**1.OS的作用是什么？**

1.操作系统作为用户与硬件系统之间的接口。

2.操作系统作为资源的管理者。

3.操作系统实现了对资源的抽象。

**2.单道批处理是什么概念，解决了什么问题，是如何解决的？还存在什么问题？**

1.单道批处理是计算机系统对一批作业（但在内存中仅有一道作业）自动进行处理。

2.解决了人机矛盾和CPU与I/O设备速度不匹配矛盾。

3.通过提高系统资源的利用率和系统吞吐量。

4.不能充分利用系统资源。

**3.多道批处理是什么概念，解决了什么问题，是如何解决的？**

1.在内存中同时存放多道相互独立程序，按照一定的算法，共享CPU和系统中的各种资源。

2.提高了资源的利用率和系统吞吐量。

3.它能使多道程序交替运行，使得CPU和其他资源保持忙碌状态

**4.实现多道程序设计应解决那些问题？**

1.处理机管理问题

2.内存管理问题

3.I/O设备管理问题

4.文件管理问题

5.作业管理问题

**5.对比分时系统和实时系统的特征，它们的含义有何异同？**

分时系统是指在一个系统中多个用户分时地使用同一台计算机；

实时系统是指计算机及时响应外部事件地请求并在规定时限内完成对该事件地处理，控制所有实时外设和实时任务协调一致地运行。

共同点：两者同时具有多路性 交互性 独立性及时性四个特征

区别：实时系统额外具有可靠性分时系统没有实时系统具有的容错机制可靠

**6.为什么引入进程的概念？**

这是因为并发执行的程序（即进程）是“停停走走”地执行，只有在为它创建进程后，在它停下时，方能将其现场信息保存在它的PCB中，待下次被调度执行时，再从PCB中恢复CPU现场并继续执行，而传统的程序却无法满足上述要求。

**7.如何理解并发性和共享性的联系？**

并发性和共享性互为存在条件，是操作系统最基本的特征。

一方面资源共享是以程序（进程）的并发性执行为条件的，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，则也必将影响到程序并发执行。如果自己用的操作系统，不能并发执行，你每次只能打开一个程序，要打开另外一个就得关闭前一个，会十分痛苦。而你打开多个程序的时候不可避免会用到同一个资源。

**8.什么是同步和异步的概念？如何理解操作系统的异步性？**

异步：在多批道处理系统中，由于并发进行的进程中，存在调度的随机性和执行速度的独立性，即间接制约，导致各个进程走走停停

同步：多个进程之间存在直接制约关系，而为了提高执行速度，让各个进程之间互相发送消息，确定执行顺序，而不会出现走走停停的情况，达到同步。同步不是指多个进程同时执行，而是指多个进程之间互相发送消息，不会出现走走停停的混乱情况，将无序变为有序。

