

软件设计师笔记（中级）

计算机系统知识

冯式结构计算机：存储器、运算器、控制器、输入设备和输出设备。

算术运算和逻辑运算的等价关系，一般来讲，左移相当于乘 2，右移相当于除 2，但注意符号位。被操作数的最高位移入“进位”位，其余所有位接收其相邻位低位值，最低位移入 0 的操作是逻辑左移指令。被操作数的最高位保持不变，其余所有位接收其相邻高位值，最低位移到“进位”位中的操作是除 2 运算指令。在程序执行过程中改变程序计数器顺序读出指令的指令属于转移指令。所谓寻址就是寻找操作数的地址，操作数所处的位置，可以决定指令的寻址方式。操作数包含在指令中，寻址方式为立即寻址；操作数在寄存器中，寻址方式为寄存器寻址；操作数的地址在寄存器中，寻址方式为寄存器间接寻址。相对寻址方式的实际地址是程序计数器的内容加上指令中形式地址值，得到需要的地址。特权指令是计算机的操作系统的基本指令，是由系统执行的，主要用于系统资源的分配和管理。

中央处理器 (CPU, Centrol Processing Unit) 由寄存器组、算术逻辑单元 ALU 和控制单元 CU 这三部分组成。CPU 的功能包括：(1) 读取指令 (2) 解释指令 (3) 读取数据 (4) 处理数据 (5) 保存数据。

计算机中的存储系统是用来保存数据和程序的，对存储器的最基本的要求就是存储容量要大、存取速度快、成本价格低。

主存储器是指能由 CPU 直接编程访问的存储器，它通常位于所谓主机的范畴，常称为内存。如果内存的地址为 n 位，则容量为 2 的 n 次方。辅助存储器的种类很多，主要有随机存储器 (RAM)、只读存储器 (ROM)、可编程 ROM (PROM)、可擦除 PROM (EPROM)、电可擦除 PROM (EEPROM)、闪速存储器 (Flash Memory)。

信息的存取方式影响到存储信息的组织，常用的有四种：(1) 顺序存取，存储器的数据是以记录的形式进行组织，对数据的访问必须按特定的线性顺序进行。磁带存储器就是顺序存取。(2) 直接存取，共享读写装置，可以直接移动到目的数据块所在位置进行访问。因此存取时间也是可变的。磁盘存储器采用的就是直接存储。(3) 随机存取，系统可以在相同的时间内对任意一个存储单元的数据进行访问，而与先前的访问序列无关。主存储器采用的是这种方式。(4) 相联存取，也是一种随机存取形式，但是选择某一单元进行读写是取决于其内容而不是其地址。相联存储器的访问方式最重要的特征是按照存储器中的单元内容访问，而不是按照地址访问。

高速缓存 (Cache) 用来存放当前最活跃的程序和数据。Cache 位于主存与 CPU 之间，解决主存与 CPU 速度匹配问题。其功能全部由硬件来实现，用于提高 CPU 的数据 I/O 效率，对程序员和系统程序员都是透明的。Cache 容量小但速度快，它在计算机的存储体系中是访问速度最快的层次。一般在几 KB 到几 MB 之间，速度比一般主存快 5 到 10 倍。Cache 由两部分组成：控制部分和 Cache 存储器部分。

Cache 中常见的映射方法有三种：直接映射，即一对一；全相联映射，即多对多；组相联映射，将块划分成组，主存中的一组与 Cache 相对应。

替换算法的目标是使 Cache 获得最高的命中率。常用算法有：随机替换算法、先进先出算法、近期最少使用算法、优化替换算法。为了保证 Cache 中的数据与

主存中的内容一致，对写操作来说有以下几种方法：直写式、缓写式、回写式、设置不可 Cache 区的方式。

FORTRAN 是第一个被广泛用于科学计算的高级语言。

COBOL 是一种面向事务处理的高级语言。

PASCAL 语言提供为数不多但相当紧凑的机制，使得这个语言具有相当强的表达能力。

C 是一种通用程序设计语言。C 作为一种较低级的语言，提供了指针和地址操作的功能。C 与 UNIX 操作系统紧密相关，UNIX 操作系统及其上的许多软件都是 C 编写的。

C++ 是在 C 语言的基础上发展起来的与 C 兼容的语言。主要增加了类功能，成为面向对象的程序设计语言。

磁盘存取时间=寻道时间+等待时间

寻道时间为磁头移动到目标磁道所需的时间。等待时间为待读写的扇区旋转到磁头下方所用的时间。一般用磁道旋转一周所用的时间的一半作为平均等待时间。

位密度是沿磁道方向，单位长度存储二进制信息的个数。道密度是沿磁盘半径方向，单位长度内磁道的数目。又因为每个磁道上的存储量相同，磁道越靠近圆心其圆周长越短，故不同磁道上的位密度，靠近圆心的密度大。

非格式化容量公式：记录面数*内直径的周长*位密度*内外半径之间的磁道数

格式化容量公式：记录面数*每个盘面的磁道数*每个磁道的扇区数*每个扇区的字节数

数据传输率公式：数据传输率= 一个磁道上的字节数/每转一周的时间

在微型计算机中最常见到的有两种编址方式：内存与接口地址独立的编址方法、内存与接口地址统一的编址方法。若某个计算机系统中，内存地址与 I/O 地址统一编址，访问内存单元和 I/O 设备是靠不同的地址代码区分的。

CPU 在与 I/O 设备进行数据交换时存在以下问题：速度不匹配、时序不匹配、信息格式不匹配、信息类型不匹配。

在中断响应过程中，CPU 程序计数器的主要目的是保存当前正在执行的程序的指令地址，也就是中断程序的断点处。在响应中断之前，CPU 要保存当前正在执行的程序或者指令的地址或状态，才转到中断服务程序执行相应的动作和过程，执行完毕后又回到主程序继续执行。中断的作用是提高 CPU 的处理效率，使 CPU 和 I/O 设备并行工作，还可以实现分时操作过程。

直接存储器访问(DMA, Direct Memory Access, 又称直接内存存取)是一种快速传递大量数据常用的技术。工作过程大致如下：向 CPU 申请 DMA 传送；获 CPU 允许后，DMA 控制器接管系统总线的控制权；在 DMA 控制器下，在存储器和外部设备之间直接进行数据传送，在传送过程中不需要 CPU 的参与，但在传输开始之前要提供所要传送数据的起始地址和需要传输的数据长度；传输结束后由 CPU 收回总线的控制权。

在系统有多个中断源的情况下，常用的处理方法有多中断信号线法(Multiple Interrupt Lines)、中断软件查询法(Software Poll)、雏菊链法(Daisy Chain)、总线仲裁法和中断向量表法。

流水线处理机的主要指标：吞吐率是指单位时间里流水线处理机流出的结果

数。对指令而言，就是单位时间里执行的指令数。如果流水线的子过程所用时间不一样长，则吞吐率 P 应为最长子过程的倒数。

利用并行处理技术可以缩短计算机的处理时间，所谓并行性是指在同一时间完成两种或两种以上的工作。可以采用多种措施来提高计算机系统的并行性，它们可分成三类，即资源重复、资源共享和时间重叠。提供专门用途的一类并行处理机（亦称阵列处理机）以 SIMD 方式工作，它适用于大量调整向量或矩阵运算。多处理机属于 MIMD 系统，是目前较高性能计算机的基本结构，它的并行性任务需要专门的指令来表示程序中它们的并发关系并控制它们并发执行。

精简指令系统 (RISC)，简化了 CPU 的控制器，同时提高了处理速度，其特点如下：指令种类少，一般只有十几条几十条简单指令；指令长度固定，指令格式少，这使指令译码更加简单；寻址方式少；设置最少的访问指令，访问内存比较花时间，应尽量少用；在 CPU 内部设置大量的寄存器，使大多数操作在速度很快的 CPU 内部进行；非常适合流水线操作，由于指令简单，并行执行就更容易实现。

为了大幅度提高处理器的速度，当前处理器中采用了指令及并行处理技术，如超标量，它是指采用多个处理部件，多条流水线并行执行。流水线组织是实现指令并行的基本技术，影响流水线连续流动的因素有数据相关性、转移相关性、功能部件冲突和中断系统。要发挥流水线的效率，还必须重点改进编译系统。在 RISC 设计中，对转移相关性一般采用延迟转移方法解决。

当前流行的中间件平台有 3 种：CORBA、J2EE 和 DNA2000

存储器的分区管理，按分区方式不同分为固定分区、可变分区和可重定位分区。固定分区是一种静态的分区方式，在系统中已经将主存划分为若干个分区，每个分区的大小可不等。但是存在一个问题：分区固定而作业大小不固定，导致系统中碎片过多。可变分区是一种动态分区方式，分区大小可变，可随着作业的大小改变。可重定位分区是可以有效的解决碎片问题的简单方法，好处就是可以移动已有的分区使之靠拢，得到连续的区域。最佳适应算法是每次都找到一个与用户请求的最接近的分区分配给用户；最差适应算法总是将用户的作业装到系统当前的最大分区中；首次适应算法是系统每次都从主存的低地址空间开始寻找能装用户作业的空闲分区；循环首次适应算法每次都从刚分配的空闲分区开始，寻找一个能满足用户需求的空闲分区。

