

电子科大复试综合面试题回忆（2018版）

写在前面

这份文件为电子科大复试综合面试题回忆（2018版），其中大部分题目为今年的更新版本，建议以后的学弟学妹们，将群里的专业课复试问答完全版---默焱整理的pdf文件与该份文件结合复习。

数据库

1.数据库中的表如何设置主键？

首先说明设置主键的目的：

- 为了保证数据查找唯一；
- 提高存储效率和索引效率。

主键的设置方法有3种：一种是设置自增长主键,第二种是采用业务主键，第三种是生成唯一序列(使用 **uuid/guid**)；

1.使用自增主键的场景：

- 表中找不出3个以内字段可以表示的业务主键
- 没有集中大量的插入操作
- 不使用自增id作为外键（即id变化不影响系统逻辑和功能）

此外自增主键的规律很明显，为了不让别人轻易的采集数据，同时符合以上条件，则优先选择自增主键；

2.使用业务主键做id：

- 如果你设计的表上存在非重复数据列，且常以该列检索数据，或该列还关联其他表的外键，譬如学号、会员卡号、身份证号这些，可以用来做为主键，主键在默认设置下是聚集索引，这样检索学号、卡号这些效率上就会比较高。

3.使用生成唯一的序列

- 如果不满足使用自增的要求，则建议优先使用字符主键。类似网站数据库，可以采用uuid、guid这类无规则字符做主键。因为字符主键查询速度不比自增主键慢。
- 考虑自增主键的弊端。很多表的ID会被其他表引用为外键，而且自增主键弊端是：用DELETE删除后，不会重新接着自增，而是接着删除前的ID自增;在多数据库迁移下，自增则会很繁琐，需要另外做联合主键才能保证数据的唯一性。

2.事务的概念?

所谓事务是用户定义的一个数据库操作序列，这些操作要么全做要么全不做，是一个不可分割的工作单位。例如，在关系数据库中，一个事务可以是一条SQL语句、一组SQL语句或者整个程序。

事务和程序是两个概念。一般的讲，一个程序中包含多个事务。事务的开始和结束可以由用户显示控制。如果用户没有显示的定义事务，则有DBMS按照缺省规定自动划分事务。在SQL中，定义事务的语句有3条：

```
1 BEGIN TRANSACTION
2 COMMIT
3 ROLLBACK
```

事务通常以BEGIN TRANSACTION开始，以COMMIT或ROLLBACK结束。COMMIT表示提交，即提交事务的所有操作。具体的说就是将事务中所有对数据库的更新写回到磁盘上的物理数据库中去，事务正常结束。ROLLBACK表示回滚，即在事务运行的过程中发生了某种故障，事务不能继续执行，系统将事务中对数据库的所有已完成的操作全部撤销，回滚到事务开始时的状态。这里的操作指对数据库的更新操作。

3.数据库中如果只满足1NF，会有哪些问题?

数据冗余过大、插入异常、删除异常、修改异常，详细可参考：<https://www.zhihu.com/question/24696366>

计算机网络

1.DHCP的工作过程

DHCP分为两个部分：一个是服务器端，另一个是客户端。

所有客户机的IP地址设定资料都由DHCP服务器集中管理，并负责处理客户端的DHCP请求；而客户端则会使用从服务器分配下来的IP地址。

1.DHCP服务器IP分配方式

DHCP服务器提供三种IP分配方式：

- 自动分配（Automatic Allocation）
自动分配是当DHCP客户端第一次成功地从DHCP服务器端分配到一个IP地址之后，就永远使用这个地址。
- 动态分配（Dynamic Allocation）
动态分配是当DHCP客户端第一次从DHCP服务器分配到IP地址后，并非永久地使用该地址，每次使用完后，DHCP客户端就得释放这个IP地址，以给其他客户端使用。
- 手动分配
手动分配是由DHCP服务器管理员专门为客户端指定IP地址。

2.DHCP服务工作流程

DHCP客户机在启动时，会搜寻网络中是否存在DHCP服务器。如果找到，则给DHCP服务器发送一个请求。DHCP服务器接到请求后，为DHCP客户机选择TCP/IP配置的参数，并把这些参数发送给客户端。如果已配置冲突检测设置，则DHCP服务器在将租约中的地址提供给客户机之前会使用Ping测试作用域中每个可用地址的连通性。这可确保提供给客户的每个IP地址都没有被使用手动TCP/IP配置的另一台非DHCP计算机使用。