**9.为什么说操作系统是对裸机的首次虚拟？**

计算机的硬件、软件以及软件的各部分之间是一种层次结构的关系。硬件在最底层，操作系统是裸机上的第一层软件，是对硬件功能的首次扩充。

**10.为什么说处理机管理体现为进程管理？**

因为处理机是计算机中宝贵的资源，有效地将处理机分配和回收给各个进程体现才能系统性能。

**11.存储器管理有哪些功能？**

内存分配和回收，内存保护，地址映射和内存扩充

**12.设备管理有哪些功能？**

缓冲管理、设备无关性、设备分配、设备处理、虚拟设备功能

**13.如何划分程序模块？对模块有什么要求？**

划分：功能相对独立的一段代码

要求：高内聚，低耦合

**14.简述冯.诺依曼计算机的工作原理**

存储程序加程序控制；

“存储程序控制”原理的基本内容：

采用二进制形式表示数据和指令。

将程序（数据和指令序列）预先存放在主存储器中（程序存储），使计算机在工作时能够自动高速地从存储器中取出指令，并加以执行（程序控制）。

由运算器、控制器、存储器、输入设备、输出设备五大基本部件组成计算机硬件体系结构。

结构计算机的工作原理可以概括为八个字：存储程序，程序控制

**15.什么是中断？为什么要引入中断机制？**

中断是系统运行期间发生任何紧急事件，使得CPU暂停正在执行的程序，保留现场后自动转去执行相应事件的处理程序。

引入中断机制的目的有：

1、使实时处理紧急事件成为可能

2、可增加处理机的执行效率

3、可简化OS的程序设计

**16.CPU如何感知中断请求，又如何响应中断请求？**

中断源向CPU发出中断请求，CPU收到中断请求后保留现场信息转相应的事件处理程序去处理中断源，处理完后返回现场信息，继续处理之前没有完成的任务。

**17.中断处理过程是怎样的？**

1、保留现场到各个寄存器内

2、处理触发中断源的中断事件，即真正执行具体的为某个中断源服务的中断服务程序

3、关闭中断，以防止在恢复现场过程中被新的中断请求打断，接着恢复现场，然后开放中断，以便返回原来的程序后可响应其他的中断请求

**18.CPU的现场信息主要包括哪些内容？**

CPU各个寄存器（通用寄存器和指令寄存器）、程序状态字PSW以及用户栈指针

**19.什么是程序执行结果的不可再现性？如何确保执行结果的可再现？**

不可再现性：由于程序的并发执行，打破了由另一程序独占系统资源的封闭性，因而破坏了可再现性

解决：必须通过Berstein条件的限制，保证程序在异步环境下的执行结果是确定的

**20.为什么要引入进程的概念？进程有哪些特征？**

为了实现多道程序并发执行，而进程是进程实体的运行过程，资源分配和调度的独立单位。

特征：动态性，并发性，独立性，异步性, 结构性

**21.进程图与前趋图的作用分别是什么？**

进程图：描述进程家族关系，一个结点代表一个进程，一条孤代表进程间的父子关系。

前趋图：表示语句或者进程间的执行顺序。

**22.引起进程阻塞和唤醒的典型原因有哪些？**

1、请求系统服务

＊无法获得服务，进程主动阻塞

＊服务完成，由服务释放进程唤醒阻塞进程

2、启动某种操作

＊进程主动阻塞，等待操作完成

＊操作完成，由中断处理程序唤醒阻塞进程

3、合作数据尚未到达

＊合作进程的数据尚未到达，等待进程阻塞

＊新数据到达，由合作进程唤醒阻塞进程。

4、无新工作可做

＊系统进程无新工作可作，主动阻塞

＊新工作到达时，系统进程被唤醒

**23.创建进程需要执行哪些操作？**

1.申请空白PCB

2.为新进程分配资源

3.初始化PCB

4.将PCB插入队列

**24.进程的同步机制解决什么问题？**

解决进程在并发执行时产生结果的不可再现性（不确定性），也可以说进程的异步性。

其实现了多个相关进程在执行次序上的协调。

**25.什么是临界区，引入临界区的目的是什么？**

临界区：每个进程中访问临界资源的那段程序称之为临界区。

目的：为了让诸进程互斥，实现对临界资源的互斥访问

**26.利用信号量机制如何实现资源的互斥使用？**

进入临界区前，进程通过锁变量判断临界资源是否被占用

占用：进程反复测试W的状态

否则：为临界资源加锁，然后进入临界区，为临界资源开锁

**27.互斥信号量机制如何体现资源的互斥准则？**

空闲让进，忙则等待，有限等待，让权等待

**28.信号机机制在有合作关系的进程中如何描述进程同步？**

先操作的进程执行后设置V原语，在后操作的进程执行前设置P原语，这一对原语同时存在

**29.讨论生产者-消费者问题中互斥信号量设置的必要性。**

从生产者角度看，如果多个生产者进入缓冲区进行各自的操作，可能会出现覆盖等错误；

从消费者角度看，同样可能出现错误；

单纯从生产者消费者的角度来看，一个生产者和一个消费者进入缓冲区，修改的是各自的缓冲区，没有冲突，可以不必设置互斥信号量。

**30.哲学家就餐问题可以有哪些方法避免死锁**

1.最多允许4个哲学家同时坐在桌子旁边

2.仅当一个哲学家左右两边的筷子都可以用时才允许拿筷子

3．给所有哲学家编号，奇数号的哲学家必须首先拿左边的筷子，偶数号的哲学家则反之，最后总有一个哲学家能获得两只筷子而进餐

**31.读者-写者问题是如何实现读者优先或写者优先的？**

读者优先：第一个读者可以读，只有第一个读者阻塞写者，只有最后一个读者才能唤醒写者。

写者优先：第一个写者可以读，只有第一个写者阻塞读者，只有最后一个写者才能唤醒读者。

**32.直接消息通信机制中，接收进程的PCB需要保存哪些与通信有关的数据？**

1. mq (消息队列队首指针)

2.mutex (消息队列互斥信号量)

3.sm (消息队列资源信号量)

**33.管道通信的工作原理和基本要求是什么？**

工作原理：

＊管道是连接发送进程和接收进程的共享文件（Pipe 文件）

＊发送进程以字符流的形式将消息写入管道

＊接收进程按先进先出的顺序从管道读出数据

基本要求：

＊对管道的互斥使用

＊发送进程与接收进程的同步

＊确定通信对方的存在

**34.讨论管道通信的代码描述**

创建管道文件

建立通信然后实现连接

信息交换后由任意一方断开连接

**35.引入线程的目的是什么？**

增加并发程度和减少并发开销

**36.线程与进程有什么区别与联系？**

1.调度的基本单位:线程是调度和分派的基本单位，而在未设置线程时，进程也是进行资源分配和调度执行的独立单位

2.并发性:进程之间、线程之间均可并发执行

3.拥有资源:进程拥有资源,线程继承资源

4.独立性:同一进程中的线程的独立性低于不同进程的线程独立性

5.系统开销:线程的系统开销小于进程

6.支持多处理机系统:可将一个进程中的多个线程分配到多个处理机中

**37.调度有哪几个层次，分别作用于什么范围？**

**1. 高级调度（作业调度）**

调度对象：作业。

功能：根据某种算法，决定将外存上处于后备队列中那几个作业调入内存，为它们创建进程，分配必要的资源，并将它们放入就绪队列。

主要用于多道批处理系统，而在分时和实时系统中不设置高级调度。

频率很低，周期很长，大于几分钟一次。

**2. 低级调度（进程调度）**

调度对象：进程。

功能：根据某种算法，决定就绪队列中的那个进程获得处理机。并由分派程序将处理机分派给选择的进程。

最基本的调度，在多道批处理，实时和分时三种类型的OS中，都必须配置进程调度。

频率很高，周期很短，在分时系统中大概仅10~100nm.

