

1. 什么是操作系统，目标和功能是什么，特征是什么？

操作系统是指控制和管理整个计算机的硬件和软件资源，合理的组织调度计算机的工作和资源的分配，提供给用户和其他软件方便的接口和环境的程序集合。他是一个系统软件。

目标：方便性，有效性，可扩充性，开放性。

功能：作为计算机系统资源的管理者（处存文设），作为用户和硬件系统之间的接口（命令接口和程序接口），用作扩充机器。

特征：并发，共享，虚拟，异步，其中并发和共享是最两个基本的特征。

并发是两个或多个事件在同一时间间隔内发生。

共享即资源共享，是指系统的资源可供内存中多个并发执行的进程共同使用。

虚拟是把一个物理上的实体变为若干个逻辑上的对应物。

异步是指进程的执行不是一贯到底的，而是走走停停，以不可预知的速度向前推进。

2. 何谓批处理操作系统？

批处理系统指用户的作业成批的处理，作业建立、处理、完成都自动由系统成批完成。引入批处理系统的目的是要解决人机矛盾及CPU与I/O设备之间速度不匹配，提高设备的利用率，提高系统吞吐量。

3. 什么是单道批处理系统，主要特征是什么？

单道批处理系统：系统对作业的处理是成批进行的，但内存中始终保持一道作业

自动性：磁带上的一批作业能自动的逐个依次运行，而无需人工干预。

顺序性：磁道上的各道作业是顺序地进入内存，各道作业的完成时间与他们进入内存的顺序基本一致

单道性：内存中仅有一道程序运行。

4. 什么是多道程序设计技术？基本特征是什么？引入技术哪些好处？需要解决什么问题？

多道程序技术就是同时把多个程序放入内存，允许它们交替在CPU中运行，共享系统资源。

当一道程序因I/O请求而暂停运行时，CPU便立即转向运行另一道程序。

多道程序运行的特征（特点）

多道：计算机内存中同时存放多道相互独立的程序。

宏观上并行：同时进入系统的多道程序都处于运行过程中，即先后开始了各自的运行，但都未运行完毕。

微观上串行：内存中的多道程序轮流占有CPU，交替执行。

优点是：资源利用率高（多道程序共享计算机资源，从而使各种资源得到充分利用），系统吞吐量大（CPU和其他资源保持忙碌状态）。

需要解决处理机，内存，设备分配情况，如何组织和存放大量的程序和数据，以使用户使用和保证其安全性与一致性。

5. 什么是分时技术？什么是分时系统？最关键的问题？基本特征？

分时技术：处理器的运行时间分成很短的时间片，按时间片轮流把处理器分配给各联机作业使用。

分时系统：多个用户通过终端同时共享一台主机，这些终端连接在主机上，用户可以同时与主机进行交互操作而不互相干扰。

最关键的问题：是如何使用户能与自己的作业进行交互，即当用户在自己的终端上输入命令时，系统能及时接收并及时处理该命令，再将结果返回用户。

同时性：也叫多路性，指允许多个终端用户同时使用一台计算机。

交互性：用户能方便地与系统进行人机交互。

独立性：系统中的多个用户可以彼此独立的进行操作，互不干扰。

及时性：用户请求能在很短时间获得响应。

6. 批处理系统和分时系统和实时操作系统各有什么特点？

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，系统响应时间长。

分时操作系统：多个用户同时使用计算机，人机交互能力强，具有每个用户独立地使用计算机的独占性，系统响应时间及时。

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

7. 为什么要处理器为什么要区分核心态和用户态两种操作方式？在什么情况下进行两种方式的切换？用户态转向核心态的例子？

管态：当执行操作系统程序时，处理机所处的状态

目态：当执行普通用户程序时，处理机所处的状态

区分执行态的主要目的是保护系统程序。用户态到和核心态的转换发生在中断产生时而核心态到用户态的转换则发生在中断返回到用户程序时。

系统调用，发生一次中断，用户程序产生错误状态和企图执行以条特权指令，执行特权指令。

8. 试说明访管指令、特权指令和原语

访管指令是一类机器指令，执行访管指令可以引起访管中断

特权指令是计算机中不允许用户直接使用的指令。

系统调用是用户在程序中调用操作系统所提供的一些子功能，是提供编程人员的接口。

原语是指由若干条机器指令构成，并用于完成特定功能的一段程序，在执行期间不可分割。主要特点是不可分割性。基本特点：最底层最接近硬件的部分，具有原子性——其操作只能一气呵成，运行时间短和调用频繁。

9. 什么是系统调用？执行过程？与一般调用有什么区别？

所谓系统调用是用户在程序中调用操作系统所提供的一些子功能，是提供编程人员的接口。

通过系统调用命令，中断现行程序而转去执行相应的子程序，以完成特定的系统功能。完成后，又返回到发出系统调用命令之后的一条指令，被中断的程序将继续执行下去。

系统调用与一般过程调用不同，其主要区别是：

运行的状态不同。在程序中的过程一般或者都是用户程序，或者都是系统程序，即都是运行在同一个系统状态的（用户态或系统态）。进入的方式不同。一般的过程调用可以直接由调用过程转向被调用的过程。而执行系统调用时，由于调用过程与被调用过程是处于不同的状态，因而不允许由调用过程直接转向被调用过程，通常是通过访问管中断（即软中断）进入，先进入操作系统，经分析后，才能转向相应的命令处理程序。返回方式的不同。代码层次不同。一般过程调用中的被调用程序是用户级程序，而系统调用是操作系统中的代码程序，是系统级程序。

9. 什么是中断？中断处理的一般过程分为哪几个阶段？用哪几种？

所谓中断是指 CPU 对系统发生的某个事件（中断源）作出的一种反应：CPU 暂停正在执行的程序，保留现场后自动地转去执行相应的处理程序，处理完该事件后再返回断点继续执行被“打断”的程序。中断处理的一般过程分为以下阶段：保存现场，分析原因，处理中断，返回断点。

中断：也称外中断，指来自于 CPU 执行指令以外的事件发生，如设备发出的 I/O 结束中断。

异常：也称内中断，例外或陷入，指来自于 CPU 执行指令内部的事件发生。

10. 为什么说直到出现中断和通道技术后，多道程序概念才变为有用的？

多道程序并发执行是指有的程序正在 CPU 上执行，而另一些程序正在 I/O 设备上传输。在时间上的重叠必须有中断和通道技术支持其原因如下：1. 通道是一种控制一台或多台外部设备的硬件机构，它一旦被启动就独立于 CPU 运行，因而做到了 I/O 设备与 CPU 并行工作。但早期 CPU 通过向通道发出询问指令来了解通道工作是否完成。若未完成则主机就循环询问直到通道工作结束为止。因此这种询问方式是无法真正做到并行工作的。2)在硬件上引

入了中断技术。所谓中断是指 CPU 对系统发生的某个事件（中断源）作出的一种反应：CPU 暂停正在执行的程序，保留现场后自动地转去执行相应的处理程序，处理完该事件后再返回断点继续执行被“打断”的程序。因此通道技术和中断技术结合起来就可以实现并行工作。即 CPU 启动通道传输数据后便去执行其他程序的计算工作而通道则进行输入/输出操作；当通道工作结束时再通过中断机构向 CPU 发出中断请求 CPU 则暂停正在执行的操作对出现的中断进行处理处理完后再继续原来的工作。这样就真正做到了 CPU 与 I/O 设备并行工作。此时多道程序的概念才变为现实。