RAID 存储器（廉价冗余磁盘阵列，Redundant Arrays of Inexpensive Disks）是一种经济的磁盘冗余阵列，采用智能控制器和多磁盘控制器以提高数据传输率。RAID 的基本思想是用多个小的磁盘存储器，通过合理的分布数据，支持多个磁盘同时进行访问，从而改善磁盘存储器的性能。目前，RAID 分六级：

RAID0 无冗余和无校验的数据分块，具有最高的 I/O 性能和最高的磁盘空间利用率，但系统的故障率高。

RAID1 磁盘镜像阵列，由磁盘对组成，每一个工作盘都有其对应的镜像盘，磁盘利用率只有 50% 具有最高的安全性。

RAID2 采用纠错汉明码的磁盘阵列，大量数据传输时 I/O 性能较高，但不利于小批量数据传输。

RAID3 和 RAID4 采用奇偶校验码的磁盘阵列，读数据很快，但因为写入数据时要计算校验位，速度较慢。

RAID5 无独立校验盘的奇偶校验码磁盘阵列，与 RAID4 相似，但没有独立的校验盘。I/O 请求可以并行处理。

带有热备份的系统称为双重系统。它是两子系统同时同步运行，当联机子系统出错时，它退出服务，由备份系统接替，因此只要有一个子系统能正常工作，整个系统仍能正常工作。

乔姆斯基把文法分成 4 种类型，即 0 型、1 型、2 型和 3 型。0 型文法也称短语文法，0 型文法的能力相当于图灵机，或者说任何 0 型语言都是递归可枚举的。1 型文法也称上下文有关文法，其能力相当于线形界限自动机。2 型文法也称上下文无关文法，其能力相当于非确定的下推自动机。3 型文法也称右线性文法，由于这种文法等价于正规式，所以也称为正规文法。3 型文法的能力相当于有限自动机。从文法描述语言的能力来说，0 型文法最强，3 型文法最弱。3 型文法必是 2 型文法。每一种正规文法都是上下文无关的，每一种上下文无关文法都是上下文有关的，每一种上下文有关文法都是 0 型文法。称 0 型文法产生的语言为 0 型语言。

系统开发和运行知识

软件的生存周期，主要有可行性分析与项目开发计划、需求分析、设计(概要设计和详细设计)、编码、测试、维护等活动。软件生存期模型用来描述这些阶段的具体实现。总体分为三种模型：第一个是文档驱动，第二个是需求驱动，第三个是风险驱动。

根据 GB8566-88 计算机软件开发规定，软件的开发和维护分为 8 个阶段，分为可行性研究和计划、需求分析、概要设计、详细设计、实现、组装测试、确认测试和使用维护。其中单元测试是在实现阶段完成的，组装测试的计划是在概要设计阶段制定的，确认测试的计划是在需求分析阶段制定的。

新老系统转换方式有三种：直接转换确定新系统运行无误时，立刻启用新系统，终止老系统。旧系统和新系统并行工作一段时间，再由新系统代替旧系统的策略称为并行转换。分段转换是指在新系统全部正式运行前，一部分一部分地代替旧系统的策略，它是两种转换方式的结合，既保证了可靠性，又节约费用。

需求分析的结果是软件需求规格说明书。需求分析主要是确定待开发软件的功能、性能、数据和界面等要求。具体说可有以下几点：1、确定软件系统的综合要求 2、分析软件系统的数据要求 3、导出系统的逻辑模型 4、修正项目开发计划 5、开发一个原型系统

概要设计的目标是建立软件的体系结构，完成全局数据结构设计，同时进行处理方式设计、运行配置设计、出错处理设计、故障恢复设计等。详细设计是对每一个模块的操作的控制流程和局部数据结构进行设计。

不完善的系统定义是软件项目失败的主要原因。

数据流图 DFD 的基本成分(图形如下)有四个，分别为加工、数据流、数据存储和外部实体(数据的源点/终点)。其它外部实体是指存在于软件系统之外的人员或组织，它指出系统所需的数据的源地和系统所产生的数据归宿地。

DFD 的信息流大体可分为交换流和事务流两种类型。



数据存储(文件)

在结构化分析方法中，数据字典是最重要的文档。对加工的描述是数据字典的组成内容之一，常用的加工描述方法有结构化语言、判定树和判定表。

结构化设计(Structured Design, SD)方法是一种面向数据流的设计方法，它可以与 SA 方法衔接。结构化设计方法的基本思想是将系统设计成由相对独立、功能单一的模块组成的结构。

Jackson 设计方法是一种典型的面向数据结构的设计方法。以数据结构作为设计的基础，适用于规模不大的数据处理系统。Jackson 分析方法是面向数据流的分析方法。

常见的软件生存周期模型有瀑布模型、演化模型、螺旋模型、喷泉模型。瀑布模型本质上是一种线性顺序模型，各阶段之间存在着严格的顺序性和依赖性，特别强调预先定义需求的重要性。瀑布模型属于整体开发模型，它规定在开始下一个阶段的工作之前，必须完成前一阶段的所有细节，缺乏灵活性，无法解决软件需求不明确或不准确的问题，突出的缺点是不适应用户需要的变动。与瀑布模型相适应的软件开发方法是结构化方法。

软件开发模型用于指导软件的开发。演化模型是在快速开发一个原型的基础上，逐步演化成最终的软件，演化模型特别适用于对软件需求缺乏准确认识的情况。螺旋模型综合了瀑布模型和演化模型的优点，并增加了风险分析。喷泉模型是一种以用户需求为动力，以对象作为驱动力的模型，适用于面向对象的开发方法，反映了该开发过程的迭代和无间隙特征。采用螺旋模型时，软件开发沿着螺旋线自内向外旋转，每转一圈都要对风险进行识别和分析，并采取相应的对策。螺旋线第一圈的开始点可能是一个概念项目。第二圈开始，一个新产品开发项目开始了，新产品的演化沿着螺旋线进行若干次迭代，一直运行到软件生命期结束。在开发一个系统时，如果用户对系统的目标不是很清楚，难以定义需求，这里最好使用原型法。

软件设计包括四个既独立又相互联系的活动，分别为体系结构设计、接口设计、数据结构和过程设计。系统中模块的连续性不仅意味着作用于系统的小变动将导致行为上的不变化，也意味着规格说明的小变动将影响到一小部分模块。

模块化的目的是使程序的结构清晰，容易阅读、理解、测试、修改。

模块的独立程序由两个定性标准度量：内聚和耦合。耦合衡量不同模块彼此间相互依赖的紧密程度。内聚衡量一个模块内部各个元素彼此结合的紧密程度。高内聚，低耦合是一个好的模块的标准，是模块设计的追求。

内聚按强度由低到高有以下几种类型：

- 1、偶然内聚，如果一个模块的各成分之间毫无关系，则称为偶然内聚。
- 2、逻辑内聚，内个逻辑上相关的功能被放在同一模块中，则称为逻辑内聚。
- 3、时间内聚，如果一个模块完成的功能必须在同一时间内执行，但这些功能只是因为时间因素关联在一直，则称为时间内聚。
- 4、过程内聚，如果一个模块内部的处理成分是相关的，而且这些处理必须以待定的次序执行，则称为过程内聚。
- 5、通信内聚，如果一个模块的所有成分都作同一数据集或生成同一数据集，则称为通信内聚。

6、顺序内聚，一个模块的各个成分和同一个功能密切相关，而且一个成分的输出作为另一个成分的输入。

7、功能内聚，模块的所有成分对于完成单一的功能都是必须的，则称为功能内聚。

耦合按**从强到弱**的顺序分为以下几种：

1、内容耦合，一个模块直接修改或作为另一个模块的数据，或者直接转入另一下模块时，就发生了内容耦合。此时，被修改的模块完全依赖于修改它的模块。

2、公共耦合，两个以上的模块共同引用一个全局数据项就称为公共耦合。

3、控制耦合，一个模块在界面上传递一个信号(如开关值、标志量等)控制另一个模块，接收信号的模块的动作根据信号值进行调整，称为控制耦合。

4、标记耦合，模块间通过参数传递复杂的内部数据结构，称为标记耦合。此数据结构的变化将使相关的模块发生变化。

5、数据耦合，模块间通过参数传递基本类型的数据，称为数据耦合。

6、非直接耦合，模块间没有信息传递时，属于非直接耦合。

如果模块间必须存在耦合，就尽量使用数据耦合，少用控制耦合，限制公共耦合的范围，坚决避免使用内容耦合。

模块设计的原则是对修改封闭，对扩展开放。如果一个模块可以作为一个独立体被其它程序调用，则称模块具有封闭性。如果一个模块可以被扩充，则称模块具有开放性。

一个模块直接调用其它模块的个数称为模块的扇出系数；反之，一个模块被其它模块调用时，直接调用它的模块个数称为模块的扇入系数。经验表明，一个设计成功的系统的平均扇入、扇出系数通常是 3 或 4，一般不应超过 7，否则出错的概率增大。