**3. 中级调度（内存调度）**

目的：提高内存利用率和系统吞吐量。

作用：就是将暂时不能运行的进程，调至外存等待(挂起转台)，和将外存上已经满足条件的就绪进程在调入内存中。

频率和周期：处于作业调度和进程调度之间。

**38.引起进程调度的时机和原因有哪些？**

1、当前运行进程运行结束。因任务完成而正常结束，或者因出现错误而异常结束。

2、当前运行进程因某种原因，比如 I/O 请求、 P 操作、阻塞原语等，从运行状态进入阻塞状态。

3、执行完系统调用等系统程序后返回用户进程，这时可以看作系统进程执行完毕，从而可以调度一个新的用户进程。

4、在采用抢占调度方式的系统中，一个具有更高优先级的进程要求使用处理器，则使当前运行进程进入就绪队列（与调度方式有关）。

5、在分时系统中，分配给该进程的时间片已用完（与系统类型有关）。

**39.作业与进程有什么区别与联系？**

区别：进程是一个程序的一次执行，而作业是任务。

联系：一个作业通常包括多个进程，多个进程共同完成一个作业。一个是任务的静态描述，一个是任务的动态描述。互相补充

**40.作业有哪几种状态？**

后备状态、 提交状态 、执行状态和完成状态

**41.作业调度的时机与任务是什么？**

时机：内存中的进程数小于多道度。任务：从后备队列中选取多少个作业调入内存，取决于多道度，而接纳哪些作业取决于调度算法。

**42.进程调度有哪两种调度方式？各自的调度时机是什么？**

1.非抢占式：

发生调度的时机：正在运行的进程运行正常/异常结束；进程发生阻塞；

2.抢占式

发生调度的时机：有优先权更高的进程到达；有更短的进程到达；时间片用完了。

43.进程的优先权有静态优先权和动态优先权两种形式，试分析基于这两种优先权的非抢占式调度和抢占式调度的**调度时机**和**调度原则**。

非抢占式调度方式的调度时机：CPU空闲

抢占式调度方式的调度时机：CPU空闲或新进程到达

基于优先权的抢占式调度：

新进程到达时：

静态优先权：新老进程比较优先级，动态优先权：重新计算新老进程、就绪进程优先级

CPU 空闲时：

静态优先权：选择就绪队列优先权最高者

动态优先权：重新计算新进程、就绪进程优先级

**44.哪种调度方式既考虑等待时间又兼顾服务时间？**

高响应比优先调度算法

**45.为什么说多级反馈队列调度算法是综合性能比较好的？**

1.对于终端型用户，作业比较小，可在第一队列即可完成，响应时间很短。

2.对于短批处理作业用户，这类作业属于稍长的短作业，可在第二或第三队列即可完成，周转时间也比较短。

3.对于长批处理作业用户，它总会被运行的，不用担心它长期得不到处理。

**47.什么是死锁？引起死锁的原因是什么？**

两个或多个事务在同一资源上相互占用，并请求锁定对方的资源，从而导致恶性循环的现象。

当多个进程因竞争资源而造成的一种僵局（互相等待），若无外力作用，这些进程都将无法向前推进，这种情况就是死锁。

引起死锁的原因有：竞争不可抢占性资源、竞争可消耗性资源、进程推进顺序不当。

**48.产生死锁的必要条件是什么？**

1、互斥条件：进程对占有的资源进行排他性使用

2、请求和保持条件：动态分配策略中，进程占有资源却又申请新的资源

3、不可抢占条件：对己经分配给进程的资源不可抢占使用

4、环路等待条件：发生死锁时，系统的RAG必然出现环路

**49.预防死锁的策略有哪几个，分别破坏什么条件？**

1.破坏“请求和保持条件”

2.破坏“不可抢占”条件

3.破坏“环路等待”条件

**50.简述银行家算法的工作流程**

1、检测请求资源数量是否合理：不合理拒绝请求并报错，合理继续；

2、检测系统是否有足够可用资源分配；没有等待，有继续；

3、利用相应的数据结构进行试探性分配；

4、利用安全性算法，检测此次分配后系统是否处于安全状态；是则分配资源，否则恢复数据结构。

**51.如何检测申请不同类资源的进程是否存在死锁？**

1.在RAG中找一个非孤立，非阻塞的进程结点，去掉所有边将其变成孤立结点

2.把等待该资源的请求边变成分配边

3.如果所有进程结点均成为孤立结点，则资源分配图是可以完全化简的

4.若可以完全化简资源分配图，则不存在死锁，否则，系统中存在死锁

**52.程序要运行必须经过哪几个步骤，各自完成什么任务？**

1.编译：由编译程序对源程序编译得到0,1代码的目标模块

2.链接：由链接程序将编译后形成的一组目标模块以及库函数链接形成可执行的装入模块

3.装入：由装入程序将装入模块装入内存

**53.地址映射有静态映射和动态映射，试比较两者的优劣**

静态映射灵活性低，但对硬件要求也低；动态映射灵活性高，速度快，但是对硬件要求高

**54.用界限寄存器如何实现内存保护？**

每当CPU要访问内存时，硬件自动将被访问的内存地址与界限寄存器的内容进行比较，以判断是否越界，如果没有越界，则按此地址访问内存，否则将产生越界中断。

**55.固定分区和动态分区分配初始的分区是如何得到的？**

固定分区：分区说明表。

动态分区：空闲分区表，空闲链。

**56.引入对换技术的目的是什么？对换有哪两种类型？**

使进程在内存和外存之间能够动态调度。

有整体对换和页面对换两种类型。

**57.分页存储管理方式是如何实现的？**

系统将一个进程的逻辑地址空间分成若干个大小相等的页面，相应地，将内存空间分成若干个与页面同样大小的物理块，内存的分配以块为单位，并允许将一个进程的若干页分别装入到多个可以不相邻接的物理块中。

逻辑地址分为两部分：页号和页内地址。在程序运行时，为了能在内存中找到每个页面对应的物理块，系统为每个进程建立了一张页表。进程的每个页占页表的一个表项，其中记录了相应页对应的内存块的块号，以及用于分页保护的存取控制信息。

页式存储管理系统中，逻辑地址到物理地址的转换是在进程执行的过程中，由硬件地址变换机构借助页表自动进行的。

**58.分页存储管理方式中的数据结构有哪些，分别有什么作用？**

面向进程：页面请求表，整个系统统一一张页面请求表，记录所有进程使用内存的请况。

面向内存：内存块表（空闲块表，空闲块链，位示图）。

进程和内存的对应：页表，记录每个页面在内存所对应的物理块。

**59.分页方式中如何将逻辑地址转换为物理地址？**

当进程要访问某个逻辑地址中的数据时，分页地址变换机构会自动地将有效地址（相对地址）分为页号和页内地址两部分再以页号为索引去检索页表。查找操作由硬件执行。

在执行检索前，先将页号与页表进行比较，如果页号大于或等于页表长度，则表示本次所访问的地址已超越进程的地址空间。

于是，这一错误将被系统发现，并产生一地址越界中断。若未出现越界错误，则将页表始址与页号和页表项长度的乘积相加，变得到该表项在页表的位置，于是可从得到该页的物理块号，将之装入物理地址寄存器中，与此同时，在将有效地址寄存器中的页内地址送入物理地址寄存器的块内地址字段中。