根据客户端是否第一次登录网络，DHCP的工作形式会有所不同。客户端从DHCP服务器上获得IP地址的所有过程可以分为以下六个步骤：

- 工作过程1：寻找DHCP服务器
- 工作过程2：分配IP地址
- 工作过程3：接受IP地址
- 工作过程4：IP地址分配确认
- 工作过程5：重新登录
- 工作过程6：更新租约

详细可参考：<http://www.zyops.com/dhcp-working-procedure>

2.为什么要用UDP协议以及使用UDP协议的优势？

为什么要用UDP协议？

1.网速的提升给UDP稳定性提供可靠网络保障

CDN服务商Akamai（NASDAQ: AKAM）报告从2008年到2015年7年时间，各个国家网络平均速率由1.5Mbps提升为5.1Mbps，网速提升近4倍。网络环境变好，网络传输的延迟、稳定性也随之改善，UDP的丢包率低于5%，如果再使用应用层重传，能够完全确保传输的可靠性。

2.对比测试结果UDP性能优于TCP

为了提升浏览速度，Google基于TCP提出了SPDY协议以及HTTP/2。Google在Chrome上实验基于UDP的QUIC协议，传输速率减少到100ms以内。

3.TCP设计过于冗余，速度难以进一步提升

TCP为了实现网络通信的可靠性，使用了复杂的拥塞控制算法，建立了繁琐的握手过程以及重传策略。由于TCP内

置在系统协议栈中，极难对其进行改进。

4.UDP协议以其简单、传输快的优势，在越来越多场景下取代了TCP

使用UDP协议有三个优点：

- 能够对握手过程进行精简，减少网络通信往返次数；
- 能够对TLS加解密过程进行优化；
- 收发快速，无阻塞。

3.简单介绍下c/s体系结构？

C = Client, S = Server。C/S 架构即“客户端-服务器”架构。这里的“客户端”可以有 GUI（图形用户界面）的定制软件，也可以是浏览器，甚至可以通过SSH访问服务器的命令行脚本。只要是客户端通过访问服务器调取计算或者存储资源的，都是 C/S 结构。

C/S结构的优点是能充分发挥客户端PC的处理能力，很多工作可以在客户端处理后再提交给服务器。对应的优点就是客户端响应速度快。

4.c/s体系结构和p2p网络有什么区别？

C/S 架构是一种典型的两层架构，全称是Client/Server，即客户端/服务器端架构，其客户端包含一个或多个在用户的电脑上运行的程序，而服务器端有两种，一种是数据库服务器端，客户端通过数据库连接访问服务器端的数据；另一种是Socket服务器端，服务器端的程序通过Socket与客户端的程序通信。

比如：微信/客户端QQ等是基于C/S架构。

P2P是英文Peer-to-Peer（对等）的简称，又被称为“点对点”。“对等”技术，是一种网络新技术，依赖网络中参与者的计算能力和带宽，而不是把依赖都聚集在较少的几台服务器上。

P2P还是英文Point to Point（点对点）的简称。它是下载术语，意思是在你自己下载的同时，自己的电脑还要继续做主机上传，这种下载方式，人越多速度越快但缺点是对硬盘损伤比较大（在写的同时还要读），还有对内存占用较多，影响整机速度。

P2P架构的核心思想是每个节点既可以充当客户端（Client），又可以充当服务器端（Server）。

比如：BT/电驴下载，非法传播视频的网站等，因为每个结点既是客户端可以进行下载，又是服务器端可以继续上传资源以提供下载服务给其他人，所以找不到真正的Server，打击非法网站也就难上加难了，所以P2P架构就是一种“我为人人，人人为我”的资源共享思想。如果是C/S架构，比如HTTP协议，只需单点攻击Server，整个C/S架构就失去了最核心的服务器端部分，基于C/S架构的通信也就被攻破了。