1.什么是进程？为什么引入进程？进程和程序的区别？

答：进程是具有独立功能的程序在一个数据集合上运行的过程，他是系统进行资源分配和调度的一个独立单位。在多道程序环境下，允许多个程序并发执行，此时他们将失去封闭性，并具有间断性和不可再现性的特征。为此引入了进程的概念，以便更好地描述和控制程序的并发执行，实现操作系统的并发性和共享性。引入进程的目就是为了是程序能与去其他进程的程序并发执行，以提高资源利用率。

进程是动态，程序是静态的；

进程是独立运行的单位，程序不能作为运行单位；

进程间在并发执行过程中会产生相互制约关系，而程序由于是静态的，所以不存在异步特征。

2.进程的最主要的特征有哪些？

动态性：进程是程序的一次执行，他有着创建、活动、暂停、终止等过程，具有一定的生命周期，是动态的产生、变化和消亡的。动态性是进程最基本的特征

并发性：多个进程实体，同存于内存中，能在一段时间内同时运行

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收调度的基本单位。

异步性：每个进程都以其相对独立、不可预知的速度向前推进

结构性：每个进程有一个控制块 PCB

3.什么是进程实体？什么是 PCB？为什么要引入 PCB？为什么说 PCB 是进程唯一存在的标志？PCB 包含哪些内容？

进程实体由程序段、相关数据段和 PCB 三部分组成。

进程控制块（PCB）是记录进程的动态执行情况的一种数据结构。为了使参与并发执行的程序能独立的运行，必须为之配置一个专门的数据结构。

每个被创建的进程都由惟一的 PCB 来标识，操作系统根据 PCB 对进程实施控制和管理；当一个进程完成它的工作被系统撤销时，它的 PCB 也被撤销。因此，PCB 是进程存在的惟一标志，进程的动态、并发等特征都是通过 PCB 表现出来的。

PCB 主要包括：进程描述信息、进程控制和管理信息、资源分配清单和处理器相关信息等。

4.说明进程在三个基本状态之间转换的典型原因。

A. 处于就绪状态的进程，当进程调度程序为之分配了处理机后，该进程便由就绪状态变为执行状态。

B. 当前进程因发生某事件而无法执行，如访问已被占用的临界资源，就会使进程由执行状态转变为阻塞状态。

C. 当前进程因时间片用完而被暂停执行，该进程便由执行状态转变为就绪状态。

5.什么是进程控制？什么是原语？进程控制原语主要有哪些？

进程控制的主要功能是对系统中所有进程实施有效地管理，她具有创建新进程、撤销已有进程、实现进程状态转换等功能。原语是指由若干条机器指令构成，并用于完成特定功能的一段程序，在执行期间不可分割。主要特点是不可分割性。基本特点：最底层最接近硬件的部分，具有原子性——其操作只能一气呵成，运行时间短和调用频繁。

创建原语，撤销原语，挂起原语，激活原语，阻塞原语，唤醒原语。

6.创建原语过程？终止原语过程？

创建原语：为新进程分配一个唯一的进程标识号，并申请一个空白的 PCB 。

为进程分配资源。为新进程的程序和数据，以及用户栈分配必要的空间。

初始化 PCB。主要包括初始化标志信息、初始化处理器状态信息和初始化处理器控制信息，以及设置进程的优先级。

1) 如果进程就绪队列能够接纳新进程，就将新进程插入到就绪队列，等待被调度运行。

终止原语：据被终止进程的标识符，检索 PCB，从中读出该进程的状态。

2) 若被终止进程处于执行状态，立即终止该进程的执行，将处理器资源分配给其他进程。

3) 若该进程还有子进程，则应将其所有子进程终止。

4) 将该进程所拥有的资源、或归还给父进程或归还给操作系统。

5) 将该 PCB 从所在队列（链表）中删除。

7.进程的切换是什么？过程是怎么样的？

进程切换是指处理机从一个进程的运行转到另一个进程运行。

保存处理器上下文，包括程序计数器和其他寄存器。更新 PCB 信息。把进程的 PCB 移入相应的队列，如就绪、在某时间阻塞等队列。选择另一个进程执行，并更新其 PCB。更新内存管理的数据结构。恢复处理器的上下文。

8.为什么需要系统调用来实现通信？什么是进程的通信？进程通信有哪些？

每个进程有自己相互独立的地址空间。在操作系统和硬件的地址机构保护机制下，进程无法访问其他进程的地址空间，所以必须借助于操作系统的系统调用函数实现进程之间的通信。进程通信就是进程之间的数据交换。低级通信方法和高级通信方法。高级通信方法可分为共享存储、消息传递和管道

共享存储：在通信的进程之间存在一块可直接访问的共享空间，通过对这片共享空间进行读/写操作实现进程之间的信息交换。在对共享空间进行操作时，需要使用同步互斥工具（如 P 操作、V 操作）共享存储又分为两种：低级方式的共享是基于数据结构的共享；高级方式则是基于存储区的共享。

消息传递：在消息传递系统中，进程间的数据交换是以格式化的消息(Message)为单位的。进程通过系统提供的发送消息和接收消息两个原语进行数据交换。

1) 直接通信方式：发送进程直接把消息发送给接收进程，并将它挂在接收进程的消息缓冲队列上，接收进程从消息缓冲队列中取得消息。

2) 间接通信方式：发送进程把消息发送到某个中间实体中，接收进程从中间实体中取得消息。这种中间实体一般称为信箱。

管道：消息传递的一种特殊方式。所谓“管道”，是指用于连接一个读进程和一个写进程以实现它们之间通信的一个共享文件，又名 pipe 文件。向管道提供输入的写进程，以字符流形式将大量的数据送入管道；而接收管道输出的读进程，则从管道中读数据。为了协调双方的通信，管道机制必须提供以下三方面的协调能力：互斥、同步和确定对方的存在。

9.什么是线程？为什么引入线程？线程和进程的比较？

线程是进程的一个实体,是系统独立调度和分派的基本单位，线程自己不拥有系统资源（只拥有一点在运行中必不可少的资源）但线程可以访问其隶属进程的系统资源。

而引入线程,则是为了减小程序在并发执行时所付出的时空开销,提高操作系统的并发性能。

调度性。传统操作系统中,拥有资源和调度的基本单位是进程。在引入线程的 OS 中,进程是拥有资源的基本单位,线程是调度的基本单位。

并发性。引入线程的 OS 中,进程可以并发,一个进程的多个线程也可以并发,不同进程的线程也可以并发。

拥有资源。不管传统操作系统还是有线程的操作系统进程都是拥有资源的基本单位, 线程

不拥有系统资源（只拥有一点在运行中必不可少的资源），但线程可以访问其隶属进程的系统资源。

开销。创建和撤消进程时，必须为之分配和回收资源，因而付出的开销要明显大于线程

10. 什么是作业调度，进程调度，内存调度？什么是作业？

作业调度：用于决定把外存上处于后备队列中的哪些作业调入内存，并为它们创建进程、分配必要的资源，再将新创建的进程排在就绪队列上，准备执行

进程调度：用来决定就绪队列中的哪个进程应获得处理机。

内存调度：又称中级调度，主要任务是按照给定的原则和策略，将处于外存对换区中的重新具备运行条件的进程调入内存，或将内存中暂时不能运行的进程交换到外存对换区。

作业：用户在一次上机过程中要求计算机系统所做工作的集合

引起进程调度：有进程运行完毕，进程调用阻塞原语，p 操作时资源不足，v 操作激活等待队列的进程，时间片用完（分时系统），优先级跟高的进程到来（抢占式调度）

11. 非抢占式和可抢占式高优先级调度算法的区别是什么？调度的准则是什么？

最高优先级调度算法原则上总是调度就绪队列中优先级最高的那个进程。采用非抢占式最高优先级调度算法，当就绪队列中某进程的优先级高于正在处理器中运行的进程的优先级时，并不会让正在运行的进程退出处理器，而是将高优先数的排在就绪队列的首部。而采用抢占式最高优先级进程调度算法，则高优先数的进程会抢占处理器，让正在处理的进程处于就绪队列。

