

Introduction to Computer System

Chapter 2 数字设备

2.1 A部分：设备基础

2.1.1 计算机

计算机的核心是多用途设备，接受输入、处理数据、存储数据，并根据一系列存储的指令产生输出。

输入：打印、提交或传送到计算机上的各种内容。

输出：计算机产生的结果。

输入和输出可以通过计算机中包含的组件或附加组件（例如键盘和打印机）来进行处理，这些组件可通过电缆或无限连接来连接到计算机

计算机通过执行计算来处理数据、修改文档和图片、绘制图形以及排序单词或数字列表。

计算过程有计算机的中央处理单元（Central Processing Unit, CPU）处理

- 大多数现代计算机的CPU是一个微处理器（microprocessor）。
 - 其是一个用于处理数据的可编程电子原件。

计算机存储数据以及用于处理数据的软件

- 大多数电脑在硬盘或闪存驱动器上部署了长期存储（storage）；
- 还在称为内存（memory）的设备上拥有临时存储区域。
-

计算机存储指令的能力：

- 计算机运行软件：
 - 计算机程序（简称为程序）：指导数字设备进行任务处理的指令。
 - 这些程序构成了软件。
 - 计算机在运行软件会通过运行指令来执行任务。
 - 第一批计算机通过以某种特定方式连接线路并被“编程”来执行特定的任务。
 - 即改变任务意味着重新布线。
 - 存储程序使计算任务的一系列指令被加载到内存中。

- 当需要使用多用途计算机设备时，这些指令可以很容易地被另一组指令所替代（这样就省去了重新布线的麻烦）。

计算机能运行的软件类型：

- 应用软件：帮助人们执行任务的一组计算机程序。
- 系统软件：帮助计算机进行自我监控（例：operating system，计算机内所有活动的主控制器）。
- 开发工具：用于创建应用程序软件、网站、操作系统和应用程序

2.1.2 电路和芯片

数字电子技术：

数字电子把数据位表示为在电路上传播的电信号。为了表示二进制数据，将高压信号用于1位，低压信号用于0位。

集成电路（Integrated Circuit, IC）是蚀刻在半导体薄片上的一组显微电子元件。

- 计算机芯片、微芯片、芯片通常指集成电路。
 - 当今数字设备的芯片内包含了数十亿个晶体管。
- 半导体：是处于导体和绝缘体之间的物质。
- 集成电路被封装在不同形状和尺寸的保护性载体中。

多个芯片的工作原理：大多数数字设备的电子组件都安装在一个电路板上，即主板（系统板，母版），其包括所有关键芯片并为其提供电路。

2.1.3 组件

外形：设备或部件。

系统单元：持有系统板的数字设备的一部分。

- 主板
- CPU
- 内存
- 电源
- 硬盘

组件系统的特点：大多数第一代个人电脑都是组件系统。

2.1.4 维护

2.2 B部分：设备选择

2.2.1 企业计算机

如今，计算机类型之间的区别不是微处理器，因为每台计算机都使用了一个或多个微处理器作为其CPU。

最强大的计算机：通常用于企业和政府机构，能够以非常快的速度为许多同时在线的用户提供服务并进行数据处理。

- 超级计算机：
 - 典型用途：破解密码，模拟全球天气系统、模拟核爆炸。
- 大型计算机：
 - 企业和政府机构通常使用大型机来提供大量集中的数据存储、处理和管理。
- 服务器：
 - 为连接到网络的计算机“提供”数据。当我们访问网站时，所获得的信息就是由服务器提供的。
 - 从技术上讲，任何计算机都可以配置为服务器。

2.2.2 个人计算机

个人计算机最初被称为微型计算机。

个人计算机的选择：

- 台式计算机
- 便携式计算机：
 - 笔记本电脑：大多数笔记本都使用与台式机相同的操作系统，但Chromebook除外。
 - Chromebook用户使用基于web的软件并将所有数据存储在云中。
 - 平板电脑
 - 智能手机：与基础手机不同，智能手机是可编程的。
 - 智能手机从基础手机和PDA（个人数字处理）演变而来。