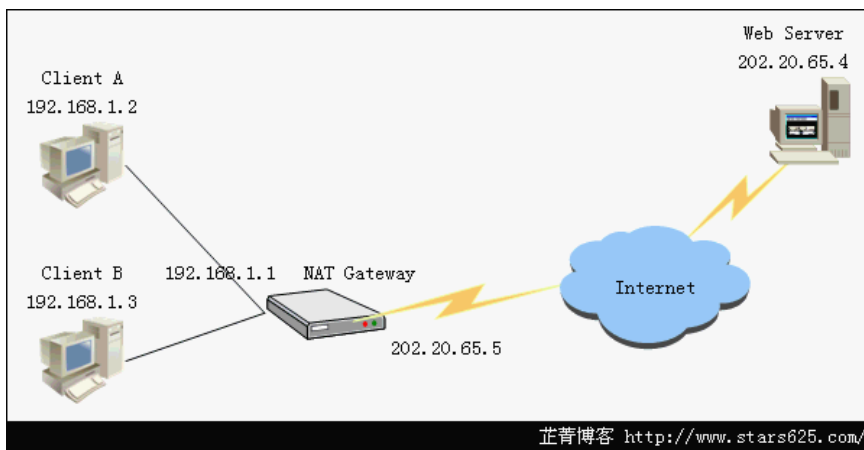
5.NAT工作的流程(让你解释什么ip地址转发的流程)

NAT英文全称是“Network Address Translation”，中文意思是“网络地址转换”，它是一个IETF(Internet Engineering Task Force, Internet工程任务组)标准，允许一个整体机构以一个公用IP（Internet Protocol）地址出现在Internet上。顾名思义，它是一种把内部私有网络地址（IP地址）翻译成合法网络IP地址的技术。因此我们可以认为，NAT在一定程度上，能够有效的解决公网地址不足的问题。

5.1 地址转换

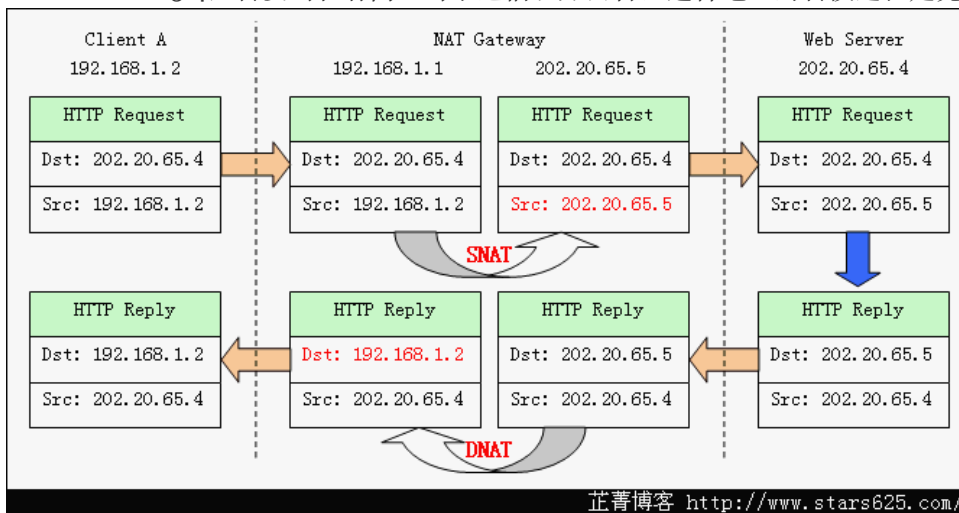
NAT的基本工作原理是，当私有网主机和公网主机通信的IP包经过NAT网关时，将IP包中的源IP或目的IP在私有IP和NAT的公共IP之间进行转换。

如下图所示，NAT网关有2个网络端口，其中公共网络端口的IP地址是统一分配的公共IP，为202.20.65.5；私有网络端口的IP地址是保留地址，为192.168.1.1。私有网中的主机192.168.1.2向公网中的主机202.20.65.4发送了1个IP包(Dst=202.20.65.4,Src=192.168.1.2)。



当IP包经过NAT网关时，NAT Gateway会将IP包的源IP转换为NAT Gateway的公共IP并转发到公共网，此时IP包（Dst=202.20.65.4，Src=202.20.65.5）中已经不含任何私有网IP的信息。由于IP包的源IP已经被转换成NAT Gateway的公共IP，Web Server发出的响应IP包（Dst= 202.20.65.5,Src=202.20.65.4）将被发送到NAT Gateway。