CPU 利用率

系统吞吐量（表示单位时间内 CPU 完成作业的数量）

周转时间（是指从作业提交到作业完成所经历的时间）

等待时间（进程处于等处理器状态时间之和）

响应时间（指从用户提交请求到系统首次产生响应所需的时间）

12. 典型的调度算法优缺点？

先来先服务(FCFS)：是一种最简单的调度算法，即可用于作业调度，也可用于进程调度。按照作业/进程进入系统的先后次序进行调度，先进入系统者先调度。算法简单，但效率低。比较有利于长作业，而不利于短作业。有利于 CPU 繁忙型作业，而不利于 I/O 繁忙型作业。

短作业优先调度算法(SJF)是从队列中选出一个估计运行时间最短的作业优先调度，即可用于作业调度，也可用于进程调度。对长作业不利。严重的是，若一长作业进入系统的后备队列，由于调度程序总是优先调度那些短作业，将导致长作业长期不被调度——饥饿。完全未考虑作业的紧迫程度，因而不能保证紧迫性作业会被及时处理。由于作业的长短只是根据用户所提供的估计执行时间而定的，而用户又可能会有意或无意地缩短其作业的估计运行时间，致使该算法不一定能真正做到短作业优先调度。

非抢占式和可抢占式高优先级调度算法

静态优先权是在创建进程时确定，且在进程的整个运行期间保持不变。

动态优先权是在进程运行过程中根据进程的情况变化的动态调整优先级。

高响应比优先调度算法：既考虑作业的执行时间也考虑作业的等待时间，综合了先来先服务和最短作业优先两种算法的特点。优点是等待时间相同的作业，则要求服务的时间愈短，其优先权愈高，对短作业有利。要求服务的时间相同的作业，则等待时间愈长，其优先权愈高，是先来先服务。长作业优先权随等待时间的增加而提高，其等待时间足够长时，其优先权便可升到很高，从而也可获得处理机，对长作业有利。是一种折中，既照顾了短作业，又考虑了作业到达的先后次序，又不会使长作业长期得不到服务。 缺点：要进行响应比计算，增加了系统开销

简单的时间片轮转法：系统将所有的就绪进程按先来先服务的原则排成一个队列，每次调度时，把 CPU 分配给队首进程，并令其执行一个时间片；当执行的时间片用完时，由一个计时器发出

时钟中断请求，调度程序便停止该进程的执行，并将其放就绪队列尾；然后，再把处理机分配给就绪队列中新的队首；时间片的大小从几 ms 到几百 ms

缺点：紧迫任务响应慢。时间片选取 太小，会频繁发生中断、进程上下文切换，增加系统开销，但利于短作业。太大，退化成 FCFS 。

13.为什么说多级反馈队列能够较好的满足要求？

对于终端型用户来说，提交的大多数都是较小的交互型，通常可在第一队列规定的时间片内让其完成工作，使终端型用户都感到满意；对短批处理作业用户来说，在第一队列执行一个时间片或至多只在第二队列和第三队列各执行一个时间片即可完成，周转时间仍然很短。对长批处理作业用户，只要将作业依次在第 1,2,...n 队列中运行，然后按轮转方式运行，用户不必担心作业长期得不到处理。

1.什么是进程的同步与互斥？什么是临界资源？什么是临界区？临界资源的访问过程有哪些？

进程互斥：也称间接制约关系。当一个进程进入临界区使用临界资源时，另一个进程必须等待，当占用临界资源的进程退出临界区后，另一个进程才允许去访问此临界资源。

进程同步：也称相互制约关系，指多个相关进程在协调他们的工作次序上而产生的制约关系。临界资源是在一段时间内，只允许一个进程访问的资源。每个进程中访问临界资源的那段程序称为临界区。

临界资源的访问过程分为四个部分。进入区，临界区，退出区，剩余区。

2.简述信号量 S 的物理含义。

$S > 0$ 时，S 表示可使用的资源数；或表示可使用资源的进程数；

$S = 0$ 时，表示无资源可供使用；或表示不允许进程再进入临界区；

$S < 0$ 时， $-S$ 表示等待使用资源的进程个数；或表示等待进入临界区的进程个数；

当 $S > 0$ 时，调用 $P(S)$ 的进程不会等待；调用 $V(S)$ 后使可用资源数加 1 或使可用资源的进程数加 1；

当 $S < 0$ 时，调用 $P(S)$ 的进程必须等待；调用 $V(S)$ 后将释放一个等待使用资源者或释放一个等待进入临界区者。

3.什么是管程？管程的组成部分？基本特性？为什么引入？

管程是一组数据以及定义在这组数据之上的对这组数据的操作组成的软件模块

- 1) 局部与管程的共享结构数据说明
- 2) 对该数据结构进行操作的一组过程
- 3) 对局部于管程的共享数据设置初始值的语句

管程的基本特性

- 1) 局部于管程的数据只能被局部于管程内的过程访问。
- 2) 一个进程只有通过调用管程内的过程才能进入管程访问共享数据。
- 3) 每次仅允许一个进程在管程内执行某个内部过程。

解决临界区分散所带来的管理和控制问题。

4. 什么是死锁？死锁的四个必要条件是什么？以及原因？

死锁是指多个进程由于竞争资源而造成的一种僵局（互相等待），若无外力作用，它们都将无法推进下去。

原因：系统资源的竞争和进程推进顺序非法。

必要条件：互斥条件，不剥夺条件，请求和保持条件，循环等待条件。

进程对所分配到的资源进行排它性的使用，即在一段时间内某资源仅为一个进程使用
进程已获得的资源在未使用完之前不能被剥夺

进程已经至少保持了一个资源，但又提出了新的资源请求，而该资源又已被其他进程占有，在发生死锁时，必然存在一个进程资源的循环等待链，已获得的资源被下一个进程所请求。

5. 简述解决死锁问题的三种方法。

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。（安全状态是指系统能按某种进程推进顺序，为每个进程分配其所需的资源，直至满足每个进程对资源的最大需求，使每个进程都可以顺序地完成）

③ 死锁的检测与解除。对资源的申请和分配不加限制，只要有剩余的资源就呆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。

6. 死锁的预防四个方法？

破坏互斥条件：允许系统资源都能共享使用。

破坏不剥夺条件：当一个以已保持了某些不可剥夺资源的进程，请求新的资源时得不到满足，必须释放已经保持的所有资源，待以后需要时再重新申请。

破坏请求和保持条件：使用预先静态分配方法，即进程在运行前一次申请完他所需要的全部资源，在他的资源未满足前，不把它投入运行。一旦运行后，这些资源就一直归它所有，也不再提出其他资源请求，不会发生死锁，但是系统资源严重浪费，而且还会导致“饥饿”现象。