软件设计的主要任务是设计软件的结构、过程和模块，其中软件结构设计的主要任务是确定模块间的组成关系。

一个软件产品在交付使用前要经历以下五种测试：1、模块测试，也称单元测试，是针对每个模块单独进行的测试。2、子系统测试，把已通过单元测试的模块连接成为子系统来测试，着重检验模块间的接口。3、系统测试，把经过测试的子系统连接成为一个完整的系统来测试，主要检查系统是否实现了说明书中指定的功能，系统的动态性能是否符合要求。4、验收测试，是为了确认已开发的软件能否满足标准，是否合格。与系统测试不同的是，它是在客户的积极参与下进行的。5、平行运行，即新、旧两个系统同时运行，比较结果有什么不同。

软件测试实际上分成四步进行：1、单元测试，也称模块测试 2、组装测试，也称集成测试。通常组装测试有两种方法：非增量式集成测试和增量式集成测试。3、确认测试是，进一步检查软件的功能和性能是否与用户要求的一样。可靠性测试属于确认测试阶段的活动。4、系统测试，系统测试是将软件系统与硬件、外设和网络等其他因素相结合，对整个软件进行测试。其目的是通过与系统的需求相比较，发现所开发的系统与用户需求不符或矛盾的地方。常见的系统测试主要有以下内容：恢复测试、安全性测试、强度测试、性能测试、可靠性测试、安装测试。

白盒测试方法一般适用于单元测试。黑盒测试也称为功能测试。黑盒测试不能发现是否存在冗余代码。软件测试的目的是尽可能多地发现软件系统中的错误和缺陷。在进行单元测试时，常用的方法是采用白盒测试，辅之以黑盒测试。

Alpha 测试是由一个用户在开发者的场所来进行的，Beta 测试是由软件的最终用户在一个或多个用户场所来进行的，开发都通常不在现场。

软件的可维护性、可用性、可靠性构成了衡量软件质量的几个重要尺度。

在软件运行/维护阶段对软件产品进行的修改就是所谓的维护。软件维护活动所花费的工作占整个生存期工作量的 70%以上。在整个软件维护过程中，完善性维护所占比重最大，平均 50%左右。维护的类型有三种：改正性维护、适应性维护和完善性维护，软件维护的内容为：(1)纠正和修改软件中含有的错误、(2)因环境发生变化，软件需做相应的变更、(3)为扩充功能、提高性能而做的变更。软件的易维护性是指理解、改正、改进软件的难易度。通常影响软件易维护性的因素有可理解性、可修改性和可测试性。在软件开发过程中往往采取各种措施来提高软件的易维护性，如采用良好的编程风格有助于提高软件的易理解性。信息隐蔽原则有助于提高软件的易修改性。所谓维护管理主要指的是判定修改的合理性并审查修改的质量。在软件质量特性中，可靠性是指在规定的一段时间和条件下，与软件维持其性能水平的能力有关的一组属性。安全性是指防止对程序及数据的非授权访问的能力。为了减少维护工作的困难，可以考虑采取的措施：(1)切实加强维护管理 (2)在开发过程中采取有利于将来维护的措施。

软件能力成熟度模型 (CMM, Capability Maturity Model)描述和分析了软件过程能力的发展与改进的程序，确立了一个软件过程成熟程度的分级标准。在初始级，软件过程定义几乎处于无章可循的状态，软件产品的成功往往依赖于个人的努力和机遇。在可重复级，已建立了基本的项目管理过程，可对成本、进度和功能特性进行跟踪。在已定义级，用于软件管理与工程两方面的软件过程均已文档化、标准化，并形成了整个软件组织的标准软件过程。在已管理级，对软件过程和产品质量有详细的度量标准。在优化级，通过对来自过程、新概念和新技术等方面的各种有用信息的定量分析，能够不断地、持续地对过程进行改进，目标是达到一个持续改善的境界。

当在软件工程的环境中考虑风险时，主要基于 Charette 提出的三个概念：关心未来、关心变化、关心选择。项目风险关系着项目计划的成败，商业风险关系着软件的生存能力。在进行软件工程风险分析时，项目管理人员要进行四种风险评估活动：建立表示风险概率的尺度，描述风险引起的后果，估计风险影响的大小、确定风险估计的正确性。

软件危机主要表现在以下几个方面：

- 1、软件需求增长得不到满足。
- 2、软件生产高成本，价格昂贵。
- 3、软件生产进程无法控制。
- 4、软件需求定义不明确，易偏离用户需求。
- 5、软件质量不易保证。
- 6、软件可维护性差。

用来辅助软件开发、运行、维护、管理、支持等过程中的活动的软件称为软件开发工具，通常也称为 CASE 工具。(计算机辅助软件工具 Computer Aided Software Engineering) 集成型软件开发环境通常由工具集和环境集成机制组成，这种环境应具有开放性和可剪裁性。环境集成机制主要有数据集成机制、控制集成机制和界面集成机制。数据集成机制为环境中的工具提供统一的数据接口规范，控制集成机制为各工具之间的通信、切换、调度和协同工作提供支持，界面集成机制使得环境中的所有工具具有统一的界面风格和操作方式。

软件工程技术应遵循分解、一致性、确定性及抽象和信息隐蔽的原则。软件工程的三要素：方法、工具和过程。软件可行性研究包括技术可行性、经济可行性、法律可行性、社会可行性。

Gantt 图能清晰地描述每个任务从何时开始，到何时结束以及各个任务之间的并行性，但是它不能清晰地反映出各任务之间的依赖关系，难以确定整个项目的关键所在，也不能反映计划中有潜力的部分。

常用的软件成本估算模型有 Putnam 模型和 COCOMO 模型。项目管理工具用来辅助软件的项目管理活动。如成本估算模型，采用某种成本估算模型如 COCOMO 对项目的成本进行估算。它可以通过间接的测量来估算项目的规模大小，并描述总的项目特征。又如项目进度管理工具采用 PERT 制作项目的 PERT 也称网路图，并通过对 PERT 的分析，计算每个任务的最早开始及最迟开始时间，得到关键路径。此外还有一种采用 Gantt 图的进度管理工具，其目的主要也是对项目的进度进行管理。(PERT, Program Evaluation & Review Technique 计划评审技术) PERT 图不仅给出了每个任务的开始时间、结束时间和完成该任务所需的时间，还给出了任务之间的关系，但它不能反映任务之间的并行关系。

原型化方法较好地解决了需求难以精确定义的问题。其主要工作步骤为：

- 1、弄清用户的基本信息要求
- 2、开发初始原型系统
- 3、用原型系统完善用户的需求
- 4、根据用户的需求修改和完善原型系统

原型化方法的最主要特征是需求定义过程中有完整的生命周期。原型化方法要求软件开发人员能够快速获取用户需求。

在一项软件开发过程中，一般地说应该产生 14 种文件，其中管理人员主要使用的有项目开发计划、可行性研究报告、模块开发卷宗、开发进度月报和项目开发总结报告；开发人员主要使用的有项目开发计划、可行性研究报告、软件需求说明书、数据需求说明书、概要设计说明书、详细设计说明书、数据库设计说明书、测试计划和测试分析报告。维护人员主要使用的有设计说明书、调度分析报告和模块开发卷宗。

CRM 项目的实施可以分为 3 步，即应用业务集成、业务数据分析和决策执行。

在设计测试用例时，等价类划分是用得最多的一种黑盒测试方法。在黑盒测试方法中，等价类划分方法测试用例的步骤是：1、根据输入条件把数目极多的输入数据划分成若干个有效等价类和若干个无效等价类；2、设计一个测试用例，使其覆盖尽可能多的尚未被覆盖的有效等价类，重复这一步，直至所有的有效等价类均被覆盖。3、设计一个测试用例，使其覆盖 1 个尚未被覆盖的无效等价类，重复这一步，直至所有的无效等价类均被覆盖。

因果图方法是根据输入与输出之间的因果关系来设计测试用例的。

CVS 是一种版本控制工具。(Concurrent Version System 版本控制系统)