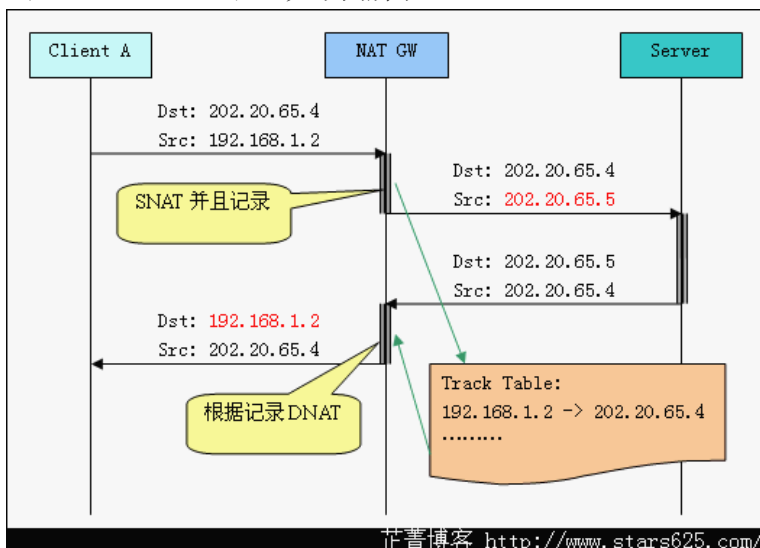
这时，NAT Gateway会将IP包的目IP转换成私有网中主机的IP，然后将IP包（Des=192.168.1.2，Src=202.20.65.4）转发到私有网。对于通信双方而言，这种地址的转换过程是完全透明的，转换示意图如下。



如果内网主机发出的请求包未经过NAT，那么当Web Server收到请求包，回复的响应包中的目的地址就是私网IP地址，在Internet上无法正确送达，导致连接失败。

5.2 连接跟踪

在上述过程中，NAT Gateway在收到响应包后，就需要判断将数据包转发给谁。此时如果子网内仅有少量客户机，可以用静态NAT手工指定；但如果内网有多台客户机，并且各自访问不同网站，这时候就需要连接跟踪（connection track）。如下图所示：