破坏循环等待条件：使用顺序资源分配法。首先给系统中的资源编号，规定每个进程，必须按编号递增的顺序请求资源，同类资源一次申请完。

7.死锁的解除有几种方法？

资源剥夺法。挂起某些死锁进程，并抢占它的资源，将这些资源分配给其他的死锁进程。但应防止被挂起的进程长时间得不到资源时，而处于资源匮乏的状态。

进程撤销法。强制撤销一个或一部分进程并剥夺这些进程的资源。撤销的原则可以按进程的优先级和撤销进程代价的高低进行。

进程回退法。让一个或多个进程回退到足以回避死锁的地步，进程回退时资源释放资源而不是被剥夺。要求系统保持进程的历史信息，设置还原点。

8.同步机制应遵循哪些准则？

空闲让进：临界区空闲时，可以允许一个请求进入临界区的进程进入临界区。

忙则等待：当已有进程进入临界区，其他试图进入临界区的进程必须等待。

有限等待：对请求访问的进程，应保证能在有限时间内进入临界区。

让权等待：当进程不能进入临界区时应立即释放处理机，防止进程忙等待。

9.何谓用户级线程和内核支持线程？有何区别？

答：（1）用户级线程：仅存在于用户空间中的线程，无须内核支持。

（2）内核支持线程：在内核支持下运行的线程。

系统型线程依赖内核；用户型线程不依赖内核。

系统型线程是由操作系统内核完成创建和撤销的线程；用户型线程是由应用程序利用线程库提供创建，同步，调度和管理线程函数来控制的线程。

当一个系统型线程因 I/O 操作阻塞时，不会影响其他进程的运行；由于操作系统不了解用户级线程的存在，所以当有一个线程阻塞时，整个进程必须等待。

1.在虚拟段式存储系统中，引入了段的动态连接。

a 试说明为什么引入段的动态连接。

b 请给出动态连接的一种实现方法。

(1) 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。称为静态链接。静态链接常常因为目标模块个数多而花费大量的 CPU 时间，而实际运行时又常常只用到其中的部分模块，因而也造成了存储空间的浪费。动态链接是作业运行时先装入主程序，运行过程中需要某模块时，再将该模块的目标程序调入内存并进行链接，它克服了静态链接的不足。

(2) 分段存储管理就是最典型的动态链接。分段管理允许用户将作业按逻辑关系进行自然分段，各段的大小可以不同。逻辑段内的地址是由两部分组成的（s：段号，d：段内位移量），即分段地址空间是用户定义的二维空间。内存分配以段为单位，段可以在作业运行过程中根据请求而动态链接和装入。

2.何为静态链接？何谓装入时动态链接和运行时动态链接？

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，再装入内存时，采用边装入变链接的方式。

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

3.绝对装入方式？

绝对装入：在编译时，如果知道程序将驻留在内存的某个位置，编译程序将产生绝对地址的目标代码。绝对装入程序按照装入模块的地址，将程序和数据装入内存。装入模块被装入内存后，由于程序中的逻辑地址与实际地址完全相同，故不需对程序 and 数据的地址进行修改。绝对装入方式只适用于单道程序环境。另外，程序中所使用的绝对地址，可在编译或汇编时给出，也可由程序员直接赋予。

4.为什么要引入动态重定位？如何实现？

答：静态重定位是地址变换在链接装入时一次完成的，但它要求连续的一片区域，且重定位后不能移动，不利于内存空间的有效使用，所以要引入动态重定位。动态重定位是在程序运行过程中要访问数据时再进行逻辑地址与物理地址的变换它是靠硬件地址变换部分实现的，通常采用重定位寄存器等实现

5.什么是逻辑地址？什么是物理地址？逻辑地址空间与物理地址空间？什么是地址重定位？什么是碎片？

逻辑地址指由程序产生的与段相关的偏移地址部分。

物理地址是在存储器里以字节为单位存储信息，为正确地存放或取得信息，每一个字节

地址重定位：把逻辑地址转变为内存的物理地址的过程。在装入时对目标程序中指令和数据的修改过程。

逻辑地址空间：一个目标程序所限定的地址范围。

物理地址空间实质内存中物理单位的集合，它是地址转换的最终地址。

碎片是指内存中很多容量太小、无法被利用的空闲块。

6.什么是覆盖技术？什么是交换？什么是换入和换出？

覆盖技术是指一个程序的若干程序段或几个程序的某些部分共享某一个存储空间。

交换的基本思想是：把处于等待状态的进程从内存移到辅存，把内存空间腾出来，这一过程又叫换出；把准备好竞争 CPU 运行的进程从辅存移到内存，这一过程又称为换入。

交换技术主要是在不同进程之间进行，而覆盖则用于同一个程序中。

7.覆盖技术与虚拟存储技术有何本质不同?交换技术与虚存中使用的调入/调出技术有何相同

覆盖技术与虚拟存储技术最本质的不同在于覆盖的程序段的最大长度要受到物理内存容量的限制，而虚拟存储器的最大长度不受物理内存容量的限制，只受计算机地址结构的限制。交换技术与虚存中使用的调入/调出技术的主要区别在于：交换技术换进换出整个进程，因此一个进程的大小受物理存储器的限制；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存储页或存储段，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物理存储空间大得多。

8.什么是虚拟存储器?虚拟存储器基本特征是什么?虚拟存储器的容量主要受到什么限制?试着举一例子?

虚拟存储器是指具有请求调入功能和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统。

基本特征：多次性（作业无需一次装入内存，分为多次调入内存运行），

对换性（作业运行时不必一直常驻内存，允许换入和换出。），

虚拟性（逻辑上扩充了容量，使用户看到的内存容量，远大于实际的内存容量。）

虚拟存储器的容量与物理主存大小无关，而受限于计算机的地址结构。

例如在请求分页存储管理系统中，用户作业的所有页面并不一定都在实存，在作业运行过程中再请求调入所用的虚页。为了实现从逻辑地址空间到物理地址空间的变换，在硬件上必须提供一套地址变换机构，动态地址变换机构自动地将所有的逻辑地址划分为页号和页内地址两部分，并利用页表将页号代之以块号，把块号和页内地址拼接就得到了内存的物理地址，从而实现了虚拟存储器。

7.简述固定分区和可变分区在管理方式上的区别?和优缺点?

答：单一连续分配方式：将内存分为系统区和用户区，系统区供操作系统使用，用户区供用户使用，是最简单的一种存储方式，但只能用于单用户单任务的操作系统中

固定分区是一种最简单的多道程序存储管理方式，它将用户内存空间划分为若干个固定大小的区域，每个分区只装入一道作业。程序可能太大而放不进任何一个分区中，这时用户不得不使用覆盖技术使用内存空间。主存利用率低，当程序小于固定分区大小时，也占用了一个完整的内存空间，会有内部碎片。

可变分区是一种动态划分内存的分区方法。不预先将内存划分，而是在作业装入内存时，根据作业的大小动态的建立分区，并使分区的大小正好适合作业的需要。分区的大小数目可变引入可变分区方法，使内存分配有较大的灵活性，也提高了内存利用率。但是可变分区会引起碎片的产生。

9.分区存储管理中常采用哪些分配策略?比较它们的优缺点

分区存储管理中常采用的分配策略有：首次适应算法、循环首次适应算法、最佳适应算法、最坏适应算法。

首次适应算法的优缺点：保留了高址部分的大空闲区，有利于后到来的大型作业的分配；低址部分不断被划分，留下许多难以利用的、小的空闲区，且每次分区分配查找时都是从低址部分开始，会增加查找时的系统开销。