通常在软件的编码活动中，不需要用户的参与。

软件容错性的提高，有利于软件可行性的提高。

ISO/ICE9126 软件质量模型中第一层定义了六个质量特性，并主各质量特性定义了相应的质量子特性，其中易分析子特性属于软件的可维护性质量特性。

按照 ISO/IEC9126 软件质量度量模型中，一个软件可靠性的子特性包括成熟性、容错性和易恢复性。

用户手册详细描述软件的功能、性能和用户界面，以使用户了解如何使用软

件。

软件工程是将系统化的、规范的、可量化的方法应用于软件的开发、运行和维护的过程，它包括方法、工具和过程 3 个要素。ISO9000 是由 ISO/TC176 制定的关于质量管理和质量保证的国际标准。CMM 提供了一个框架，将软件过程改进的进化步骤组织成 5 个成熟度等级。除第 1 级外，每个等级都包含了实现该成熟度等级目标的若干关键过程域。在软件开发的各种资源中，人员是最重要的资源。软件的复杂性是固有的，它引起人员通信困难、开发费用超支、开发时间超时等问题。

操作系统知识

操作系统的基本特征是并发和共享。

进程是操作系统中可以并行工作的基本单位，也是核心调度及资源分配的最小单位，它由程序、数据和进程控制块 PCB 组成，它与程序的重要区别之一是：进程是有状态的，而程序没有，程序是静态的。在 SMP 系统中，操作系统还提供了线程机制，它是处理器分配的最小单位。

进程：是程序在并发环境中的执行过程。

线程：是进程内的一个执行单位，或说它是进程内的一个可调度的实体。

管程：是一种并发性的构造。管程定义了一个数据结构和能为并发进程所执行(在该数据结构上)的一组操作，这组操作能同步进程和改变管程中的数据。

进程控制块 PCB(Process Control Block)：操作系统中用来标志进程并对其进行管理和控制的数据结构。PCB是进程存在的惟一标志。

进程同步：进程间共同完成一项任务时直接发生相互作用的关系。

进程互斥：两个或两个以上的进程不能同时进入访问同一临界资源的临界区。实现进程互斥可以用加锁机制、信号量机制。

信号量：一个与队列有关的表示资源实体的整型变量，其值仅能由 P、V 操作来改变。信号量的物理意义：当信号量的值大于或等于 0 时，表示系统当前可用资源实体数目；当信号量的值小于 0 时，其绝对值表示系统中等待该信号量的进程数目。

操作系统的功能模块有：处理器(机)管理、存储管理、设备管理、文件管理、用户接口。

所谓死锁，是指多个进程因竞争资源而造成的一种僵局，若无外力作用，这些进程将永远不能再向前推进。死锁的原因主要有两点：竞争资源和进程推进顺序非法。针对产生死锁的必要条件，处理死锁的方法有：预防死锁、避免死锁、检测死锁、解除死锁。

产生死锁的四个必要条件：

1、互斥条件，进程对其要求的资源进行排他性控制，即一次只允许一个进程使用。

2、请求保持条件，零星地请求资源，即已获得部分资源，但又请求资源被阻塞。

3、不可剥夺条件，进程已获得的资源在未使用完之前，不能被剥夺，只能在使用完时由该进程自己释放。

4、环路条件，当发生死锁时，在进程资源有向图中必构成环路，其中每个进程占有了下一下进程申请的一个或多个资源。

进程具有五个基本特征：动态性、并发性、独立性、异步性、结构特征。并

发性是指多个进程实体同存于内存中，且能在一段时间内同时运行。异步性是指进程按各自独立的、不可预知的速度向前推进。

进程在运行中不断地改变其运行状态，通常，具有三种基本的状态：运行、就绪、阻塞。不少系统中，还增加了两种基本状态：新状态和终止状态。进程调度即处理机调度，它的主要功能是确定在什么时间将处理机分给哪个进程。

在文件存储设备管理中，有 3 类常用的空闲块管理方法，即位图矢量法、空闲块链表链接法和索引法。

安全状态序列：是指系统能按某种顺序来为每个进程分配其所需的资源，直至最大需求，使每个进程都可以顺序完成。若系统不存在这样一个安全序列，就称系统处于不安全状态。

虚拟存储管理系统的基础是程序的局部性理论，这个理论的基本含义是指程序执行时往往会不均匀地高度局部化地访问主存，这种局部性表现为时间局限性和空间局限性两种。根据这个理论，Denning 提出了工作集理论。工作集是进程运行时被频繁访问的页面集合。在进程运行时，如果它的工作集页面都在主存内，能够使用该进程有效地运行，否则会出现频繁的页面调入调出现象。根据局部性原理，一个作业在运行之前，没有必要把作业全部装入主存，而仅将那些当前要运行的页面或段先装入主存以启动运行，其余部分暂时留在磁盘上。如果所要访问的页(段)尚未调入主存，此时程序就利用操作系统所提供的请求面功能，将它们调入主存，以使进程能继续执行下去。从用户角度看，该系统所具有的主存容量将比实际主存容量大得多，人们把这样的存储器称为虚拟存储器。

虚拟存储器的组织方式有：分段存储组织、页式存储组织、段页式存储组织。虚拟存储器管理的置换策略有几种算法：

- 1、最优算法：选择淘汰不再使用或最远的将来才使用的页。
- 2、随机算法：随机的选择淘汰的页。
- 3、先进先出算法(FIFO)：选择淘汰主存驻留时间最长的页。
- 4、最近最少使用算法(LRU, Least Recently Used)：选择淘汰离当前时刻最近的一段时间使用得最少的页。

一台 PC 计算机系统启动时，首先执行的是 BIOS 引导程序，然后加载分区引导记录、配置系统，并执行分区引导记录。在设备管理中，虚拟设备的引入和实现是为了充分利用设备，提高系统效率，采用 spooling 技术，利用磁盘设备来模拟低速设备(输入机或打印机)的工作。

外部设备打印机适合于连接到字节多路通道。

对系统程序员透明的有：系列机各档不同的数据通路宽度、Cache 存储器和指令缓冲寄存器。对应用程序员透明的有：系列机各档不同的数据通路宽度、虚拟存储器、Cache 存储器、程序状态字和指令缓冲寄存器。

段页式存储组织综合了段式组织与页式组织的特点，主存被划分为定长的页，段页式系统中的虚地址形式是(段号、页号、位移)。系统为每个进程建立一个段表，为每个段建立一个页表。也就是说，先将程序按逻辑模块(如子程序、主程序和数据段等)分为若干段，再将每个段分为若干页。对于多道程序环境，每道程序有一个基号与其他程序相区分，每道程序可以有多个段，但只有一个段表，每个程序可以有多个页表。段页式存储体系中逻辑地址与物理地址的转换：首先由基号段号得到段表的地址，再访问段表得到页表的地址，再由页表得到物理块的地址，此时得到的地址是高 11 位的地址，因此需乘以 2 的 11 次方再加上页内地址，才得到真正的物理地址。

数据库技术

数据库按数据模型来分，可分为层次模型数据库、网状模型数据库和关系数据库 3 大类。

数据库系统由数据库、数据库管理系统和硬件等组成，数据库系统是在文件系统的基础上发展起来的。数据库系统由于能减少数据冗余，提高数据独立性，并集中检查数据完整性，由此获得广泛的应用。数据库提供给用户的接口是数据库语言，它具有数据定义、数据操作和数据检查功能，可独立使用，也可以嵌入宿主语言使用。SQL 语言已被国际标准化组织采纳为标准的关系数据语言。

事务的特性 (ACID)：原子性、一致性、隔离性、持久性。

并发操作带来的问题是数据的不一致性。数据的不一致性主要有三类：丢失更新、不可重复读和读“脏”数据。

计算机网络

早期的计算机网络是由终端—通信线路—终端组成的系统。资源共享是计算机网络的最主要功能。计算机网络的资源共享主要包括硬件资源共享、软件资源共享和数据资源共享。

ISO 所制定的 OSI/RM 是一个计算机网络体系结构标准。

ISO/OSI 的参考模型共有七层，由低层到高层分别为：物理层、数据链路层、网络层、传输层、会话层、表示层、应用层。各层的功能分别为：

物理层提供为建立、维护和拆除物理链路所需的机械、电气、功能和规程的特性。提供有关在传输介质上传输非结构的二进制流及物理链路故障检测指示。在这一层，数据还没有被组织，仅作为原始的位流或电气电压处理，单位是比特。

数据链路层负责在两个相邻结点间的线路上，通过校验、确认和反馈重发等方法将原始的不可靠的物理连接发行成无差错的数据通道，并进行流量控制。每一帧包括一定数量的数据和一些必要的控制信息。与物理层相似，数据链路层要负责建立、维护和释放数据链路的连接。在传送数据时，如果接收点检测到所传数据中有差错，就要通知对方重发这一帧。