2.2.3 利基设备

利基设备的共同点：包含一个微处理器。

- 树莓派：一个完整的计算机系统单元，可以连接键盘和屏幕以获得完整的计算机体验。
- 游戏机
- 便携式媒体播放器
- 智能手表
- 健身追踪器
- 智能家电

2.2.4 选择数字设备

兼容性：操作方式基本相同并使用相同软件的计算机被称为是兼容的。

2.3 C部分：处理器和内存

架构：数字设备内部工作原理。

2.3.1 微处理器

微处理器是用来处理指令的集成电路，其规格包括制造商的名称和芯片型号。

适用于台式机和笔记本电脑的处理器：

1971年英特尔推出了世界上第一个微处理器——4004。

如今的台式机和笔记本电脑中使用的处理器是兼容X86的。

- 基于x86技术的处理器被安装在所有台式机和笔记本电脑的内部；
- 基于ARM技术的处理器主导着平板电脑和智能手机。
 - 基于ARM的处理器能够高效率地使用能源。

2.3.2 处理器是如何工作的

微处理器指令集：微处理器通过硬连线执行一系列有限的活动，这种与编程活动的集合称为指令集。指令集是处理器系列独有的，相互之间不兼容。

一个指令集包含一组指令的集合，用于处理微处理器中的电路的行动。

微处理器不能直接理解编程语言，程序必须转换成与微处理器指令集相对应的机器语言，其是二进制的0和1字符串，可以通过数字设备来处理。

一个微处理器包含数英里的微型电路和数百万个卫星组件，其分为不同的操作单元，如ALU（算术逻辑单元）和控制单元。

ALU是微处理器中执行算术运算的部分，如加减运算。它还执行逻辑操作，如比较两个数字。ALU使用寄存器来保存正在处理的数据。

微处理器的控制单元负责读取每条指令。数据被加载到ALU的寄存器中，最后，控制单元给ALU发送信号并开始处理。

指令周期：计算机执行单一指令的过程。

指令周期的某些部分由微处理器的控制单元执行，其它部分由ALU执行，其步骤为：

1. 取指令
2. 翻译指令
3. 执行指令
4. 寻找下一个指令

用于特定程序的机器语言指令被保存在内存中。当程序开始时，第一个指令的内存地址被放置在微处理器控制单元的一部分内，称为指令指针。

控制单元进入内存地址》将该地址的数据复制到它的指令寄存器来获取指令》控制单元解释指令并执行该指令。

当ALU从控制单元获得启动信号时，ALU处理数据并将结果放置在累加器中，在累加器中，数据可以被发送到内存，或用于进一步的处理。当计算机完成指令时，控制单元会将指令指针增加到下一条指令的内存地址，指令周期又再次开始。

2.3.3 性能

微处理器的性能受到如下因素的影响：时钟速度、内核数量、处理技术、缓存大小、字长和指令集。

一项处理器的规格，如3.4GHz，代表了微处理器狮种，意味着执行指令的速度。

- 大多数计算机广告都是以千兆和指定微处理器的速度。GHz意味着每秒十亿次循环。
- 一个周期（即一个循环）是微处理器系统中最小的时间单位。处理器执行的每一个动作都由这些周期来衡量。
- 时钟速度只在比较同一芯片家族内的处理器时有效。