循环首次适应算法的优缺点：使内存中的空闲分区分布得更为均匀，减少了查找时的系统开销；缺乏大的空闲分区，从而导致不能装入大型作业。

最佳适应算法的优缺点：每次分配给文件的都是最适合该文件大小的分区；内存中留下许多难以利用的小的空闲区。

最坏适应算法的优缺点：给文件分配分区后剩下的的空闲区不至于太小，产生碎片的几率最小，对中小型文件分配分区操作有利；使存储器中缺乏大的空闲区，对大型文件的分区分配不利。

10. 分段和分页的主要区别是什么？

页是信息的物理单位；而段是信息的逻辑单位；

页的大小固定且由系统决定；而段的长度却不固定，段含有一组意义相对完整的信息，决定于用户所编写的程序。

分页的作业地址空间是一维的；而分段的作业地址空间是二维的。

分页是出于系统管理的需要，分段是为了满足用户的需要

分页中有内碎片，无外碎片。分段无内碎片，有外碎片。

11. 为什么说分段系统较之分页系统更易于实现信息共享和保护？

a. 对于分页系统，每个页面是分散存储的，为了实现信息共享和保护，则页面之间需要一一对应起来，为此需要建立大量的页表项；

b. 而对于分段系统，每个段都从 0 开始编址，并采用一段连续的地址空间，这样在实现共享和保护时，只需为所要共享和保护的程序设置一个段表项，将其中的基址与内存地址一一对应起来即可。

12. 何为页表和快表？它们各起什么作用？

页表指出逻辑地址中的页号与所占主存块号的对应关系。快表是具有并行查找能力的高速缓冲存储器，又称联想寄存器 TLB，用以存放当前访问的若干页表项。

作用：页式存储管理在用动态重定位方式装入作业时，要利用页表做地址转换工作。

由于采用页表做地址转换，读写内存数据时 CPU 要访问两次主存。有了快表，有时只要访问一次高速缓冲存储器，一次主存，这样可加速查找并提高指令执行速度。

补充：段页式存储器管理存取一次数据须经过 3 次对内存的访问？

首先通过段表查到页表起始地址，然后通过页表找到帧号，最后形成物理地址。

13. 在有快表的情况下地址转换过程？

在具有快表的分页机制中，地址的变换过程：CPU 给出有效地址后，由硬件进行地址转换，并将页号送入高速缓存寄存器，并将此页号与快表中的所有页号同时进行比较。

如果有找到匹配的页号，说明索要访问的页表项在快表中，则可以直接从中读出该页对应的页框号，送到屋里地址寄存器。这样存取数据可以直接一次访存实现。

如果没有找到，则需要访问主存中的页表，在读出页表项后，应同时将其存入快表中，以供后面可能的再次访问。但是如果快表已满，就必须按照一定的算法对其中旧的页表项进行替换。注意，有些处理器设计为快表和主存同时查找，如果在快表中匹配成功则终止主存中的查找

13. 虚拟存储器的基本特征是什么？虚拟存储器的容量主要受到什么限制？

虚拟存储器是指具有请求调入功能和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统。

基本特征：多次性（作业无需一次装入内存，分为多次调入内存运行），

对换性（作业运行时不必一直常驻内存，允许换入和换出。），

虚拟性（逻辑上扩充了容量，使用户看到的内存容量，远大于实际的内存容量。）

虚拟存储器的容量与物理主存大小无关，而受限于计算机的地址结构和可用磁盘容量。

14. 请求页式存储管理的优缺点

答：优点：

(1) 虚存量大，适合多道程序运行，用户不必担心内存不够的调度操作。动态页式管理提供了内存与外存统一管理的虚存实现方式。

(2) 内存利用率高，不常用的页面尽量不留在内存。

(3) 不要求作业连续存放，有效地解决了“碎片”问题。与分区式比，不需移动作业；与多重分区比，无零星碎片产生。注：分区式分配包括：固定式分配，可变分区分配，可重定位分

区分配和多重分区分配四种。

缺点：

(1)要处理页面中断、缺页中断处理等，系统开销较大。

(2)有可能产生“抖动”。注：刚刚换出的页面马上又换入内存，刚刚换入的页面马上又换出内存，这种频繁的调度行为叫做抖动

(3)地址变换机构复杂，为提高速度采用硬件实现，增加了机器成本。

15.什么是缺页中断？缺页中断之后需要怎么处理？与一般的中断有什么区别？

在请求分页系统中，每当所要访问的页面不在内存时，便产生一个缺页中断，请求操作系统将所缺的页调入内存。

过程：缺页中断处理程序根据页面在外存的位置将其调入内存。在此过程中，内存中如果有空闲空间，则缺页中断处理程序会将该页面调入任一空闲存储块，还需对页表的其他表项做修改，如物理块号等。如果没有空闲空间，必须淘汰某些页面，如果被淘汰的页面之前被修改过，要将其写回内存。

区别：在指令执行期间产生和处理中断信号，而非一条指令执行完后。一条指令在执行期间，可能产生多次缺页中断。

16. 试说明改进型 Clock 置换算法的基本原理。

基本原理：在将一个页面换出时，如果该页已被修改过，便须将该页重新写回到磁盘上；但如果该页未被修改过，则不必将它写回磁盘上。在改进型算法中，除需考虑页面的使用情况外，还须再增加一个因素，即置换代价，这样，选择页面换出时，既要是未使用过的页面，又要是未被修改过的页面。

17.什么是抖动？引起抖动的原因有哪些？

刚刚换出的页面马上又换入内存，刚刚换入的页面马上又换出内存，这种频繁的调度行为叫做抖动。产生抖动的原因是由于 CPU 的利用率和多道程序度的对立统一矛盾关系引起的，为了提高 CPU 利用率，可提高多道程序度，但单纯提高多道程序度又会造成缺页率的急剧上升，导致 CPU 的利用率下降，而系统的调度程序又会为了提高 CPU 利用率而继续提高多道程序度，形成恶性循环，我们称这时的进程是处于“抖动”状态。

18. 什么是固定分配局部置换，可变分配全局置换，可变分配局部置换？

固定分配局部置换它为每个进程分配一定数量的物理块，在整个运行期间都不改变。若进程在运行中发现缺页，则只能从该进程在内存的页面中选出一个换出，然后再调入需要的页面。实现这种策略难以确定为每个进程应分配的物理块数量：太少会频繁出现缺页中断，太多又会使 CPU 和其他资源利用率下降。

可变分配全局置换。这是最易于实现的物理块分配和置换策略，为系统中的每个进程分配一定数量的物理块，操作系统自身也保持一个空闲物理块队列。当某进程发现缺页时，系统从空闲物理块队列中取出物理块分配给该进程，并将于调入的页装入其中。

可变分配局部置换。它为每个进程分配一定数目的物理块，当某进程发现缺页时，只允许从该进程在内存的页面中选出一页换出，这样就不会影响其他进程的运行。如果进程在运行中频繁的换页，系统需再为该进程分配若干附加物理块，直至该进程缺页率趋于适当程度为止；反之，若一个进程在运行过程中缺页率特别低，则此时可适当减少该进程的物理块。

19.什么是预调页策略，请求调页策略？

预调页策略。根据局部性原理，一次调入若干个相邻的页可能比一次调入一页更高效。但如果调入的一批页面中大厦多数都未被访问，则又是低效的。所以需要采用以预测为基础的预调页策略，将预计在不久之后便会被访问的页面预先调入内存。但目前预调页的成功率仅约 50%。股这种策略主要用于进程的首次调入时，有程序员指出应该先调入哪些页。