**60.引入分段存储管理方式有什么好处？**

1.方便编程

2.分段便于实现信息共享，信息保护

3.而且分段可以动态增长，动态链接

**61.段页式存储管理方式的地址变换中，三次访问内存的目的分别是什么？**

第一次：获取页表地址

第二次：获取指令或数据的物理地址

第三次：从所得地址中取出指令或数据

**62.什么是虚拟存储器？虚存的容量如何衡量？**

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。

其逻辑容量由内存容量和外存容量之和所决定。

**63.阐述实现虚拟存储器的理论依据。**

1.基于局部性原理，应用程序在运行之前并不必全部装入内存，仅须将当前要运行的那部分程序和数据装入内存便可启动程序的运行，其余部分仍驻留在外存上；当要执行的指令或访问的数据不在内存时，再由OS请求调入；如果内存已满，则通过置换功能进行程序或数据之间的置换。

2.虚拟存储器还必须建立在离散分配的基础上，其实现方式可分成请求分页、请求分段和请求分段分页等方式。

**64.虚拟存储器具有哪些特征？**

多次性 对换性 虚拟性

**65.请求分页存储管理方式的页表中各字段有什么作用？**

1.页号和内存块号：当页面在内存时，用于地址变换

2.中断位：表示该页是在内存还是在外存

3.外存地址：如果页面在外存，记录磁盘块号

4.访问位：记录该页最近被访问的次数或多久未被访问，根据访问位来决定淘汰哪页

5.修改位：查看此页是否在内存中被修改过

**66.缺页率与哪些因素有关？**

1.页面大小

2.进程所分配物理块的数目

3.页面置换算法

4.程序固有特性

**67.为什么FIFO置换算法存在Belady异常现象？**

因为驻留在内存最久的页面不代表它的访问频率最小，可能是经常访问或者是刚刚访问过，如果把这样的页面换出去，在未来可能会频繁地换进来，导致命中率低下。

**68.LRU置换算法的思想是什么，如何实现页面的“计时”？**

LRU算法是选择最近最久未使用的页面予以淘汰。利用寄存器实现页面的“计时”。为每个在内存中的页面配置一个移位寄存器。

当进程访问某物理块时，将相应寄存器的Rn-1位置成1。此时，定时信号将每隔一定时间将寄存器右移一位。

如果我们把n位寄存器的数看作是一个整数，那么，具有最小数值的寄存器所对应的页面，就是最近最久未使用的页面。

**69.简述改进的Clock算法的执行过程。**

先为每一个页面设置一个访问位，再将所有的页面通过指针链接起来形成一个循环队列，当页面没有被访问时，访问位为1；

当需要换出页面时，替换指针从原来的位置开始，每次判断所指的页面的访问位是否为0；

若为0则换出，若为1,则设置为0，寻找下一个页面，若到队尾则重新从对首开始寻找

**70.产生抖动的原因是什么？**

同时在系统中运行的进程太多，由此分配给每一个进程的物理块太少，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，频繁地出现缺页，必须请求系统将所缺之页调入内存。