网络层为传输层实体提供端到端的交换网络数据传送功能，包括通信子网中从源到目标路径的选择。提供数据报服务和虚电路服务两类典型的数据分组传送服务方式。

运输层也称传输层，为会话层实体提供透明、可靠的数据传输服务，保证端到端的数据完整性。

会话层的作用是允许在不同的主机上的各种进程之间进行会话。

表示层为应用层提供共同需要的数据或信息语法表示交换。数据的压缩、解压缩、加密和解密等工作都由表示层负责。

应用层提供 OSI 用户服务，是用户访问开放式系统互联模型的界面，它向用户提供各种直接的应用服务，如文件传送、电子邮件、虚拟终端、远程数据库访问等。

整个模型如下图所示：

应用层
表示层
会话层
传输层
网络层
数据链路层
物理层

以太网(Ethernet)遵循的是美国电气电子工程师协会 IEEE 制定的 802.3 标准，物理媒体可以采用双绞线、细同轴电缆、粗同轴电缆或光纤。采用粗同轴电缆的标准称为 10Base-5。10Base-5 标准规定每段粗同轴电缆的长度不能超过 500 米，超过上述长度时，段与段之间要用重发器相连。重发器的相连则是在物理层的连接。重发器不能无限制地使用。若总长度超过了 2500 米，就一定要分成两个网，网之间再用网桥来相连，也就是在 ISO 的开放式系统互联(OSI)七层参考模型中数据链路层的相连。将两个网络在网络层连接的设备通常称为路由器，在运输层以上高层连接的网际互连设备则称为网关。有时也用网关来泛指网际互连设备。

路由器是网络互联设备，用于连接多个逻辑上分开的网络，可以实现点对点、多点之间的通信。

在 Windows 操作系统中，采用 Tracert 命令来测试到达目标所经过的路由器数目及 IP 地址。

在默认情况下，客户机采用最先到达的 DHCP 服务器分配的 IP 地址。

某种中继设备提供运输层及运输层以上各层之间的协议转换，这种中继设备是网关。从 OSI 协议层次来看，用以实现不同网络间地址翻译、协议转换的数据格式转换等功能的路由器属于网络层范畴。当采用数据服务时，负责端到端的流量控制的是主机(端系统)。路由器的主要功能有：1、在网络间转发发送到网络的报文 2、选择最合理的路由 3、多协议的路由器可以连接使用不可同通信协议的网络段，作为不同通信协议网络段通信连接的平台。

ISP 接入 Internet 的方式一般采用 3 种方式，分别为帧中继方式、专线方式与 ISDN 方式。用户接入的方式则主要采用仿真终端方式、拨号 IP 方式和局域网连接方式。

ATM 的协议的应用适配层将从其他网络接收到的数据进行重组，物理层实现比特的传输，数据适配层负责数据的封装。

X.25 协议属 OSI 规定了主机 DTE 和网络设备 DCE 之间的 3 个层次上的接口。物理层相当于 OSI 参考模型的第一层。链路层相当于 OSI 参考模型的每二层。分组层相当于 OSI 参考模型的第三层。

ISO 为运输层定义了 4 种的服务原语，由运输层服务用户产生的原语是请求原语和响应原语。

IEEE802 规范主要与 OSI 模型的物理层和数据链路层有关。

802.11 标准定义的分布式协调功能采用了 CSMA/CA 协议。

因为 ATM 用虚电路，即信元沿同一条路径走，所以信元一般不会失序。

整个报文的端到端传递是运输层的事情。

Ethernet 和 Token Ring 都使用包交换技术。

计算机网络具有良好的安全性，应当由真实性、可用性、完整性、实用性和

占有性、保密性这些指标组成。

计算机网络中对等实体间通信时必须遵循约定的标准协议。不同的协议有不同的功能，如：SNMP用于网络管理，WAP用于移动电话无线上网，PPP可用于家庭拨号上网，HDLC是一种面向比特的数据链路通信规程，OSPF是一种路由选择协议。

某公司服务器上存储了大量的数据，员工使用服务器前首先必须登录。为了保证安全，使用认证技术对登录系统的用户身份进行认证。为保证传输效率，使用3DES三重对称加密算法对传输的数据进行加密。

SNMP, Simple Network Management Protocol: 简单网络管理协议。

PPP, Point-to-Point Protocol: 点到点协议。家庭拨号上网就是通过 PPP 在用户端和运营商的接入服务器之间建立通信链路。PPP 也衍生出新的应用，典型的应用就是 ADSL。PPP 协议的简单完整使它得到了广泛的应用。

RIP, Routing Information Protocol, 路由信息协议。作用在于能够实现路由信息的交流。TCP/IP 是一项可进行路由的协议，即每个信息包都设有一个指定路由方向的特定地址。

WAP, Wireless Application Protocol, 是一种向移动终端提供互联网内容和先进增值服务的全球统一的开放式协议标准，是简化了的无线 Internet 协议。WAP 将 Internet 和移动电话技术结合起来，使随时随地访问丰富的互联网络资源成为现实。

OSPF, Open Shortest Path First, 开放最短路径优先协议。

HTTP, Hypertext Transfer Protocol, 超文本传输协议，是 WWW 服务程序所用的协议。

HDLC, High-Level Data-Link Control, 是可靠性高，高速传输的控制规程。相当于 OSI 基本参考模型的数据链路层部分的标准方式的一种。

RARP, Reverse Address Resolution Protocol, 反向地址转换协议。允许局域网的物理地址从网关服务器的 ARP 表或者缓存上请求其 IP 地址。RARP 可以使用于以太网、光纤分布式数据接口及令牌环 LANs。

ARP 协议的作用是由 IP 地址查找对应的 MAC 地址，ARP 报文封装在以太帧中传送。

Internet 是目前世界范围内最大的互联网。如此多的各种计算机之所以能通过 Internet 相互通信，是因为它们遵循了一套共同的 Internet 协议。这套协议的核心是 IP，在其上建立的无连接的传输层协议是 UDP，万维网 WWW 上超文本传输遵循 HTTP，电子邮件传输遵循 MTP 或 SMTP，Ethernet 与 Internet 连接时要用到 ARP。

目前常见的局域网类型包括：以太网 (Ethernet)、光纤分布式数据接口 (FDDI)、异步传输模式 (ATM)、令牌环网 (Token Ring)、交换网 (Switching) 等，它们在拓扑结构、传输介质、传输速率、数据格式等多方面都有许多不同，应用最广泛的是以太网——一种线结构的 LAN。

在 TCP/IP 网络中，为各种公共服务保留的端口号范围是 1-1023。

FTP 默认的数据端口号是 20，FTP 默认的控制端口号是 21，HTTP 默认的端口号是 80，Telnet 服务的 23 端口，SMTP 服务的 25 端口，DNS 服务的 53 端口和 SNMP 服务的 161 端口。

N-ISDN 只能提供 128kbit/s 的接入数据速率，ADSL 则是通过电话双绞线向端用户更高信息传输带宽的一种接入技术，而采用 B-ISDN 和电缆调制解调器也

可获得和后者同样数量级的接入带宽。第 3 代无线通信的 CDMA 可提供高达 2Mbit/S 的接入数据速率。光纤到户，即 FTTH 则是将来的发展方向。

C/S 又称 Client/Server 或客户/服务器模式。服务器通常采用高性能的 PC、工作站或小型机，并采用大型数据库系统。客户端需要安装专用的客户端软件。

B/S 是 Browser/Server 的缩写，客户机上只需要安装一个浏览器 (Browser)，如 Netscape 或 Internet Explorer。浏览器通过 Web Server 同数据库进行数据交互。

当网络用户通过网络与另一台主机 X 通信，发现响应太慢，可运行 Ping 程序，把一个分组发向主机 X，通过查看所返回的分组首部的时间戳，发现问题的所在，并采取相应的措施。对于一个大中型网络，需要有一个网络管理系统进行管理。当前流行的各类网管平台软件都支持 SNMP 协议。驻留在网络管理工作站上的网管平台软件可通过该协议软件调阅被管的网络结点内的管理信息库中的内容。目前常用的网络平台软件有 Sun 公司的 Net Manager，HP 公司的 Open View，IBM 公司的 Net View。

协议是一组约定的规则，它有助于外部设备之间的相互理解和正确进行通信。作为协议必须具备 3 个要素，语法、语义和时序。语法定义数据的表示形式；语义则能使数据管理所需的信息得到正确理解；时序则规定通信实体之间应答信号的相互间隔和顺序关系。

防火墙是一种位于计算机和它所连接的网络之间的软件，建立在内外网络边界上的过滤封锁机制。防火墙还可以关闭不使用的端口，并且能禁止特定端口的流出通信，防止来历不明入侵者的所有通信，封锁特洛伊木马。

主动攻击涉及修改数据流或创建错误的数据流，包括试图阻断或攻破保护机制、引入恶意代码、偷窃或篡改信息。它包括假冒、重放、修改信息和拒绝报务等方式。

一个功能完备的网络系统应该提供以下基本的安全服务功能：保密性、认证、数据完整性、防抵赖、访问控制。

多媒体知识

媒体 (Media) 包括其中两点含义。一是指信息的物理载体 (即存储和传递信息的实体)，二是指承载信息的载体，即信息的表现形式 (或者说传播形式)。

多媒体技术的主要特征有多样性、集成性和交互性。

用超文本方式组织和处理多媒体信息就是超媒体。互联网的 WWW 应用是超媒体技术的最好例子。

音量：声音的强弱程度，取决于声音波形的幅度，即取决于振幅的大小和强弱。

音调：人对声音频率的感常表现为音调的高低，取决于声波的基频。基频越低，给人的感觉越低沉，频率高则声音尖锐。

音色：由混入基音 (基波) 的泛音 (谐波) 所决定。一个声波上的越丰富，音色越好。

数字音频文件中对音频的采样频率为 44.1kHz

模拟声音在时间上是连续的，而数字表示的声音在时间上是断续的。

采样：把时间连续的模拟信号转换成时间离散、幅度连续的信号。每隔相等的一小段时间采样一次，其时间间隔称为取样周期，它的倒数称为采样频率。语音信号的采样频率一般为 8kHz，音乐信号的采样频率则应在 10kHz 以上。