请求调页策略。进程在运行中需要访问的页面不在内存而提出的请求，由系统将所需页面调入内存。这种策略调入的页一定会被访问，且这种策略比较易于实现，故在目前的虚拟存储器中大多采用此策略。它的缺点在于每次调入一页，会花费过多的 IO 开销。

20.什么是分段的共享和保护？

分段的共享通过共享段来实现，每一个表项都是共享段的信息，记录了共享此段的每个进程情况。分段的保护越界检查：短号超过段表长度或段内偏移超过段长时越界中断处理。存取控制检查：段表项中存取控制字段规定了对该段的访问方式，比如只读、读写等。环保护机构：一个程序可以访问驻留在相同环或较高特权环中的数据；一个程序可以调用驻留在相同环或较高特权环中的服务。

1.对目录管理的主要要求是什么？

实现“按名存取”；提高对目录的检索速度；文件共享；允许文件重名。

2.什么是按名存取？

按名存取即用户不必考虑文件存储在哪里，怎样组织输入、输出等工作，只要使用文件名，操作系统通过查找目录，就能对存储介质上的信息进行相应的操作。

3.什么是绝对路径和相对路径名和索引节点

绝对路径：从根目录出发的路径。

相对路径：进程对各文件的访问都是相对于当前目录进行的。

索引节点：在检索文件时只用到文件名，也就是说，在检索目录时，文件的其他信息时不会被用到的，也不会被调入内存。因此有些系统采用了文件名和文件描述信息分开的方法，将文件的描述信息单独形成一个索引节点。

磁盘高速缓存是利用内存中的部分存储空间来暂存从磁盘中读出的盘快内容。物理上是驻留在内存的盘快，逻辑上属于磁盘。

4.什么是文件？什么是文件系统？基本操作有哪些？

文件是以计算机硬盘为载体存储在计算机的信息的集合，文件可以是文本文档，图片，程序等等。操作系统负责管理和存储文件信息的软件机构成为文件管理系统，简称文件系统。文件系统由三部分组成：与文件管理有关软件，被管理文件以及实施文件管理所需的数据结构。

最基本的文件操作包括创建文件，删除文件，读文件，写文件，截断文件，文件重定位

文件重定位：按某条件搜索目录，将当前文件位置设为给定值，并且不会读写文件。

截断文件：允许文件所有属性不变，并删除文件内容，即将长度设为 0 并释放空间。

补充：记录是一组相关的数据项集合，用于描述一个对象在某方面的属性，如一个考生报名记录包括考生姓名、出生日期、报考学校代号、身份证号等一系列域。

5. 为什么在大多数 OS 中都引入”打开“这一文件系统调用？打开的含义是什么？

当用户要求对一个文件实施多次读/写或者其他操作时，每次都要从检索目录开始。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。所谓”打开“，是指系统将指名文件的属性（包括该文件在外存上的物理位置），从外存拷贝到内存打开文件表的一个表目，并将该表目的编号（或称索引号）返回给用户。换言之，”打开“，就是在用户和指定文件之间建立起一个连接。此后，用户通过该连接直接得到文件信息，从而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，系统根据用户提供的索引号可以直接在打开文件表中查找到文件信息。这样不仅节省了大量的检索开销，也显著提高了对文件的操作速度。如果用户已不再想要对该文件实施相应的操作，可利用”关闭“系统调用来关闭此文件，即断开此连接，OS 将会把该文件从打开文件表中的表目上删除掉。

6. 什么是文件的逻辑组织和物理组织？文件的逻辑结构有几种形式？

文件的逻辑结构是从用户的观点出发，所观察到的文件组织形式，是用户可以直接处理的数据及其结构，它独立于物理特性。文件的物理结构，又称为文件的存储结构，是指文件在外存上的存储组织形式。这不仅与存储介质的存储性能有关，而且与所在外存的分配方式有关。按逻辑结构，文件有无结构文件和有结构文件两种类型，文件的有结构文件有顺序文件、索引文件、索引顺序文件，散列文件。

7. 什么是 FCB？为什么引入 FCB？

为了能对一个文件进行正确的存取，操作系统必须为文件设置用于描述和控制文件的数据结构，称之为“文件控制块（FCB）”。为实现目录管理，操作系统中引入了文件控制块的数据结构。

8. 简述文件的二级目录组织形式。欲实现文件共享如何处理？

把记录文件的目录分成主文件目录和由其主管的若干个子目录，各子目录的位置由主目录中的一项指出。应用中常设一个主文件目录，而为系统中每一个用户设立一张主文件目录 MFD，每个用户的所有文件均设立一个用户文件目录 UFD，作为 MFD 中的一项。用以描述 UFD 的文件名和物理位置，即 UFD 是用户全部文件的文件控制块的全体。

在二级文件目录中，欲共享文件需给出一个文件的全路径名。由系统从根目录开始检索；或者用户将其当前目录指向另一用户的子目录上，以实现共享访问。

9. 文件目录和目录文件各起什么作用？目前广泛采用的目录结构形式是哪种？它有什么优点？

文件目录记录文件的名称、文件长度、文件存放在外存上的物理地址，以及文件属性和文件建立时间、日期等信息也称之为文件控制块。目录文件是文件系统把同一卷上的若干文件的文件目录组成一个独立的文件，这个全部由文件目录组成的文件称目录文件。文件目录和目录文件是两个不同的概念，文件目录记录文件的管理信息，它用于对单个文件的控制；目录文件是由全部文件目录组成的文件，它用于整个文件系统的管理。目前广泛采用的目录结构是树形目录结构，它的主要优点是：检索效率高，允许文件重名，确切反映了信息的层次结构，并且可以利用层次结构实现文件共享和保护

10. 文件物理结构中的顺序结构、链接结构与索引结构三者之间相比各有什么优缺点？

顺序结构优点：存储管理简单，且容易实现。支持顺序存取和随机存取。顺序存取速度快。所需的磁盘寻道次数和寻道时间最少。缺点不利于文件的动态增长，需要为每个文件预留连续的空间以满足文件动态增长。

链式结构优点是提高了磁盘利用率，不需要为每个文件预留物理块。有利于文件插入和删除。有利于文件动态增长。缺点存取速度慢，不适于随机存取，当物理块间的连接指针出错时，数据丢失。更多的寻道次数和寻道时间。链接指针占用一定的空间，降低了空间利用率。

索引结构优点是不需要为每个文件预留物理块。既能顺序存取，又能随机存取。满足了文件动态增长需要。

缺点较多的寻道次数和寻道时间。索引表本身带来了系统开销。如内存空间，存取时间等

11. 在磁盘上进行一次读写操作需要那几部分时间？其中哪部分时间最长？

寻道时间：磁头移动到指定磁道所需要的时间

延迟时间：磁头定位到某磁道的扇区（块号）所需要的时间

传输时间：从磁盘读出或向磁盘写入数据所需要的时间。

一般来说，寻道时间因为要移动磁臂，所以占用时间最长。

12. FCFS, SSTF, SCAN, C-SCAN 算法的优缺点

FCFS--优点：公平，简单 缺点：平均寻道时间长，仅应用在磁盘 I/O 较少的场合

SSTF—优点：性能比“先来先服务”好，减少了平均寻道时间 缺点：不能保证平均寻道时间最短，可能会出现“饥饿”现象

SCAN—优点：寻道性能较好，可避免“饥饿”现象。缺点：不利于远离磁头一端的访问请求

C-SCAN-优点：消除了两端磁道请求的不公平。缺点：无

13.文件保护？

文件保护通过口令保护、加密保护和访问控制等方式实现。其中，口令保护和加密保护是为了方式用户文件被他人存取或盗取，而访问控制则用于控制用户对文件的访问方式

解决访问控制最常用的方法是根据用户身份进行控制。最普通的方法是给每个文件和目录增加一个访问控制列表，以规定每个用户名及其所允许访问的类型。这种方法的优点是可使用复杂的访问方法。其缺点是长度无法预期并且可能导致复杂的空间管理，使用精简的访问列表可以解决这个问题。