**71.为什么引入工作集机制可以预防抖动？**

在进程初始分配内存块时，分配大于等于工作集的内存块数，使其在比较低的缺页率下开始工作。

**72.请求分段存储管理方式的地址越界中断如何处理？**

处理地址越界中断时，首先判断该段的扩充位，如果可扩充，则增加段长，否则发生越界中断，进行出错处理

**73.如何实现分段共享？**

配置一张共享段表，所有各共享段表中记录共享分段的进程数，表中包括该段的各种信息，以及各进程利用该共享段的信息。

**74.I/O系统的软件有哪几个层次？**

1.用户层I/O软件

2.设备独立性软件

3.设备驱动程序

4.中断处理程序

**75.简述字符设备控制器的组成和原理。**

组成：寄存器、I/O逻辑、接口。

**76.引入通道的目的是什么？**

为了建立独立的I/O操作，不仅使数据的传送能独立于CPU，而且也希望有关对I/O操作的组织、管理及其结束处理尽量独立，以保证CPU有更多的时间去进行数据处理。

**77.操作系统是如何识别中断请求，如何调用中断处理程序的？**

每个中断源用一固定触发器寄存中断信号，称为中断位——值为1表示有中断信号，为0则无。

当某一中断源需要CPU为其进行中断服务时，就输出中断请求信号，使中断请求触发器置位，向CPU请求中断。

每一种中断都有一个与之相联系的中断号，并有一个与之相联系的中断处理程序，而每种中断的中断处理程序按中断号的顺序存放在一张中断向量表中。

在中断响应时，系统会根据中断号去查找中断向量表，从中获得相应的中断处理程序的入口地址，这样便可以转入中断处理程序执行。

**78.中断和陷阱有什么区别？**

1.陷阱由处理机正在执行的指令引起，而中断由与现行指令无关的中断源引起

2.陷阱处理程序提供的服务为当前进程所用，而中断处理程序提供的服务不是为当前进程

3.CPU可以在指令执行中响应陷阱，而必须在指令执行后响应中断

**设备驱动程序的主要功能是什么？**

1.接收由与设备无关的软件发来的命令和参数，并将命令中的抽象要求转换为与设备相关的低层操作序列；

2.检查用户I/O请求的合法性，了解I/O设备的工作状态，传递与I/O设备操作有关的参数，设置设备的工作方式；

3.发出I/O命令，如果设备空闲，便立即启动I/O设备，完成指定的I/O操作；如果设备忙碌，则将请求者的请求块挂在设备队列上等待；

4.及时响应由设备控制器发来的中断请求，并根据其中断类型，调用相应的中断处理程序进行处理。

**79.中断驱动的I/O控制方式与DMA的I/O控制方式有什么区别？**

1.中断方式是以字符为单位传送数据，而DMA方式是以块为单位传送数据

2.中断方式是在DR满后发中断信号，而DMA方式是在数据块传送完成后发中断信号

3.中断方式的数据传送是在CPU的控制下由中断处理程序完成的，而DMA方式的数据传送是在控制器的控制下盗用CPU周期完成的

**80.DMA控制方式中，进入控制器DR中的数据如何传送到内存？**

不断挪用CPU周期，将DR中的数据写入内存指定单元，直至DC=0

**81.通道控制方式中什么情况下需要盗取CPU周期**

1.通道获取下一条通道指令

2.数据控制器将数据传到内存

**82.操作系统如何实现逻辑设备名向物理设备名的映射？**

在系统中配置一张逻辑设备表，每个表目包含三项——逻辑设备名、物理设备名和设备驱动程序的入口地址。

当进程用逻辑设备名请求分配I/O设备时，系统根据当时的具体情况，为它分配一台相应的物理设备。

与此同时，在逻辑设备表上建立一个表目，填上应用程序中使用的逻辑设备名和系统分配的物理设备名，以及该设备驱动程序的入口地址。

当以后进程再利用该逻辑设备名请求I/O操作时，系统通过查找LUT，便可找到该逻辑设备所对应的物理设备和该设备的驱动程序

**83.独占设备的分配流程是怎样的？**

1.分配设备

2.分配控制器

3.分配通道

**84.用户层实现的SPOOLing技术其原理和目的是什么？**

目的是实现虚拟设备功能

**85.引入缓冲区的目的是什么？**

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CPU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设备间的并行性

**86.简述缓冲池的基本组成和工作过程。**

缓冲池由多个缓冲区组成，其中的缓冲区可供多个进程共享，且既能用于输入又能用于输出。

当输入进程需要输入数据时，便用空缓冲队列的队首摘下一个空缓冲区，把它作为收容输入工作缓冲区，

然后把数据输入其中，装满后再将它挂到输入队列队尾，当计算进程需要输入数据时，便从输入队列取得一个缓冲区作为提取输入工作缓冲区，计算进程从中提取数据，数据用完后再将它挂到空缓冲队列尾。

当计算进程需要输出数据时，便从空缓冲队列的队首取得一个空缓冲区，作为收容输出工作缓冲，其中装满输出数据后，再将它挂到输出队列尾，

当要输出时，由输出进程从输出队列中取得一个装满输出数据的缓冲区，作为提取输出工作缓冲区，当数据提取完后，再将它挂到空缓冲区队列的末尾。

87.空闲缓冲队列 emq: 由空缓冲区所链成的队列

输入队列 inq:由装满输入数据的缓冲区所链成的队列

输出队列 outq:由装满输出数据的缓冲区所链成的队列

四个工作缓冲区

输入收容hin

输出收容hout

输入提取sin

输出提取sout

**88.简述磁盘的格式和物理扇区号的结构**

（1）硬盘有数个盘片，每盘片两个面，每个面一个磁头

（2）盘片被划分为多个扇形区域即扇区

（3）同一盘片不同半径的同心圆为磁道

（4）不同盘片相同半径构成的圆柱面即柱面

**89.逻辑扇区号与物理扇区号如何相互转换？**

通过逻辑扇区号计算 柱面号->磁道号->扇区号

**90.磁盘访问时间如何计算**

寻道时间+平均旋转延迟时间+传输时间

**91.对比FCFS、SSTF、SCAN磁盘调度算法的优劣。**

**先来先服务算法（FCFS）**

这是一种比较简单的磁盘调度算法。它根据进程请求访问磁盘的先后次序进行调度。

优点：

公平、简单

每个进程的请求都能依次得到处理，不会出现某一进程的请求长期得不到满足的情况。

缺点：

此算法由于未对寻道进行优化，在对磁盘的访问请求比较多的情况下，此算法将降低设备服务的吞吐量，致使平均寻道时间可能较长，但各进程得到服务的响应时间的变化幅度较小。

**最短寻道时间优先算法（SSTF）**

该算法选择这样的进程，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，

优点：

可以得到比较好的吞吐量，但却不能保证平均寻道时间最短。

缺点：

是对用户的服务请求的响应机会不是均等的，因而导致响应时间的变化幅度很大。在服务请求很多的情况下，对内外边缘磁道的请求将会无限期的被延迟，有些请求的响应时间将不可预期。

**扫描算法（SCAN）**

扫描算法不仅考虑到欲访问的磁道与当前磁道的距离，更优先考虑的是磁头的当前移动方向。

优点：

克服了最短寻道时间优先算法的服务集中于中间磁道和响应时间变化比较大的缺点，而具有最短寻道时间优先算法的优点即吞吐量较大，平均响应时间较小，但由于是摆动式的扫描方法，两侧磁道被访问的频率仍低于中间磁道