量化后的样本是用二进制数来表示的，二进制数的位数的多少反映了度量声音波形幅度的精度，称为量化精度，也称为量化分辨率。

个人计算机中的数字声音有两种不同的表示方法，一种是波形声音，另一种是合成声音，例如 MIDI 音乐。

MIDI 是一种数字音乐的国际标准，MIDI 文件存储的是指令序列而不是波形声音。它的重要特色是占用的存储空间少。

未经压缩的数字音频数据传输率可按下式计算：

数据传输率 (b/s) = 采样频率 (Hz) × 量化位数 (b) × 声道数

存储量 = 采样频率 × 采样位数 × 声道数 × 时间 / 8 单位：字节数

波形声音经过数字化后所需占用的存储空间可用如下公式计算：

声音信号数据量 = 数据传输率 × 持续时间 / 8 (B)

声卡的功能有：1、录制(采集)数字声音文件 2、播放数字声音文件 3、编辑与合成声音文件 4、控制音源的音量 5、压缩和解压缩 6、文語转换(语音合成) 7、语音识别(注意：语音识别不是语音特征识别，语音特征识别依靠软件和算法实现) 8、提供 MIDI 功能。

彩色空间：RGB 彩色空间，计算机中的彩色图像一般都用 R、G、B 分量表示，CRT 彩色显示器选用 RGB 原色，是加色混色系统；CMY 彩色空间，根据三基色原理，油墨或颜料的三基色是青(Cyan)、品红(Magenta)和黄(Yellow)，彩色喷墨打印机等彩色图形印刷设备选用 CMY 三原色，是减色混色系统。YUV 彩色空间，亮度信号 Y、色差信号 U 和 V，电视图像一般都采用 Y、U、V 分量表示，如果只有 Y 分量而没有 U、V 分量，那么所表示的图像是黑白灰度图像。C、M、Y 分别是 R、G、B 的补色，当 RGB=111 时呈现白色，RGB=000 时呈黑色，CMY=000 时呈白色，CMY=111 时，它吸收了白色入射光中的红、绿、蓝三色，呈黑色，CMY 中的青色是 CMY=100，在 RGB 中等价 RGB=011。

图像的分辨率越高，图像的浓度越深，则数字化后的图像效果越逼真，图像数据量越大。对于视频来说，空间分辨率是指一帧图像包含的与每行显示的像素数之乘积，而时间分辨率是指每秒种显示或者传输的图像帧数。对于图像，因为没有时间的连续变化，所以它只有空间分辨率，没有时间分辨率。我们一般用水平和垂直方向上所能显示的像素数来表示分辨率，例如 1600*1200，这也是我们通常说的点阵大小。

图像的数据量 = 图像的总像素 × 图像浓度 / 8 (B)

例如一幅图像 640 × 480 的 256 图像，其文件大小为：640 × 480 × 8 / 8 ≈ 30KB。

若每个像素具有 8 位颜色的深度，则可表示 256 ($2^8 = 256$) 种不同的颜色，若某个图像具有 640 × 480 个像素点，其未压缩的原始数据需要占用 307200 字节的存储空间。(640 × 480 × 8 / 8 = 307200)

JPEG 文件采用一种有损压缩算法，其压缩比约为 5: 1 到 50: 1，目前使用最广泛的位图文件格式之一。

在 PC 机中，BMP 是常用的图像格式文件的后缀名。

我国电视制式 (PAL) 采用 625 行隔行扫描光栅，分两场扫描。场扫描频率为 50Hz。

视频压缩的目标是在尽可能保证视觉效果的前提下减少视频数据率。

帧内压缩也称为空间压缩，帧内压缩一般达不到很好的压缩率。帧间压缩也称为时间压缩。若视频图像序列中两帧相邻图像之间存在着极大的相关性，则这

种相关性称为时间冗余。

数据压缩技术可分为可逆与不可逆两大类，或者分为无损压缩和有损压缩两大类。常用的有损压缩编码技术有子带编码、模型编码和矢量化编码等。有损压缩适用于重构信号不一定非要和原始信号完全相同的场合。目前用得较多且技术成熟的无损编码技术有Huffman 编码、算术编码和词典编码(如著名的 LZW 算法)等。多数的无损压缩都采用 RLE 行程编码算法，这种算法特别适合于由计算机生成的图像，它们一般具有连续的色调。

对称性是压缩编码的一个关键特征。对称性意味着压缩和解压缩占用相同的计算处理能力和时间。对称算法适合实时压缩和传送视频，如视频会议应用。不对称或非对称意味着压缩时需要花费大量的处理能力和时间，而解压缩时则能较好地实时回放，即以不同的速度进行压缩和解压缩，如在电子出版和其他多媒体应用中，一般把视频预先处理好，尔后再播放，压缩 3 分钟的视频片段可能需要 10 多分钟的时间，而该片实时回放只需 3 分钟。

GIF 格式采用无损压缩方法中效率较高的 LZW 算法，主要用于图像文件的网络传输。目前因特网上大量采用的彩色动画文件多为这种 GIF 格式。

AVI 文件目前主要应用在三媒体光盘上，用来保存电影、电视等各种影像信息，有时也出现在因特网上，供用户下载、欣赏新影片的片段。

MPEG 文件格式是运动图像压缩算法的国际标准，它包括 MPEG 视频、MPEG 音频、MPEG 系统(视频、音频同步)3 个部分。MPEG 的平均压缩比为 50: 1，最高可达 200: 1，压缩效率非常高，同时图像和音响的质量也非常好，并且在 PC 机上有统一的标准格式，兼容性相当好。

MPEG-1 编码器输出视频的数据率大约为 1.5M B/s ，PAL 制式下其图像高度信号的分辨率为 352×288 ，帧速为 25 帧/秒。

国际标准 MPEG-2 采用了分层的编码体系，提供了 4 种技术，它们是空间可扩展性、时间可扩充性、信噪比可扩充性、数据分块。

节点、链和网络是超文本所包含的 3 个基本要素。

为保证用户在网络上边下载边观看视频信息，需要采用流媒体技术。一个流媒体系统一般由 3 部分组成，流媒体开发工具、流媒体服务器组件、流媒体播放器。目前应用比较广泛的流媒体系统主要有 Windows Media 系统、Real System 系统和 Quick Time 系统等。

交互式电视最常用的是节目间的交互，即点播电视技术系统(VOD)。典型的 VOD 系统主要由四部分组成：视频服务器、编码/路由器、用户请求计算机和记帐计算机、机顶盒。

虚拟现实 VR(Virtual Reality)技术的特点在于，运用计算机对现实世界进行全面仿真，创建与现实社会类似的环境，通过多种传感设备使用户“投入”到该环境中，实现用户与该环境直接进行自然交互。其关键技术包括基本模型构建、空间跟踪、声音定位、视频跟踪和视点感应等。

面向对象技术

面向对象技术中，对象是类的实例。对象有三种成份：消息、属性和方法(或操作)。

静态成员函数仅能访问静态的数据成员。静态数据成员的初始化必须在类外。

类具有实例化功能，包括实例生成和实例消除。

在收到消息时，对象要予以响应。不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态。多态有几种不同的形式，参数多态和包含多态称为通用多态，重载多态和强制多态称为特定多态。

面向对象技术中，对已有的实例的特征稍作改变就可生成其他的实例，这种方式称为继承。

继承分为单重继承和多重继承。单重继承是指一个子类只继承一个父类的特征。多重继承是指一个子类可以继承多个父类的特征。

封装是一种信息隐蔽技术，其目的是把定义与实现分离，保护数据不被对象的使用者直接存取。

重置是指在子类中保留既有父类的函数名，但使用不同类型的参数，即在面向对象编程语言中，允许同名、具有不同类型参数的函数共同存在。重置的基本思想是通过动态绑定机制的支持，使得子类在继承父类界面定义的前提下，用适合自己要求的实现去替换父类中的相应实现。

绑定是建立函数调用和函数本体的关联。动态绑定是面向对象程序设计语言中的一种机制，又称后期绑定。这种机制实现了方法的定义与具体的对象无关，而对方法的调用则可以关联于具体的对象。