精简的访问列表采用拥有者、组合其他三种用户类型。

- 1) 拥有者：创建文件的用户。
- 2) 组：一组需要共享文件且具有类似访问的用户。
- 3) 其他：系统内的所有其他用户。

1.设备管理的目标和功能是什么？

设备管理的目标：

- (1) 向用户提供外部设备的方便、统一的接口，控制设备工作，完成用户的输入输出请求；
- (2) 充分利用中断技术、通道技术和缓冲技术，提高 CPU 与设备、设备与设备之间的并行工作能力，以充分利用设备资源，提高外部设备的使用效率；
- (3) 设备管理就是要保证在多道程序环境下，当多个进程竞争使用设备时，按照一定的策略分配和管理设备，以使系统能有条不紊地工作。

设备管理的功能：

- (1) 设备分配和回收；
- (2) 管理输入/输出缓冲区；
- (3) 设备驱动，实现物理 I/O 操作；
- (4) 外部设备中断处理；
- (5) 虚拟设备及其实现。

2.什么是块设备？什么是字符设备？

块设备由于信息的存取总是以数据块为单位，所以存储信息的设备称为块设备。它属于有结构设备，如磁盘等。磁盘设备的基本特征是传输速率高，以及可寻址，即对它可随机地读写任意块。

用于数据输入输出的设备为字符设备，因为其传输的基本单位是字符。它属于无结构类型，如打印机等。他们的传输速率低、不可寻址、并且在输入输出时常采用中断驱动方式。

3.程序直接控制方式、I/O 中断方式

(1) 程序直接控制方式。其特点是主机与 I/O 串行工作。CPU 启动 I/O 后，时刻查询 I/O 是否准备好，若设备准备就绪，CPU 便转入处理 I/O 与主机间传送信息的程序；若设备未做好准备，则 CPU 反复查询，“踏步”等待直到 I/O 准备就绪为止。可见这种方式 CPU 效率很低

(2) 程序中断方式。其特点是主机与 I/O 并行工作。CPU 启动 I/O 后，不必时刻查询 I/O 是否准备好，而是继续执行程序，当 I/O 准备就绪时。向 CPU 发中断请求信号，CPU 在适当时候响应 I/O 的中断请求，暂停现程序为 I/O 服务。这种方式消除了“踏步”现象，提高了 CPU 效率

4.什么是通道？与 DMA 方式有什么区别？

通道是专门负责输入/输出的处理机。进一步减少对 CPU 的干预，即把对一个数据块的读或写为单位的干预，减少为对一组数据块的读或写及有关的控制和管理为单位的干预。同时，又可以实现 CPU、通道和 IO 设备三者的并行操作，从而更有效的提高整个系统的资源利用率。

IO 通道和一般处理器的区别是：通道指令的类型单一，没有自己的内存，通道所执行的通道程序释放在主机内存中的，也就是说通道与 CPU 共享内存。

(1) DMA 方式是通过 DMA 控制器控制总线，在设备和主存之间直接实现 I/O 传送。；通道控制方式类似也是以内存为中心实现设备与主存直接交换数据的控制方式。

(2) 通道控制方式通过执行通道程序进行 I/O 操作的管理。

(3) 与 DMA 控制方式相比通道控制方式所需的 CPU 干预更少，而且 DMA 控制器通常只控制一台或多台同类的高速设备；而通道可控制多台同类或不同类的设备。

5.什么是 DMA 方式？它与中断控制方式的主要区别是什么？

DMA（直接存储器存取）是一种不经过 CPU 而直接从内存存取数据的数据交换模式。特点是传输的基本单位是数据块，所传送的数据，是从设备直接送入内存，或者相反。仅在传送一个或多个数据块的开始和结束时，才需 CPU 干预，整块数据的传送是在 DMA 控制器的控制下完成的。

中断控制方式在每个数据传送完成后中断 CPU，而 DMA 控制方式则是在所要求传送的一批数据全部传送结束时中断 CPU。中断控制方式的数据传送在中断处理时由 CPU 控制完成，而 DMA 控制方式则是在 DMA 控制器的控制下完成。不过，在 DMA 控制方式中，数据传送的方向，存放数据的内存始址及传送数据的长度等仍然由 CPU 控制。中断控制方式以 CPU 为核心，DMA 以存储器为核心。因此 DMA 方式能与 CPU 并行工作。DMA 方式传输批量的数据，中断控制方式传输则以字节为单位。

6. 设备驱动程序是什么？为什么要有设备驱动程序？写出设备驱动程序的处理过程？

设备驱动程序与硬件直接相关，负责具体实现系统对设备发出的操作命令，驱动 I/O 设备工作的驱动程序。是 I/O 进程与设备控制器之间的通信程序，常以进程的形式存在。

设备驱动程序是控制设备动作的核心模块，向上层用户程序提供一组接口，设备的具体区别被设备驱动程序封装，且处理用户进程发出的 I/O 请求，如 read 和 write 操作。

用户进程使用设备驱动程序时，设备驱动程序的处理过程为：将抽象的 I/O 请求转换为具体的请求，检查 I/O 请求的合法性，读出和检查设备的状态，传送必要的参数，设置设备工作方式，启动 I/O 设备。

7.什么是设备独立性？为什么引入？如何实现？

设备独立性是指应用程序独立于具体使用的物理设备。为了提高设备分配时的灵活性和设备的利用率、易于实现 IO 重定向，因此引入设备独立性。为了实现设备独立性，应引入逻辑设备和物理设备概念。在应用程序中使用逻辑设备名来请求使用某类设备，系统执行时是使用物理设备名。为了实现设备独立性，必须在驱动程序上设置一层设备独立性软件，还需在系统中设置一张逻辑设备表（LUT），用于将逻辑设备名映射为物理设备名。LUT 表项包括逻辑设备名、物理设备名和设备驱动程序入口地址；

8.引入缓冲技术（缓冲区）的主要目的？有哪几种？

缓和 CPU 与 I/O 设备速度不匹配的矛盾；减少对 CPU 的中断频率；提高 CPU 与 I/O 设备之间的并行性。根据系统设置缓冲区的个数分为单缓冲，双缓冲，循环缓冲以及缓冲池。

单：在设备和处理机之间设置一个缓冲区。设备和处理机交换数据时，先把被交换数据写入缓冲区，然后需要数据的设备或处理机从缓冲区取走数据。

双：双缓冲区机制又称缓冲对换。I/O 设备输入数据时先输入到缓冲区 1，直到缓冲区 1 满后才输入到缓冲区 2，此时操作系统可以从缓冲区 1 中取出数据放入用户进程处理，并由 CPU 计算。双缓冲的使用提高了处理机和输入设备的并行操作的程度。