**92.文件系统的管理对象是什么？**

文件、目录、磁盘（磁带）存储空间

**93.文件的逻辑结构类型如何划分**

1、按文件是否有结构分类

2、按文件的组织方式分类

**94.有结构的顺序文件和索引文件分别如何存取**

对于定长的顺序文件：

读文件：设置一个读指针Rptr，令它指向下一个记录的首地址，每当读完一个记录时，便执行Rptr=Rptr+1操作，L为记录长度；

写文件：设置一个写指针Wptr，使之指向要写的记录的首地址，每写完一个记录，执行Wptr=Wptr+1操作。

对于变长记录的文件：

同样设置读写指针，每次读或写完一个记录后，将读或写指针加上刚读或刚写完的记录的长度。

对于索引文件：

使用关键字建立索引表，根据用户（程序）提供的关键字用折半查找法去检索索引表，从中找到相应的表项；

再利用该表项中给出的指向记录的指针值去访问所需的记录。而每当要向索引文件中增加一个新记录时，便须对索引表进行修改。

**95.文件目录的基本功能有哪些**

（1）实现按名存取

（2）提高对目录的检索速度

（3）文件共享

（4）允许文件重名

**96.引入索引结点的目的是什么？**

为了减少索引文件时启动磁盘的次数

**97.文件有哪两种共享方式**

1.利用索引节点实现文件共享

2.利用符号链接实现文件共享

**文件的访问矩阵保护方式如何实现**

1.访问控制表

2.访问权限表

**98.什么是事务？设置事务的目的是什么？**

(1)事务是用于访问和修改各种数据项的-一个程序单位

(2)事务也可以被看作是一系列相关读和写操作

(3)事务的执行具有原子性

**一、常见的为什么引入**

**1：为什么引入进程？**

为了更好地控制和描述程序的并发执行，实现操作系统的并发性和共享性，提高系统资源的利用率和系统吞吐量。

定义：具有独立功能的程序在有关数据集的一个动态的执行过程，是资源分配的基本单位。

特征：

结构特性：进程除了相应的程序段、数据段，还包含一个进程控制块PCB。

动态性：进程是程序的一次执行过程。进程有一定的生命周期，它必须由创建产生，由调度而执行，由撤销而消亡。

并发性：多个进程实体同时存在于内存中，且能在同一段时间内同时运行。

独立性：进程实体是一个能够独立运行，独立分配资源和独立接收调度的基本单位。

异步性：进程可按各自独立的、不可预知的速度向前推进。

**引申：引起调度的时机？**

1️执行中的进程执行完毕或因某事不再继续执行。

2️执行中的进程因提出I/O请求而暂停执行。

3️在进程通信或同步过程中执行了某种原语操作。

4️在剥夺式的调度中，比当前进程优先级更高的进程进入就绪队列。

5️时间片轮转算法中，时间片用完。

**2：为什么引入PCB？**

为了管理和控制进程的并发执行，协助进程完成状态转换。

引申：PCB的成分？为什么说PCB时进程存在的唯一标识？

成分：

1️进程描述信息：进程标识符、用户标识符。

2️进程管理和控制信息：进程当前状态、进程优先级、进入内存时间和处理机占用时间。

3️资源分配清单：程序段指针、数据段指针、堆栈段指针。

4️处理机相关信息：通用寄存器值、状态字。

唯一标志：

在进程的创建、运行、撤销周期中，系统总是通过PCB对进程进行控制，即系统是根据PCB感知进程存在。

**3：为什么引入线程？**

为了减少进程并发执行过程中的时空开销，使操作系统具有更好的并发性。

**4：为什么引入同步机制？**

由于多个并发执行的进程共享系统内存资源，需要多个进程协同合作完成某一项工作时，就产生了相互制约的关系。其中：

间接制约：多个进程要访问临界资源，从而产生了竞争的互斥关系。

直接制约：多个进程协同完成一项工作，存在执行顺序的要求，为同步关系。

四个原则：

1️空闲让进：临界区空闲，允许一个请求进入临界区的进程立即进入临界区。

2️忙则等待：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。

3️有限等待：对请求进入临界区的进程，应保证能在有限时间内进入临界区。

4️让权等待：当进程不能进入临界区时，应立即释放处理机，防止进程忙则等待。

**引申：临界区、临界资源是什么？**

临界资源： 必须互斥访问使用的资源。

临界区： 访问临界资源的那段代码

**5：为什么引入管程？**

大量分散的同步操作给操作系统管理带来了麻烦，容易操作不当造成死锁，管程的特性保证了进程互斥，无需程序员自己实现互斥，降低了死锁发生的可能性。

**6：为什么要引入重定位？**

程序的逻辑地址与其所在内存中的物理地址无直接关系，编译时逻辑地址都是从0开始，程序地址都是相对地址；

而在程序进入内存时，不可能总是从内存起始地址0开始，因此需要根据在其内存中的具体位置，将逻辑地址转换为物理地址。

具体又分为装入时重定位和运行时重定位。

**7：为什么引入I/O通道？**

虽然在 CPU 和 I/O设备间增加了设备控制器后，已能大大减少CPU对I/O的干预，但当主机所配置的外设很多时，CPU的负担仍然很重。为此在 CPU 和 设备控制器之间又增加了通道。

主要是为了建立独立的I/O操作，不仅使数据的传送独立于CPU，而且也希望有关对I/O操作的组织、管理及结束处理尽量独立，以保证CPU有更多的时间去进行数据处理；或者说，是使一些原来由CPU处理的I/O任务转由通道来承担，从而把CPU从繁杂的I/O任务中解脱出来。

**8：为什么要在设备管理中引入缓冲技术？**

1.缓和 CPU 和 I/O设备间的速度不匹配问题。

2.提高 CPU 和 I/O设备之间的并行性。

3.减少CPU中断频率，放宽对中断响应时间的限制。

4.解决数据粒度不匹配的问题。

**9：为什么要引入虚拟存储器？**

提高系统的内存利用率和系统吞吐量。

**二、关于死锁**

**1：什么是死锁**

多个进程因为竞争资源或执行时推进的顺序不当，引发的一种胶着状态，若无外力作用，这种状态将永远保持下去。

**2：死锁的必要条件**

1.互斥条件：进程要求对分配的资源进行排他性控制，即在一般时间内某资源仅为一个进程所占用。

2.不可抢占条件：进程所获得的资源在未使用完之前，不能被其他进程强行夺走，只能由获得该资源的进程自己释放。

3.请求和保持条件：进程已经保持了至少一个资源，但又提出了新的资源请求，而该资源已经被其他进程占有，此时请求进程被阻塞，但对自己已获得的资源保持不放。

3.循环等待条件：存在一种进程资源等待链，链中每个进程已获得的资源同时被链中下一个进程所请求。

**3：死锁预防与死锁避免的区别**

1.死锁预防：通过设立一些限制条件，破坏死锁的一些必要条件，让死锁无法产生。

2.死锁避免：在动态分配资源的过程中，用一些算法防止系统进入不安全状态。

**4：死锁与饥饿的区别**

1.死锁的进程都处于阻塞态，饥饿的进程不一定处于阻塞态。

2.死锁一定是多个进程，但不一定是全部，饥饿可能是只有一个进程。

3.死锁一定是发生了循环等待，饥饿不一定。

4.死锁进程等待是永远不会释放资源，饥饿进程可以释放资源，但等待时间无限。

5.死锁无外力作用，永远无法推进，饥饿在很长时间可以推进。

**5：死锁检测**

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，操作系统要判断系统是否出现了死锁，当有死锁发生时设法解除死锁。