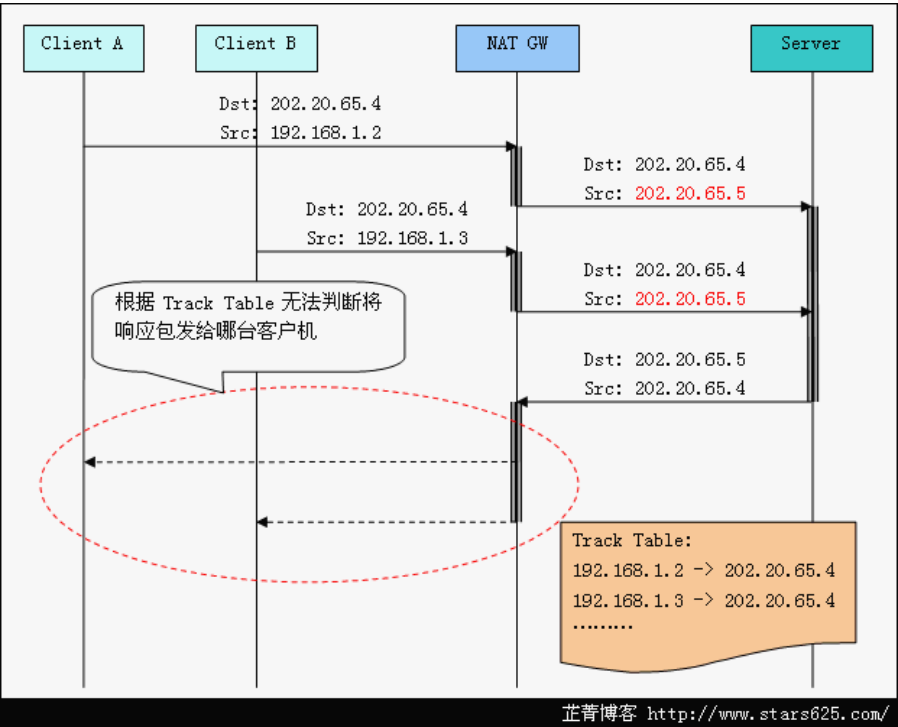


在NAT Gateway收到客户机发来的请求包后，做源地址转换，并且将该连接记录保存下来，当NAT Gateway

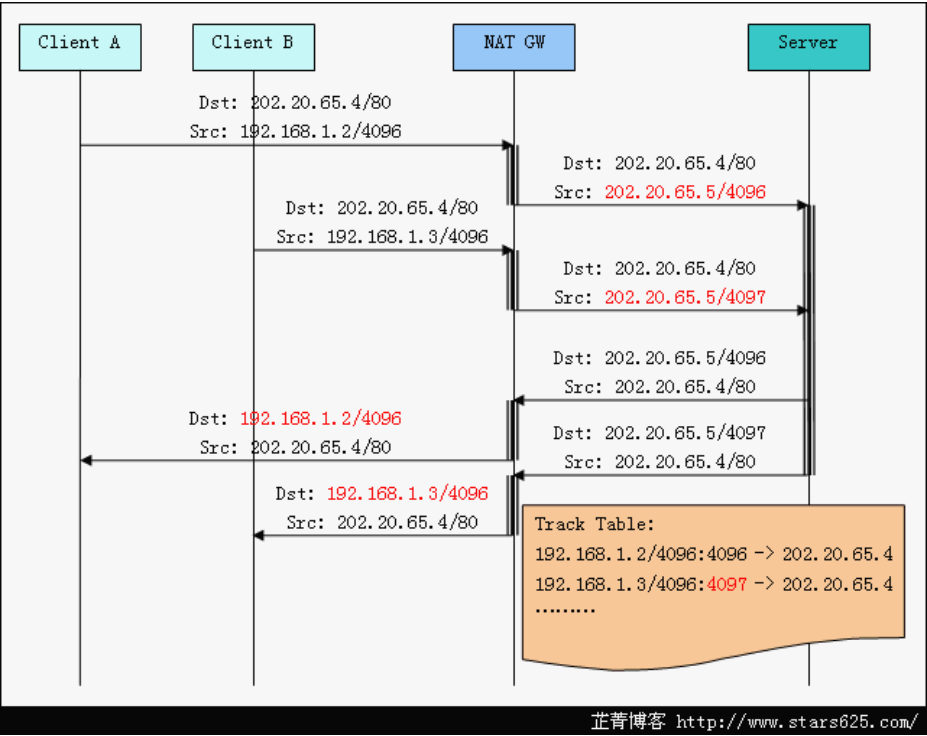
收到服务器来的响应包后，查找Track Table，确定转发目标，做目的地址转换，转发给客户机。

5.3 端口转换

以上述客户机访问服务器为例，当仅有一台客户机访问服务器时，NAT Gateway只须更改数据包的源IP或目的IP即可正常通讯。但是如果Client A和Client B同时访问Web Server，那么当NAT Gateway收到响应包的时候，就无法判断将数据包转发给哪台客户机，如下图所示。



此时，NAT Gateway会在Connection Track中加入端口信息加以区分。如果两客户机访问同一服务器的源端口不同，那么在Track Table里加入端口信息即可区分，如果源端口正好相同，那么在时行SNAT和DNAT的同时对源端口也要做相应的转换，如下图所示。



6.IP地址和MAC地址的区别

互联网协议地址（Internet Protocol Address，又译为网际协议地址），缩写为IP地址（IP Address），是分配给网络上使用网际协议（英语：Internet Protocol, IP）的设备的数字标签。常见的IP地址分为IPv4与IPv6两大类。

IP地址由32位二进制数组成，为便于使用，常以XXX.XXX.XXX.XXX形式表现，每组XXX代表小于或等于

255的10进制数。例如维基媒体的一个IP地址是208.80.152.2。地址可分为A、B、C、D、E五大类，其中E类属于特殊保留地址。

MAC地址就是在媒体接入层上使用的地址，也叫物理地址、硬件地址或链路地址，由网络设备制造商生产时写在硬件内部。MAC地址与网络无关，也即无论将带有这个地址的硬件（如网卡、集线器、路由器等）接入到网络的何处，都有相同的MAC地址，它由厂商写在网卡的BIOS里。