循环：包含多个大小相等的缓冲区，每个缓冲区中有一个链接指针指向下一个缓冲区，最后一个缓冲区指针指向第一个缓冲区，多个缓冲区构成一个环形。

缓冲池：由多个系统共用的缓冲区组成，缓冲区按其使用状况可以形成三个队列：空缓冲队列、装满输入数据的缓冲队列（输入队列）和装满输出数据的缓冲队列（输出队列）。还应具有四种缓冲区：用于收容输入数据的工作缓冲区、用于提取输入数据的工作缓冲区、用于收容输出数据的工作缓冲区、用于提取输出数据的工作缓冲区。

9.设备分配的总原则是什么？设备分配时应考虑的因素有哪些？分配方式有哪两种？

既要充分发挥设备的使用效率，又要避免造成进程死锁，还要将用户程序和具体设备隔离开。

（1）I/O 设备的固有属性（2）I/O 设备的分配算法（3）I/O 设备分配的安全性（4）I/O 设备的独立性

静态分配：主要用于独占设备的分配，它在用户作业开始执行前，由系统一次性分配该作业所要求的全部设备，控制器（如通道等）。一旦分配后，就一直为该作业所占有，直到作业被撤销。静态分配虽然不会出现死锁，但设备的使用效率低。

动态分配：是在进程执行过程根据执行需要进行。当进程需要设备时，通过系统调用命令向系统提出设备请求，由系统按照事先规定的策略给进程分配所需要的设备，I/O 控制器，一旦用完后，便立即释放。有利于提高设备的利用率。可能会进程死锁。

10.用于设备分配的数据结构有哪些？他们之间的关系是什么？

用于设备分配的数据结构有系统设备表(SDT)、设备控制表(DCT)、控制器控制表(COCT)和通道控制表(CHCT)。SDT 整个系统中只有一张记录系统中全部设备的情况，是系统范围的数据结构。每个设备有一张 DCT，系统为每一个设备配置一张 DCT 以记录本设备的情况。每个控制器有一张 COCT，系统为每一个控制器都设置一张用于记录本控制器情况的 COCT。系统为每个通道配置一张 CHCT，以记录通道情况。SDT 中有一个 DCT 指针，DCT 中有一个 COCT 指针，COCT 中有一个 CHCT 指针。

11.为什么引入 SPooling 技术？什么是 SPooling 技术？什么是 SPooling 系统？其系统由什么组成？它的功能与特点是什么？

答：为了缓和 CPU 的高速性和 I/O 设备低速性之间的矛盾。SPooling 技术是是一种外围设备同时联机操作技术，是操作系统采用的一种将独占设备改造为共享设备的技术。它是关于慢速字符设备如何与计算机主机交换信息的一种技术，又亦称为假脱机操作。

SPooling 系统是指在通道技术和中断技术的支持下，在主机的控制之下，完成 I/O 的软件系统。其系统组成：输入井和输出井；输入缓冲区和输出缓冲区；输入进程和输出进程在磁盘上的两个存储空间，输入井模拟脱机输入时的磁盘，暂存 I/O 设备输入的数据，输出井模拟脱机输出时的磁盘，暂存用户程序输出的数据。

输入缓冲区用于暂存由输入设备送来的数据，以后在传送到输入出缓冲区用于暂存从输出井送来的数据，以后再传送给输出设备。

输入进程模拟脱机输入时的外围控制机；输出进程模拟脱机输出时的外围控制机

功能与特点：提高了 I/O 的速度；将独占设备改造为共享设备；实现了虚拟设备功能。

12. SPooling 技术如何使一台打印机虚拟成多台打印机？优点是？（对用户）

（1）系统对于用户的打印输出，并不真正把打印机分配给该用户进程，而是由输出进程在输出井中申请一个空闲磁盘块，并将要打印的数据送入其中。

（2）输出进程再为用户进程申请一张空白的用户请求打印表，并将用户的打印要求填入，再将该表挂到请求打印队列上。

(3) 若打印机空闲, 输出程序从请求打印队列队首取表, 将要打印的数据从输出井传送到内存缓冲区, 再进行打印, 直到打印队列为空。

优点是在多用户情况下, 每一个用户使用打印机就好像自己拥有一台打印机。不会产生打印机“忙”而等待。

13. 什么是独享设备? 共享设备? 设备分配技术? 设备管理的主要功能?

独享设备: 即不能共享的设备, 一段时间只能由一个作业独占。如打印机。

共享设备: 可由若干作业同时共享的设备, 如磁盘机等

设备分配技术主要有: 独占分配、共享分配和虚拟分配。独占分配适用于独占设备, 系统效率低; 共享分配适用于高速、大容量直接存储的共享设备, 设备的利用率较高; 虚拟分配技术利用共享设备去实现独占设备的功能, 从而使独占设备“感觉上”成为可共享的、快速的 I/O 设备。设备管理的主要功能包括 实现外围设备的分配与回收、实现虚拟设备和 实现对磁盘的驱动调度。

数据结构

1. 逻辑结构和物理结构

逻辑结构是元素之间的逻辑关系。它与数据的存储无关, 是独立于计算机的。数据的逻辑结构分为线性结构和非线性结构, 线性表是典型的线性结构。集合, 树, 图是典型的非线性结构。

存储结构是指数据结构在计算机中的表示, 也称物理结构。它包括数据元素的表示和关系的表示。数据的存储结构是逻辑结构用计算机语言的实现, 它依赖于计算机语言。数据的存储结构主要有: 顺序存储, 链式存储, 索引存储和散列存储。

顺序存储: 把逻辑上相邻的元素存储在物理位置上也相邻的存储单元里, 元素之间的关系由存储单元的邻接关系来体现。优点: 可以随机存取, 每个元素占用最少的存储空间。缺点: 只能使用相邻的一整块存储单元, 可能产生较多的外部碎片。

链接存储: 不要求逻辑上相邻的单元在物理上也相邻。借助指示元素存储地址的指针表示元素之间的逻辑关系。优点: 不会出现碎片现象, 充分利用所有存储单元。缺点: 每个元素因存储指针而占用额外的存储空间, 并且只能实现顺序存取。

索引存储: 在存储信息的同时, 还建立其附加的索引表。索引表的每一项成为索引项, 索引项的一般形式是: (关键词, 地址)。其优点是检索速度快, 缺点是增加了附加的索引表, 占用较多的存储空间。在删除和增加数据时要修改索引表, 会花费较多的时间。

散列存储: 根据元素的关键字直接计算出该元素的存储地址, 也称 HASH 存储。优点是检索, 增加, 删除节点的速度都很快。缺点: 如果散列函数不好可能会出现元素存储单元的冲突, 解决冲突需要增加时间和空间开销

数据的运算指施加在数据上的运算包括运算的定义和实现。运算的定义是针对逻辑结构: 指出运算的功能, 运算的实现是针对存储结构: 指出运算的具体步骤。

2. 什么是算法? 有几个特性? 目标是什么?

算法是对特定问题求解步骤的一种描述。

1. 有穷性: 一个算法必须总是在执行有穷步之后结束, 且每一步都在有穷的时间内完成。
2. 确定性: 算法中每一条指令必须有确切的含义, 读者理解时不会产生二义性。对于相同的输入只能产生相同的输出。
3. 可行性: 一个算法是可行的。算法中描述的操作都是可以通过已经实现的基本运算执行有限次实现。
4. 输入: 一个算法有零个或多个输入。输入取自于某个特定的对象的集合。
5. 输出: 一个算法有一个或多个输出。输出是同输入有着某种特定关系的量。

好的算法应该考虑如下目标 1.正确性 2.可读性 3.健壮性：对输入非法数据，也要能做出适当反应或进行处理。4.效率于低存储量需求