**三、关于进程、线程和程序**

**1：进程与线程的区别**

1️调度性：传统的操作系统中拥有资源和独立调度的基本单位是进程，而引入线程的操作系统中，线程是调度的基本单位，而把进程作为资源拥有的基本单位。

2️并发性：引入线程的操作系统中，不仅进程间可以并发执行，而且统一进程内的多个线程也可以并发执行。

3️拥有资源：进程是拥有资源的基本单位，线程除本身必不可少的资源外，本身不具有资源，但可以访问其隶属进程的资源。

4️开销：创建或撤销进程时，系统都要为之分配和回收资源；进程切换时，所要保存和设置的现场信息也要明显多于线程。

**2：进程和程序的区别**

1️一个程序可以产生多个不同进程。

2️进程是动态的，程序是静态的。

3️进程是暂时的，程序是永久的。

4️进程的组成包括程序段、数据段和进程控制块PCB。

**3：现今操作系统申请CPU的基本单位？申请除CPU之外资源的基本单位是什么？**

现今的操作系统中，线程是进行处理机调度的基本单位，也就是说申请CPU的基本单位，而申请除CPU之外的资源的基本单位仍是进程。

进程为线程提供运行资源并构成静态环境。引入线程后，同一进程中的所有线程共享进程所获得的主存空间和资源，操作系统可以在线程间进行快速切换，大大提高了系统的运行性能。

**4：进程的创建**

原因：用户登录、作业调度、系统提供服务、用户程序的应用请求。

过程：

1️为新进程分配一个唯一的进程标识符，并申请一个空白的PCB。

2️为进程分配资源，为新进程的程序数据，以及用户栈分配必要的内存空间；若失败，则阻塞。

3️初试化PCB，主要包括初始化标志信息、处理机状态信息和处理机控制信息，以及设置进程优先级。

4️如果进程就绪队列能够接纳新进程，就将进程插入就绪队列等待调度执行。

**5：进程的切换**

1️保存处理机上下文，包括程序计数器和其他寄存器。

2️更新PCB信息。

3️把相应进程的PCB移入相应的队列，如就绪队列。

4️选择另一个进程执行，并更新其PCB。

5️更新内存管理的数据结构。

6️恢复处理机上下文。

**6：进程的阻塞**

原因：向系统请求共享资源失败、等待某种操作完成、新数据尚未到达、等待新任务到达。

过程：

1️找到将要被阻塞进程的标识号对应的PCB。

2️若该进程为运行态，则保护其现场，将其状态转为阻塞态，停止运行。

3️把该PCB插入相应事件的等待队列，将处理机资源调度给其他就绪进程。

**7：进程的唤醒**

原因：进程所期待的事件出现、I/O完成、数据已到达

过程：

1️在该事件的等待队列中找到相应的进程的PCB。

2️将其从等待队列中移出，并置其状态为就绪态。

3️把该PCB插入就绪队列，等待调度程序调度。

**8：父进程与子进程**

1️在操作系统，允许一个进程创建另一个进程，创建进程的进程称为父进程，被创建的进程为子进程。

2️子进程可以继承父进程所拥有的资源，但不共享虚拟空间。

3️子进程会申请一个空白的进程控制块PCB。

4️父子进程可以并发进行，但不能使用同一临界资源。

**9：进程切换与模式切换**

1️模式切换是处理机在内核态与用户态，可以在同一个进程中切换也可以在不同的进程中切换。

2️进程切换是指CPU在不同进程之间切换。

3️进程切换必导致模式切换，反之不一定。

**10：调度和切换**

1️调度是指决定资源分配给哪个进程的行为，是一种决策行为。

2️切换是指实际分配的行为，是一种执行行为。

3️一般先有资源调度，后有进程切换。

**11：程序的局部性原理**

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

时间局部性：一条指令执行后不久可能被再次执行，数据一旦被访问后，不久后数据可能被再次访问。

空间局部性：一旦程序访问了某个存储单元，不久后附近的存储单元也将被访问。

**12：系统调用与库函数的区别**

库函数是应用程序的一部分，可以运行在用户空间中；而系统调用是操作系统的一部分，是内核提供给用户的程序接口，由应用程序调用，通过预中断进入内核态，运行在内核空间中。

许多库函数会使用系统调用来实现功能，没有用系统调用的库函数执行效率比系统调用高，因为系统调用需要内核态与用户态转化。

**四、内存管理**

**1：什么是抖动？如何解决？**

在页面置换过程中，刚刚换入主存的页面马上就要换出，刚刚换出主存的页面马上就要换入，这种频繁的页面调度，导致系统的效率急剧下降的现象称为抖动。

解决：

1️修改页面置换算法

2️降低多道程序的数量

3️终止该进程

4️增加物理内存的容量

**2：虚拟存储器是？如何实现？**

定义：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。

注意：虚拟存储器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。

实际容量为：min{内存、外存容量之和，CPU寻址范围}

最大容量为：CPU寻址范围，即计算机地址结构

实现：首先需要扩充页表增加状态位、访问标志、修改标志和外存地址等标志。同时，还需要实现请求调页技术和页面置换技术。请求调页技术是通过缺页中断技术机制将进程所需页面及时调入内存。页面置换技术是在内存空间紧张时，选择被淘汰的页面，以腾出内存的空间。

基本特征：

1️多次性：作业只要部分装入内存便可启动执行，其余部分可待需要时，再调入内存，即一个作业分多次装入内存。

2️对换性：在程序运行期间，允许将那些暂时不使用的程序和数据从内存调至外存的对换区，待需要时再将它们换入。

3️虚拟性：只是逻辑上扩充内存容量，而实际的内存容量没有扩大。

**3：程序链接的三种方式**

静态链接：

在程序装入之前，先将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。

动态链接的定义：

指用户的各程序段不是在程序开始运行前就链接好，而是程序装入或运行过程中，当发现要调用的程序未链接时才进行链接。

装入时动态链接：

将用户源程序编译后所得到的一组目标模块，在装入内存时，来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模块时，才对它进行链接。其优点是便于修改和更改，便于实现对目标模块的共享。