MAC地址通常表示为12个16进制数，每2个16进制数之间用冒号隔开，如：08:00:20:0A:8C:6D就是一个MAC地址，其中前6位16进制数08:00:20代表网络硬件制造商的编号，它由IEEE分配，而后6位16进制数0A:8C:6D代表该制造商所制造的某个网络产品（如网卡）的系列号。每个网络制造商必须确保它所制造的每个以太网设备都具有相同的前三字节以及不同的后三个字节。这样就可保证世界上每个以太网设备都具有唯一的MAC地址。

IP地址与MAC地址的区别

IP地址基于逻辑，比较灵活，不受硬件限制，也容易记忆。MAC地址在一定程度上与硬件一致，基于物理，能够标识具体。这两种地址各有好处，使用时也因条件而采取不同的地址。

7.如何实现ipv4和ipv6的互联？

(1).双栈技术

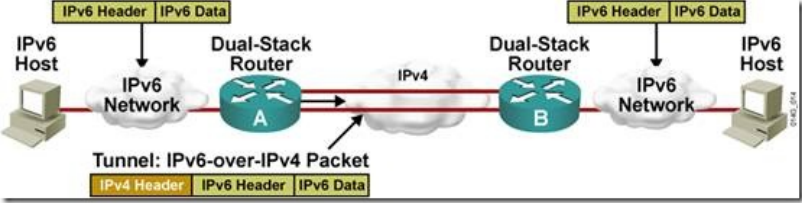
双栈方式的工作机制可以简单描述为：数据链路层解析出接收到的数据包的数据段，拆开并检查包头。如果IPv4/IPv6包头中的第一个字段，即IP包的版本号是4，该包就由IPv4的协议栈来处理；如果版本号是6，则由IPv6的协议栈处理。

(2).隧道技术

随着IPv6网络的发展，出现了许多局部的IPv6网络。为了实现这些孤立的IPv6网络之间的互通，采用隧道技术。隧道技术是在IPv6网络与IPv4网络间的隧道入口处，由路由器将IPv6的数据分组封装入到IPv4分组中。IPv4分组的源地址和目的地址分别是隧道入口和出口的IPv4地址。在隧道的出口处拆封IPv4分组并剥离出IPv6数据包。

隧道技术的优点在于隧道的透明性，IPv6主机之间的通信可以忽略隧道的存在，隧道只起到物理通道的作用。在IPv6发展初期，隧道技术穿越现存IPv4因特网实现了IPv6孤岛间的互通，从而逐步扩大了IPv6的实现范围，因而是IPv4向IPv6过渡初期最易于采用的技术。但是隧道技术不能实现IPv4主机与IPv6主机的直接通信。

IPv6网络边缘设备收到IPv6网络的IPv6报文后，将IPv6报文封装在IPv4报文中，成为一个IPv4报文，在IPv4网络中传输到目的IPv6网络的边缘设备后，解封装去掉外部IPv4头，恢复原来的IPv6报文，进行IPv6转发。



8.如果你是公司网络管理员你怎么配置公司网络（开放性问題）？

这道题是开放性问題，我这里没有搜到好的标准答案，有好的答案的小伙伴可以联系我。

操作系统

1.什么是动态链接？动态链接怎么实现？为什么要引入动态链接？

为什么要引入动态链接

随着软件规模的逐渐增大，静态链接的缺点也逐渐暴露，有如下几个问题

- 由于每个elf本身包含静态库，同时加载使也需要加载到内存中，浪费内存和磁盘空间
- 不利于程序的开发和发布

为解决上述问题，引入了动态链接

动态链接(Dynamic Linking)的基本思想

动态链接需要解决空间浪费和更新困难这两个问题，最简单的办法就是把程序的模块相互划分开来，形成独立的文件，而不再将他们静态的链接在一起。简单地讲，就是不对那些组成程序的目标文件进行链接，等到程序要运行时才进行链接。也就是说，把链接这个过程推迟到了运行时再进行。