多核处理器

微处理器的核心由控制单元和ALU组成。一个包含多个处理单元电路的微处理器称为多核处理器。拥有更多的内核意味着更快的性能。

一些处理器“串行地”执行命令，即一次执行一条指令。在串行处理中，处理器必须在指令周期你完成所有步骤才开始执行下一个指令。

当处理器在完成前面指令之前开始执行指令时，它正在使用流水线处理。

并行处理一次执行多个指令，比如当今地多核微处理器。

CPU缓存（CPU cache）是一种特殊的高速存储器，允许微处理器比主板上其他地方地内存更快地访问数据。

CPU缓存的结构分为几个级别，一级缓存（L1）是最快的，而L2和L3稍慢一些，但仍然比访问主存或磁盘存储的速度快。缓存容量通常使用兆字节作为单位。

字长指的是微处理器一次可操作的位数，指示可控制单元和ALU使用的寄存器的大小。例如，具有64为字长的处理器的寄存器可以一次处理64位。

带有复杂指令集的微处理器使用CISC（复杂指令集计算机）技术，而带有有限的简单指令集的微处理器使用RISC（简单指令集计算机）技术。

RISC处理器的执行速度比CISC处理器的块，但是前者可能需要更多的简单指令来完成同样的任务。如今的台式机和笔记本电脑的处理器大多使用了CISC技术。许多手持设备的处理器，使用ARM（高级RISC机器）技术。

2.3.4 随机存取存储器

RAM（随机存储存取器）是数据、应用程序指令和操作系统的暂存区。RAM可以封装在一个芯片载体上，其被连接到主板上，或者位于插进主板的小电路板上。

智能手机描述“内存”容量的规格不是指RAM，而是一种更持久的存储器。

RAM负责保存等待处理的原始数据，以及处理该数据的程序指令。RAM也会保存处理的结果，直到其可以移动到一个更永久的位置，例如内部存储器、闪存驱动器或云存储。

除此之外，RAM还存储着控制计算机系统基本功能的操作系统指令。每次打开数字设备时，这些指令都被加载到RAM中。

RAM的工作原理：

在RAM中，被称为电容器的微观电子部件存储着用于表示数据的位。充电为1，放电为0.每个电容的RAM地址可以帮助计算机根据需要定位数据以进行处理。

大多数RAM是不稳定的，即需要电力来保存数据，一旦断电，所有存储在RAM上的数据立即消失。这类RAM在技术上属于动态RAM（DRAM），但通常称为RAM。

如果一个程序超出了其被分配的空间，操作系统就会使用磁盘或其它存储设备的空间作为虚拟内存来存储程序或数据文件。通过有选择地将RAM中的数据与虚拟内存中的数据交换，计算机理论上可以有效地获得几乎无限的内存容量。但过分依赖虚拟内存会降低性能，特别是其位于硬盘驱动器上时。

2.3.5 只读存储器（ROM）

ROM是一种存储电路，位于主板上的一个集成电路中。RAM是临时且易变的，而ROM是永久而稳定的。即使设备关闭，ROM的内容依然存在。

数字设备在被打开时会有等待时间，在等待时，ROM正在引导设备。

ROM包含一组名为引导加载程序的指令和数据。引导加载程序告诉一个数字设备如何启动。通常，引导加载程序执行自测试，以查明硬件是否能正常运行，并验证基本程序是否已经损坏，然后将操作系统加载到RAM中。

- 为何不在RAM中存储引导加载程序？
 - RAM没有存储数据的能力。当设备关闭再打开时，RAM是空白的，不包含任何用于执行微处理器的指令。
 - 而ROM即使电源关闭，也能保存数据。

改变ROM内容的过程有时候称为刷机，可能有以下几个原因去改变ROM和引导加载程序的内容：

- 修理：ROM内容被损坏；
- 用户修改：绕过某些限制（ios的刷机，安卓的rooting，可能使设备的证书无效。）
- 取证
- 更新：如果刷机失败，再ROM被替换之前，设备将不会被正常启动。

2.4 存储

现在的存储技术都是不完美的。

2.4.1 存储基础

存储：用于永久保存数据的数字设备组件。

存储是非易失性的，即使设备关机也能保留数据。

如今的数字设备可以使用本地存储和远程存储，但其不能互换。

本地存储：

- 可以直接连接到计算机、智能手机或设备上的存储设备。
- 包括：硬盘、CD、DVD、闪存驱动器、固态硬盘和存储卡。
- 在使用数字设备时，其本地存储是永久可用的。内置存储可以通过可移动存储来补充。
- 衡量标准：持久性、可靠性、速度、容量、成本。

