入井中，此时称作业处于后备状态。系统将所有处于后备状态的作业组成后备作业队列，等待作业调度程序的调度。
- ③ 运行状态：一个后备作业被作业调度程序选中，分配了必要的资源，调入内存运行，称作业处于运行状态。
- ④ 完成状态：当作业正常运行完毕或因发生错误非正常终止时，作业进入这完成状态。

4、UNIX 系统中的优先权和优先数有什么关系？如何确定进程的优先权和优先数？

答：UNIX 中每个进程都有一个优先数，就绪进程能否占用处理器的优先权取决于进程的优先数，优先数越小则优先权越高。

UNIX 以动态方式确定优先权，如核心的进程优先权高于进入用户态的进程；降低用完一个时间片的进程的优先权；对进入睡眠的进程，其等待事件越急优先数越高；降低使用处理器时间较长的进程的优先权。

UNIX 中确定进程优先数的方法有两种：设置方法和计算方法。前者对要进入睡眠状态的进程设置优先数，若等待的事件急迫，则设置较小的优先数；后者用户进程正在或即将转入用户状态运行时确定优先数。

5、计算机系统怎样实现存储保护？

答：一般硬件设置了基址寄存器和限长寄存器。

中央处理器在目态下执行系统中，对每个访问主存的地址都进行核对，若能满足：基址寄存器值 \leq 访问地址 \leq 基址寄存器值+限长寄存器值，则允许访问，否则不允许访问。并且不允许用户程序随意修改这两个寄存器的值。这就实现了存储保护。

6、给出系统总体上的中断处理过程。（重点记住过程）

答：CPU 每执行完一条指令就去扫描中断寄存器，检查是否有中断发生，若没有中断就继续执行下条指令；若有中断发生就转去执行相应的中断处理程序。中断处理过程可粗略的分为以下四个过程：

- ① 保护当前正在运行程序的现场；
- ② 分析是何种中断，以便转去执行相应的中断处理程序；
- ③ 执行相应的中断处理程序；
- ④ 恢复被中断程序的现场。

7、PCB 的作用是什么？为什么说 PCB 是进程存在的唯一标志？

PCB 作用：为了使在多道程序环境下，不能成为独立运行的单位程序成为一个能独立运行的基本单位，且能与其他进程并发执行的单位叫做进程。

PCB 唯一标志：因为 OS 通过某个进程的 PCB 对相关进程执行创建等操作，而不是根据其他别的什么而感知进程，因此 PCB 是进程存在的唯一标志。

8、什么是多线程？为什么要引入多线程？与多任务有何不同？

多线程是一个程序可以定义多个线程并发执行。每个线程承担不同任务。

引入原因：更高的提升系统并发度。

区别：多线程针对某个程序而言。代表一个程序完成的不同线程个数，每个线程承担不同任务。

多任务针对 OS 而言。代表 OS 可同时执行的任务个数。

9、什么是管程？管程组成部分及其特征是什么？

管程：一组数据及定义于此组数据上的操作所构成的软件模块。此组操作能出事后并改变管城中的数据和进程

组成部分：

1. 局部于管程的数据结构
2. 对管程内部数据的访问语句
3. 对管程内数据的初始化语句

管程特征：

1. 局部于管程内的数据只能被管城内过程所访问
2. 外部进程必须通过调用管城内过程才能访问内部数据
3. 管程每次仅允许一个进程执行内部过程
4. 管程是语法级操作，由编译器保证其正确性

10、进程与程序的区别和联系？

进程是一个程序在某个数据集上的一次活动，是一个动态的概念。进程由 PCB，程序段数据段组成

程序是一组有序指令的集合，是静态概念

- 1、进程是程序的一次执行过程，具有生命周期；程序是有序指令的集合，可以永久存在
- 2、进程可以执行多个程序。一个程序可以生成多个进程

3、进程和程序组成不同。

11、为什么要引入虚拟内存？虚拟内存空间大小由什么决定？虚拟内存是如何解决问题的？又会带来什么问题？

1、多道程序并发执行使得进程之间共享各种资源，如果同时运行的进程过多，则需要很多内存，当物理内存有限时，需要以其他方式扩充逻辑内存。

2、虚存大小小于 MEM+DISK 容量之和，也小于 OS 能寻址的最大存储容量。

3、虚存使用外存空间来扩充主存，通过换入换出技术使得系统可以使用远远大于物理内存的逻辑内存空间

由于许村技术调换页面需要访问外存，导致平均访问时间下降。若替换算法不合适，会严重降低系统性能。

12、为什么要引入动态重定位？什么情况下需要动态重定位？

动态重定位引入原因：编译程序编译产生的程序模块一般从 0 开始编址，是相对地址。实际执行时候需要从相对地址转换成物理地址，即重定位。

程序在运行过程中需要经常在内存中移动位置（因为内存管理机制一直在起作用），引入动态重定位的目的是满足此中需要。动态重定位实现需要软硬件的支持。

13、什么是虚存？如何实现页（段）式虚存？（超级无比重要的考点，务必背下来）

虚存是指具备请求调入和置换功能，可以从逻辑上扩充内存容量的存储系统。从用户观点而言，虚存具备比物理内存大得多的容量，其逻辑容量取决于 MEM+DISK 和 OS 的寻址空间，运行速度接近内存，单位存储成本接近外存。

实现页式虚存（段式虚存）

1. 扩充页表（段表）增加状态位标示所需要的段页是否在内存。增加外存地址以便调入页面。
2. 引入请求调页（请求调段）技术。将不再 MEM 中而又需要访问的段或者页从磁盘调入。硬件实现。
3. 置换技术。当内存中没有足够空间装载机将调入的页或者段时，应当将不用的一些段页调出内存。