动态链接的实现

动态链接，在可执行文件装载时或运行时，由操作系统的装载程序加载库。大多数操作系统将解析外部引用（比如库）作为加载过程的一部分。在这些系统上，可执行文件包含一个叫做import directory的表，该表的每一项包含一个库的名字。根据表中记录的名字，装载程序在硬盘上搜索需要的库，然后将其加载到内存中预先不确定的位置，之后根据加载库后确定的库的地址更新可执行程序。可执行程序根据更新后的库信息调用库中的函数或引用库中的数据。这种类型的动态加载成为装载时加载，被包括Windows和Linux的大多数系统采用。

2.简单叙述下缓冲区“空间换时间”的思想，并举一个关于DISK CACHE的例子。（这题没有找到恰当的答案，如果有更好的答案的小伙伴们欢迎分享）

“缓存”（缓冲区），自然就是“用空间换时间”的算法。缓存（缓冲区）就是把一些数据暂时存放于某些地方，可能是内存，也有可能硬盘。总之，目的就是为了避免某些耗时的操作。我们常见的耗时的操作，比如数据库的查询、一些数据的计算结果，或者是为了减轻服务器的压力。其实减轻压力也是因查询或计算，虽然短耗时，但操作很频繁，累加起来也很长，造成严重排队等情况，服务器抗不住。

关于DISK CACHE我没有找到合适的例子，欢迎小伙伴们补充。

3.简述磁盘调度的目的，以及磁盘调度有哪些算法？

磁盘调度的目的是为了缩短寻道时间（seek time），磁盘调度有先来先服务(FCFS)、最短寻道时间优先(SSTF)、扫描(SCAN)(又称电梯调度)、循环扫描(CSCAN)等算法。

4.简述什么是中断机制及中断机制的作用

中断机制的定义

中断机制是现代计算机系统中的基本机制之一，它在系统中起着通信网络的作用，以协调系统对各种外部事件的响应和处理，中断是实现多道程序设计的必要条件，中断是CPU对系统发生的某个事件作出的一种反应。引起中断的事件称为中断源。中断源向CPU提出处理的请求称为中断请求。发生中断时被打断程序的暂停点称为断点。CPU暂停现行程序而转为响应中断请求的过程称为中断响应。处理中断源的程序称为中断处理程序。CPU执行有关的中断处理程序称为中断处理。而返回断点的过程称为中断返回。中断的实现由软件和硬件综合完成，硬件部分叫做硬件装置，软件部分称为软件处理程序。

中断机制的作用

中断是用以提高计算机工作效率、增强计算机功能的一项重要技术。最初引入硬件中断，只是出于性能上的考量。如果计算机系统没有中断，则处理器与外部设备通信时，它必须在向该设备发出指令后进行忙等待（Busy waiting），反复轮询该设备是否完成了动作并返回结果。这就造成了大量处理器周期被浪费。引入中断以后，当处理器发出设备请求后就可以立即返回以处理其他任务，而当设备完成动作后，发送中断信号给处理器，后者就可以再回过头获取处理结果。这样，在设备进行处理的周期内，处理器可以执行其他一些有意义的工作，而只付出一

些很小的切换所引发的时间代价。后来被用于CPU外部与内部紧急事件的处理、机器故障的处理、时间控制等多个方面，并产生通过软件方式进入中断处理（软中断）的概念。

5.怎样实现存储保护？

上下限寄存器法和基址限长寄存器法，具体展开可参考相关操作系统课本。

6.说明读者/写者模型在计算机中的实际应用

一个例子：读写锁是一种特殊的自旋锁，它把对共享资源的访问者划分成读者和写者，读者只对共享资源进行读访问，写者则需要对共享资源进行写操作。这种锁相对于自旋锁而言，能提高并发性，因为在多处理器系统中，它允许同时有多个读者来访问共享资源，最大可能的读者数为实际的逻辑CPU数。写者是排他性的，一个读写锁同时只能有一个写者或多个读者，但不能同时既有读者又有写者。

完整描述可参考：<http://blog.csdn.net/Yyingc/article/details/73506057>

还有其他更好的例子的小伙伴们可以补充。

7.抢占非抢占哪个更耗系统资源？为什么？

这道题没有找到比较合适的答案，从百度百科上的描述可知，抢占式系统开销大，具体可参考：<https://baike.baidu.com/item/非抢占式/669995?fr=aladdin>

8.操作系统有什么接口？

操作系统提供了以下三种类型的接口供用户使用：

1.命令接口

- 提供一组命令供用户直接或间接操作。
- 根据作业的方式不同，命令接口又分为联机命令接口和脱节命令接口。

2.程序接口

- 程序接口由一组系统调用命令组成，提供一组系统调用命令供用户程序使用。

3.图形界面接口

- 通过图标 窗口 菜单 对话框及其他元素,和文字组合,在桌面上形成一个直观易懂 使用方便的计算机操作环境.