在面向对象技术中，类属中一种参数多态机制。类属类可以看成是类的模版。一个类属类是关于一组类的一个特性抽象，它强调的是这些类的成员特征中与具体类型无关的那些部分，而用变元来表示与具体类型相关的那些部分。

类层次结构指导编译程序进行类的实例化过程，类的实例化过程是一种实例的合成过程。面向对象程序的设计和实现，首先要清楚类和类层次结构，程序运行时涉及到的具体对象不过是类的某一层次的实例化。单继承的类层次结构可用树来描述，多继承的类层次结构可用格来描述。

面向对象分析的第一步是确定问题域。

面向对象程序设计语言为面向对象实现阶段提供支持，针对接口编程，而不是针对实现编程。

面向对象的测试可分为四个层次，由低到高的顺序是：算法层 → 类层 → 模版层 → 系统层

面向对象的综合测试是由底向上的测试。面向对象测试的主要目标也是用尽可能低的测试成本和尽可能少的测试用例发现尽可能多的错误。一般来说，对面向对象软件的测试可分为下列 4 个层次进行：1、算法层，相当于传统软件测试中的单元测试 2、类层，可认为这是面向对象测试中所特有的模块测试 3、模版层，大体上相当于传统软件测试中的集成测试 4、系统层，把各个子系统组装成完整的面向对象软件系统，在组装过程中同时进行测试。

在面向对象的系统中，系统责任的良好分配原则是集中分配在少数控制类中。软件配置项是软件配置管理的对象，即软件工程过程中产生的信息项。软件测试计划中的白盒测试是属于编码阶段的测试。

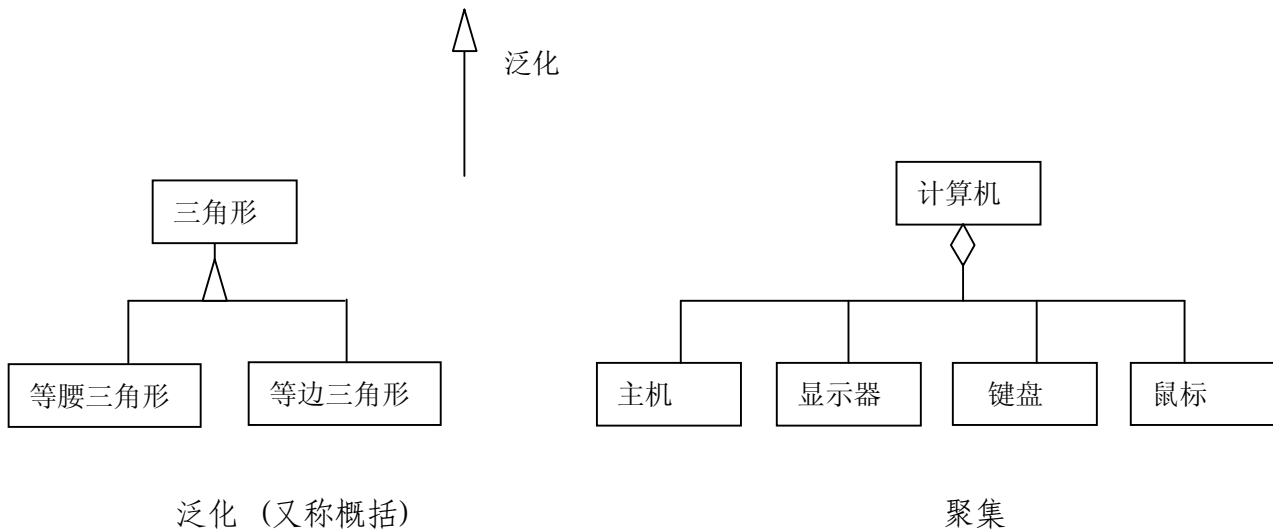
关联指明一个类的对象与另一个类的对象间的联系；两个类之间的关联表示了两个同等地位的类之间的结构关系，这两个类在概念上是同级别的。聚类是一种特殊的关联，它表示整体与部分的结构关系。

链表示实例对象间的物理或概念上的连接。实际上链是关联的一个实例。

泛化是一个类与它的一个或多个细化各类之间的关系，即一般与特殊的关系。被细化的类称为父类，每个细化的种类称为子类，子类可以继承父类的性质。

聚集是一种整体与部分的关系，在这种关系中表示整体的对象与表示部分的

对象关联。关联和聚类都表示实例之间的结构关系。对象间的“is-a”关系为聚集。



UML 由三个要素构成：UML 的基本构造块、支配这些构造块如何放置在一起的规则和运用于整个语言的一些公共机制。

UML 的词汇表包含了 3 种构造块：事物、关系和图。事物是对模型中最具有代表性的成分的抽象；关系把事物结合在一起；图聚集了相关的事物；UML 中有 4 种事物：结构事物、行为事物、分组事物和注释事物。

UML 中有 4 种关系：依赖、关联、泛化和实现。

依赖表示类之间的使用关系。

关联是一种结构关系，它描述了一组链，链是对象之间的连接。

类图给出系统的静态设计视图。用例图 (User Case Diagram) 展现了一组用例、参与者 (Actor) 以及它们之间的关系。对象图展现了一组对象以及它们之间的关系。用例图通常包括用例、参与者、扩展关系、包含关系。用例图用于对系统的静态用例视图进行建模。

在 UML 提供的图中，可以采用类图对逻辑数据库建模。状态图用于接口、类和协作的行为建模，并强调对象行为的事件顺序，若需要描述跨越多个用例的单个对象的行为，使用状态图是最为合适的。活动图用于系统的功能建模，并且强调对象间的控制流。

UML 的设计视图包含了类、接口和协作，其中，设计视图的静态方面由类图表现，动态方面由交互图、状态图和活动图表现。

OMT 定义了三种模型来描述系统。动态模型可以用状态图来表示。功能模块可以用数据流图来表示。对象模型为上述两种模型提供了基本的框架。

所谓构件，可以是一个类型、类或者其他的工作成品。构件包括使用案例、分析、设计、实现，还包括接口规格说明、子系统、属性类型，还包括其他的工作成品。对于构件，应当按可复用的要求进行设计、实现、打包、编写文档。构件应当是内聚的，并具有相当稳定的公开的接口。有的构件具有广泛的可复用性，可复用到众多种类的应用系统中，有的构件则只在有限的特定范围内被复用。

在实际应用中，一旦纠正了程序中的错误后，还应选择部分或全部原先已测试过的测试用例，对修改后的程序重新测试，这种测试称为回归测试。

UML 中用 5 种不同的视图来表示一个系统，这些视图从不同的侧面描述系统。

每一个视图由一组图形来定义。概述如下：用户模型视图，从用户的角度来表示系统。结构模型视图，从系统内部来看数据和功能性。行为模型视图，表示了系统动态和行为，还描述了用户模型视图和结构模型视图中所描述的各种结构元素之间的交互和协作。实现模型视图，将系统的结构和行为表达成为易于转换为实现的方式。环境模型视图，表示系统实现环境的结构和行为。空中多媒体电子出版物创作的主要过程可分为应用目标分析、脚本编写、设计框架、各种媒体数据准备、制作合成、测试。将基于内容的检索设计为多媒体数据库的检索引擎结构，在体系结构上划分为两个子系统：特征抽取子系统和查询子系统。

在统一建模语言 (UML) 中，用例图用于描述系统与外部系统及用户之间的交互。

数据结构及算法基础

广义表的深度定义是，广义表中括弧的层数。广义表的长度定义是，广义表中元素的个数。若广义表 $L = ((1, 2, 3))$ ，则 L 的长度和深度分别为 1 和 2。

若一个具有 n 个结点、 k 条边的非连通无向图是一个森林 ($n > k$)，则该森林必有 $n - k$ 棵树。

在一棵度为 3 的树中，若有 2 个度为 3 的结点，有 1 个度为 2 的结点，则有 6 个度为 0 的结点。

在二叉树的第 i 层至多有 2^{i-1} 个结点 ($i \geq 1$)

深度为 k 的二叉树至多有 $2^k - 1$ 个结点 ($k \geq 1$)

对任何一棵二叉树，叶子结点数 = 度为 2 的结点总数 + 1

具有 n 个结点的完全二叉树的深度为 $\lfloor \log_2 n \rfloor + 1$

设森林 F 中有 n 个非叶子结点，则由它转换得到的二叉树中右域为空的结点个数为 $n + 1$

哈夫曼树又称为最优树，是一类带权路径长度最短的树。在数据压缩编码的应用中，哈夫曼 (Huffman) 算法可以用来构造具有最优前缀码的二叉树，这是一种采用了贪心的算法。

将两个长度为 n 的递增有序表归并成一个长度为 $2n$ 的递增有序表，最少需要进行关键字比较 n 次。(如 (1, 2, 3, 4) 与 (5, 6, 7, 8) 关键字只需比较 4 次)