**4：外存分配的几种方式，以及优缺点？**

**连续分配**

指为每个文件分配一组相邻的物理块，并将文件中的信息按逻辑顺序依次存放在这些物理块中。

优点：可以随机访问磁盘顺序访问速度快。

缺点：要求有连续的存储空间，容易产生外部碎片，磁盘利用率低，且不利于文件的增长扩充。

**链接分配**

指通过每个磁盘块上的链接指针，将同属于一个文件的多个离散的盘块链接成一个链表。

优点：不要求连续的存储空间，磁盘利用率极高，有利于文件的增长扩充。

缺点：只适合顺序访问，不适合随机访问；文件数据块之间靠指针链接，耗费额外存储空间，可靠性差。

**索引分配**

为每个文件分配一个索引块，再把分配给文件的所有盘块号都记录在该索引块中。

优点：既支持顺序访问，又支持随机访问，查找效率高，便于文件的删除。

缺点：索引表占用一定的存储空间。

**5：段页式存储管理时如何形成的，从地址空间划分、地址结构及地址转换等方面简述其原理。**

形成：

分页系统能有效提高内存的利用率，而分段系统能反映程序的逻辑结构，便于段的共享与保护，将分页与分段两种存储方式结合起来，就形成了段页式存储管理方式。

地址空间划分：

在段页式存储管理系统中，作业的地址空间首先被分成若干个逻辑分段，每段都有自己的段号，然后再将每段分成若干个大小相等的页。对于主存空间也分成大小相等的页，主存的分配以页为单位。

地址结构：

段页式系统中，作业的地址结构包含三部分内容：段号、页号、页内偏移量，程序员按照分段系统的地址结构将地址分为段号和段内位移量，地址变换机构将段内位移量分解为页号和页内偏移量。

地址转换：

首先用段号S 与 表长TL 进行比较，若S<TL，表示未越界，于是利用段表起始地址和段号来求出段所对应的段表项在段表中的位置，从中得到该段的页表起始地址，并利用逻辑地址中的段内页号p来获得对应页的页表项位置，从中读出该页所在的物理块号，再利用块号b和页内地址来构成物理地址。

**五、文件管理**

**1：如何实现文件共享？**

1️基于索引结点的文件共享：

在文件的目录中填上需要共享文件的索引结点的序号，在索引结点中加上用户计数。

2️基于符号链的文件共享：

建立一种特殊的链接文件，内容为需要共享文件的路径和名字，访问该文件时，根据路径找到该文件。

2：顺序文件、索引文件、索引顺序文件和HASH文件

顺序文件：

一系列记录按照某种顺序排列所形成的文件。

优点：当需要对文件进行批量存取时，它的存取效率最高。

缺点：当文件较大时，记录的检查效率较低。记录的增加和删除比较困难。

索引顺序文件：

索引文件和顺序文件的结合，将顺序文件中所有记录分为若干个组，并为顺序文件建立一张索引表，在索引表中为每组第一个记录建立一个索引项，其中包含有该记录的键值和指向该记录的指针。

检索过程：

在对索引顺序文件检索时，先充分利用用户所给关键字去检索索引表，找到该记录所在记录组中第一个记录的表项，从中得到该记录组第一个记录在主文件中的位置；再用顺序查找去查找主文件，找到所要求的记录。

**六、设备管理**

**1：如何在设备管理中现实缓冲技术？**

1️用专门的硬件缓冲器、

2️内存缓冲区，在内存中画出一个专用的缓冲区，以便存入输出的数据。

**2：SPOOLing技术**

1.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，又称假脱机技术。

2.在多道程序环境下，利用多道程序中的一道或者两道程序来模拟脱机输入\输出中的外围控制机的功能，以达到 “脱机” 输入\输出的目的。

3.利用这种技术可以把独占设备转变为共享的虚拟设备，从而提高独占设备的利用率和进程的推进速度

**3：具有I/O通道的系统分配设备的过程是什么？**

1️分配设备：

根据进程给出的物理设备名查找系统设备表SDT，从中找出改设备的设备控制表DCT。检查设备状态字段，若设备忙或本次分配会导致系统进入不安全状态，则将该进程插入设备等待队列；否则将设备分配给该进程。

2️分配控制器：

根据DCT中指向控制器的指针找到对应的COCT表，检测控制器状态字段，忙则将请求进程插入控制器等待队列；否则，将控制器分配给请求进程。

3️分配通信：

通过COCT找到对应的通道控制表CHCT，检查通道状态，忙则将该请求进程放入通道等待队列；否则将通道分配给进程。

只有当设备、控制器和通道都分配成功，才算成功，然后便可启动I/O设备进行数据传送。

**七、几个常问的技术**

**1：何为批处理系统，以及优点**

用户准备好要执行的程序、数据和控制作业执行的说明书，由操作员输入到计算机系统中等待处理，操作系统选择作业并按作业说明书的要求自动控制作业执行。

优点：

1️ 多道作业并行工作，减少CPU空闲时间。

2️ 作业成批输入，减少I/O次数。

3️ 作业调度可以合理选择装入主存的作业，充分利用计算机系统资源。

4️ 作业执行过程不再访问低速设备，而直接访问高速的磁盘设备，缩短执行时间。

**2：多道程序设计**

1.允许多个程序同时进入一个计算机系统的主存并启动进行运算的方法，即计算机内存中可以同时存取多道独立程序，它们都处于开始和结束之间。

2.从宏观上看是并行的，多道程序都处于运行中，并且都没有结束。

3.从微观上看是串行的，多道程序轮流使用CPU，交替执行。

4.根本目的是为了提高CPU的利用率，充分发挥计算机系统的并行性。

**3：分时技术**

把处理机的运行时间分为很短的时间片，按时间片轮转把CPU分给各联机作业使用；如果某个作业分配的时间片用完，计算未完成，该作业就中断，调入就绪队列，等待下轮继续计算，此时把CPU给下一个进程。