9.前台作业和后台作业有什么区别？为什么要优先调度前台作业？

批处理操作系统实现自动控制无需人为干预，分时操作系统实现了人机交互对话，这两种操作系统具有各自的优点。为了充分发挥批处理系统和分时系统的优点，在一个计算机系统上配置的操作系统往往既具有批处理能力，又有提供分时交互的能力。这样，用户可以先在分时系统的控制下，以交互式输入、调试和修改自己的程序；然后，可以把调试好的程序转交给批处理系统自动控制其执行而产生结果。这些由分时系统控制的作业称为"前台"作业，而那些由批处理系统控制的作业称为"后台"作业。

在这样的系统中，对前台作业应该及时响应，使用户满意；对后台作业可以按一定的原则进行组合，以提高系统的效率。

10.操作系统程序、任务、进程和线程的联系与区别

概念

- 程序(program)只是一组指令的有序集合。
- 任务是一个一般性的术语,指由软件完成的一个活动。一个任务既可以是一个进程,也可以是一个线程。简而言之,它指的是一系列共同达到某一目的的操作。例如,读取数据并将数据放入内存中。这个任务可以作为一个进程来实现,也可以作为一个线程（或作为一个中断任务）来实现。

- 进程(process)常常被定义为程序的执行。可以把一个进程看成是一个独立的程序,在内存中有其完备的数据空间和代码空间。一个进程所拥有的数据和变量只属于它自己。进程是表示资源分配的基本单位。
- 线程是进程中执行运算的最小单位,亦即执行处理机调度的基本单位。如果把进程理解为在逻辑上操作系统所完成的任务,那么线程表示完成该任务的许多可能的子任务之一。例如,假设用户启动了一个窗口中的数据库应用程序,操作系统将对数据库的调用表示为一个进程。假设用户要从数据库中产生一份工资单报表,并传到一个文件中,这是一个子任务;在产生工资单报表的过程中,用户又可以输入数据库查询请求,这又是一个子任务。这样,操作系统则把每一个请求——工资单报表和新输入的数据查询表示为数据库进程中的独立的线程。线程可以在处理器上独立调度执行,这样,在多处理器环境下就允许几个线程各自在单独处理器上进行。操作系统提供线程就是为了方便而有效地实现这种并发性

进程和程序区别和联系

- (1).程序只是一组指令的有序集合,它本身没有任何运行的含义,它只是一个静态的实体。而进程则不同,它是程序在某个数据集上的执行。进程是一个动态的实体,它有自己的生命周期。反映了一个程序在一定的数据集上运行的全部动态过程。
- (2).进程和程序并不是一一对应的,一个程序执行在不同的数据集上就成为不同的进程,可以用进程控制块来唯一地标识每个进程。
- (3).进程还具有并发性和交往性,这也与程序的封闭性不同。

进程与线程区别与联系

- (1).一个线程只能属于一个进程,而一个进程可以有多个线程,但至少有一个线程。
- (2).资源分配给进程,同一进程的所有线程共享该进程的所有资源。
- (3).处理机分给线程,即真正在处理机上运行的是线程。
- (4).线程在执行过程中,需要协作同步。不同进程的线程间要利用消息通信的办法实现同步。
- (5).划分尺度:线程更小,所以多线程程序并发性更高;
- (6).资源分配:进程是资源分配的基本单位,同一进程内多个线程共享其资源;
- (7).地址空间:进程拥有独立的地址空间,同一进程内多个线程共享其资源;
- (8).处理器调度:线程是处理器调度的基本单位;
- (9).执行:每个线程都有一个程序运行的入口,顺序执行序列和程序的出口,但线程不能单独执行,必须组成进程,一个进程至少有一个主线程。简而言之,一个程序至少有一个进程,一个进程至少有一个线程。