在链式结构线性表中插入元素的算法复杂度是 $O(n)$

n 个顶点 e 条边的图，采用邻接矩阵存储，深度优先和广度优先遍历时间复杂度均为 $O(n^2)$ ，采用邻接表法深度优先和广度优先遍历时间复杂度均为

$O(n + e)$

Prim 算法求最小生成树，时间复杂度为 $O(n^2)$

Kruskal 算法求最小生成树，时间复杂度为 $O(e \log e)$

任何一个借助“比较”进行排序的算法，在最坏情况下所需的比较次数至少

为 $|\log_2(n!)|$

无论是深度优先还是广度优先搜索遍历，图中 N 个结点都必须被访问一次。用邻接矩阵做图的存储结构时，其时间复杂度为 $O(n^2)$ 。若用邻接表作为存储结构，时间复杂度为 $O(n+e)$ 。深度优先搜索遍历和广度优先搜索遍历的空间复杂度都为 $O(n)$ 。

排序算法的时间复杂度比较：

从时间性能上说：

1、按平均的时间性能来说，有三类排序方法：

时间复杂度为 $O(n \log n)$ 的方法有：快速排序、堆排序和归排序，其中以快速排序为最好。

时间复杂度为 $O(n^2)$ 的有：直接插入排序、起泡排序和简单选择排序，其中以直接插入为最好，特别是对那些关键字近似有序的记录序列。

时间复杂度为 $O(n)$ 的只有基数排序。

2、当待排记录序列按关键字顺序有序时，直接插入排序和起泡排序能达到 $O(n)$ 的时间复杂度；而对于快速排序，就是最不好的情况，此时的时间性能蜕化为 $O(n^2)$ ，因此应尽量避免。

3、简单选择排序、堆排序和归并排序的时间性能不随记录序列中关键字的分布而改变。

从空间性能上说，也就是排序过程中所需的辅助空间大小：

1、所有的简单排序方法(包括直接插入、起泡和简单选择)和堆排序的空间复杂度都为 $O(1)$ 。

2、快速排序为 $O(\log n)$ ，为栈所需的辅助空间。

3、归并排序所需辅助空间最多，其空间复杂度为 $O(n)$ 。

4、基数排序空间复杂度为 $O(rd)$ 。

各种排序方法的比较：简单排序中直接插入最好，快速排序最快，当文件为正序时，直接插入和冒泡均最佳。

不同条件下，排序方法的选择：

1、若 n 较小(如 $n \leq 50$)，可采用直接插入或直接选择排序。当记录规模较小时，直接插入排序较好。否则因为直接选择排序移动的记录数少于直接插入，就选直接选择排序为宜。

2、若文件初始状态基本有序列(正序)，则应选用直接插入、冒泡或随机的

快速排序为宜。

3、若 n 较大，则就采用时间复杂度为 $O(n \log n)$ 的排序方法：快速排序、堆排序或归并排序。

快速排序是目前基于比较的内部排序中被认为是最好的方法，当待排序的关键字是随机分布时，快速排序的平均时间最短。堆排序所需的辅助空间少于快速排序，并且不会出现快速排序可能出现的最坏情况。这两个排序都是不稳定的。

若要求排序稳定，则可选用归并排序。

排序方法	最好时间	平均时间	最坏时间	辅助空间	是否稳定
选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
直接插入	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
冒泡排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
希尔排序		$O(n^{1/3})$	$O(n^2)$	$O(\log n)$	不稳定
快速排序	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(\log n)$	不稳定
堆排序	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	不稳定
归并排序	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	稳定

有 $\frac{n(n-1)}{2}$ 条边的无向图称为完全图，完全图是连通的。

一个含有 n 个顶点和 e 条边的简单无向图，在其邻接矩阵存储结构中共有 $n^2 - 2e$ 个零元素。

散列存储结构的特点是数据结构中元素的存储地址与其关键字之间存在某种映射关系。有效解决冲突的方法：线性探测法、双散列函数法、拉链法、建立一个公共溢出区。

HASH 查找分析：HASH 的查找长度与查找表的长度无关，只与装添因子有关。

$$\text{装添因子} = \frac{\text{表中添入的记录数}}{\text{哈希表的长度}}$$

设顺序存储的某线性表共有 123 个元素，按分块查找的要求等分为 3 块。若对索引表采用顺序查找方法来确定子块，且在确定的子块中也采用顺序查找方法，则在等概率的情况下，分块查找成功的平均查找长度为 23

$$\text{解析： } ASL = \frac{1}{2} \left(3 + \frac{123}{3} \right) = 23$$

平衡二叉树的查找，平均查找的时间复杂度是 $O(\log n)$

N 阶 B 树满足以下条件：

- 1、每个结点(除根外)包含 N 到 2N 个关键字
- 2、所有叶子节点都在同一层
- 3、B 树的所有子树也是一棵 B 树

在算法设计时，通常应考虑以下原则：首先所设计的算法必须是正确的，其次应有很好的可读性，还必须具有健壮性，最后应考虑所设计算法具有高效率和低存储量。

标准化、知识产权基础知识

标准是对重复性事物和概念所做的统一规定。

标准化的特征包括横向综合性、政策性和统一性。

标准化是指在经济、技术、科学及管理等社会实践中，对重复性事物和概念通过制订、发布和实施标准达到统一，以获得最佳秩序和最大社会效益。

已发布实施的现有标准，经过一段时间后，需要对其内容两次复查，以确保其有效性、先进性和适用性。

ISO 标准每 5 年复审一次，平均标龄为 4.92 年。我国的国家标准有效期一般为 5 年。

国际标准(International Standard)是指国际标准化组织(ISO)、国际电工委员会(IEC)所制定的标准。

国家标准的编号由国家标准的代号、标准发布顺序号和标准发布年代号(四位数组成)。表示方法：

强制性国家标准：GB × × × × × - × × × ×

推荐性国家标准：GB/T × × × × × - × × × ×

企业标准的编号：企业标准的编号由企业标准代号、标准发布顺序号和标准发布年代(四位数)组成，表示方法：Q/××× ××××-××××

计算机软件著作权的权利处软件开发完成之日起产生，保护期为 50 年。保护期满，除开发者身份权以外，其他权利终止。一旦计算机软件著作权超出保护期，软件就进入公有领域。

《反不正当竞争》中商业秘密定义是指“不为公众所知悉的、能为权利人带来经济利益、具有实用性并经权利人采取保密措施的技术信息和经营信息。”商业秘密的构成条件是：商业秘密必须具有未公开性，即不为公众所知悉；商业秘密必须具有实用性，即能为权利人带来经济利益；商业秘密必须具有保密性，即采取了保密措施。

一项发明或者实用新型获得专利权的实质条件为新颖性、创造性和实用性。

《中华人民共和国专利法》规定，发明专利权的保护期限为自申请日起 20 年；实用新型专利权和外观设计专利权的保护期限为自申请日起 10 年。

《中华人民共和国著作权法》规定，两人以上合作创作的作品，著作权由合作作者共同享有。我国著作权法对公民作品的发表权的保护期限是作者有生之年加死后 50 年。

计算机安全知识

加密和解密算法的设计通常需要满足三个条件：可逆性、密钥安全和数据安

全。传输加密使用的算法有两种：对称密钥加密和非对称密钥加密。非对称密钥加密算法中通信双方分别持有两个密钥，一个用来加密的公钥对外公开，另一个用来解密的私钥，由个人秘密保存。非对称加密的保密性比较好，但加密和解密花费时间长，速度慢，不适合对文件加密而只适用于对少量数据进行加密。

数字签名是一种网络安全技术，利用这种技术，接收者可以确定发送者的身份是否真实，同时发送者不能否认发送的消息，接收者也不能篡改接收的消息。数字签名算法很多，应用最为广泛的三种是：HASH 签名、DSS 签名和 RSA 签名。HASH 算法应用较广泛，但较容易攻破，存在伪造签名的可能。DSS 和 RSA 都采用了公钥，不存在 HASH 的局限性。RSA 既可以 用来加密数据，也可以用于身份认证。Kerberos 是一种分布式环境下的身份认证系统。为了防止重放攻击(Replay)，它使用了一次性的 Ticket 和时间戳。常用的数字证书格式有 PGP 证书和 X.509 证书。

Kerberos 为分布式系统提供的认证方案，能为每种服务提供可信任的第三方认证服务。TCP/IP 协议的安全机制先天不足，因此又提出了两个 TCP/IP 安全协议，它们是：Kerberos 和 SSL/SHTTP

计算机病毒是一种程序，它可以修改的程序，使得被修改的程序也具有这种特性。计算机病毒具有寄生性、隐蔽性、非法性、传染性和破坏性的特点。通常将微机病毒分成：系统引导型病毒、文件外壳型病毒、混合型病毒、目录型病毒和宏病毒。防止手段：人工防预、软件防预和管理防预。

(此笔记为本人 08 年上半年考前整理，不足和错误之处，敬请指正!)

邮箱: zhuliting@foxmail.com ~~QQ: 393259066~~

朱礼廷 于重邮