数字设备上可用的本地存储容量取决于每个存储设备的容量和当前存储的数据量。可用的存储空间有时称为“空闲空间”。

存储技术的运作方式：

可以将计算机的存储设备看作直接通向RAM的管道。数据从存储设备复制到RAM中，等待处理。在数据被处理后，它会被暂时保存在RAM中，但其通常会被复制到存储介质中，以获得更持久的安全保护。

当数据被存储时，1和0必须转换为某种标记或信号，这是永久性的，但在必要时可以改变。

数据并不是在字面上写为1和0。相反，1和0必须转化为能保留在存储介质上的东西，如何转换取决于存储技术。

个人计算机中通常使用三种类型的存储技术：磁、光和固态。

2.4.2 磁存储技术

第一台个人计算机使用盒式磁带存储，不过软盘很快取而代之了。

磁存储通过磁化磁盘或磁盘表面上的微观粒子来表示数据。粒子保持其磁化方向直到方向改变，为数据提供永久但可修改的存储。

通过改变磁盘表面粒子的磁化方向，可以很容易地改变或删除磁存储的数据。磁存储的这个特性为编辑数据和重用包含不需要的数据的介质区域提供了高度的灵活性。

硬盘驱动器包含一个或多个磁盘及相关的读写头。

硬盘盘片是由铝或玻璃制成的扁平硬碟，表面附有磁性氧化铁粒子。

盘片作为主轴上的一个旋转单元，每分钟会进行成千上万次旋转。每个盘片都有一个读写头。磁盘驱动器上的读写磁头磁化粒子来写入数据，并通过感知粒子的极性来读取数据。

优点：

- 提供了大容量存储空间；
- 提供对文件的快速访问；
- 经济实惠。

缺点：

- 数据易受影响而改变；
- 磁介质会逐渐失去磁荷。
 - 存储在磁性媒体上的数据的可靠寿命大约为3年。
 - 建议每两年更换一次数据。

头部碰撞可能会损坏这个盘片并破坏其所包含的数据。

规格：容量、读取时间和速度。

存取时间：计算机在存储介质上定位数据并读取数据的平均时间。

数据传输速率：存储设备每秒从存储介质向RAM转移的数据量。

2.4.3 光存储技术

CD、DVD、蓝光技术被归类为光存储。

用光盘表面的亮点和暗点表示数据。

单个光驱通常可以用于处理CD、DVD和蓝光光盘。

光学技术分为三类：

- 只读（ROM），内容不能改变，寿命在百年左右。
- 可录制（R）：可被写入数据，一旦写入就不能改变。
- 可重写（RW）：数据可以写在光盘上并被修改，30年左右。

2.4.4 固态存储技术

固态存储（闪存）将数据存储在可擦写、可重写的电路中，而不是在旋转磁盘或流式磁带上，每个数据位都保存在可以打开或关闭的门式电路中。

打开或关闭门状电路的功耗非常低。一旦数据被存储，其就是非易失性的。

存储卡：临时存储，非易失性。

读卡器是一种在固态存储器上读写数据的设备。

USB1.0相当慢，USB2.0可以以800Mbps速率读取数据，USB3.0可以与硬盘驱动器的速度媲美。

2.4.5 云存储

本地存储器：内置于数字设备或可直接插入设备的存储器。

远程存储：位于可从网络访问的外部设备上。

云存储指的是一组用于传输、同步和管理存储在数据中心的高性能硬盘驱动器上的数据的技术。

2.4.6 备份

备份是指一个或多个文件的副本，用以应对原件被损坏的情况。

2.5 E部分：输入和输出

2.5.1 附加工具

外围设备：连接到计算机系统单元的打印机、显示设备、存储设备、鼠标和耳机等。

现代的外围设备包括小工具、附件和配件。

外围设备包括输入、输出和混合设备。

2.5.2 扩展端口

许多数字设备在系统单元中都有用于连接电缆和各种附加设备的端口，即扩展端口。

主板上的所有组件都通过电路进行连接。传送数据的主电路称为数据总线。

数据总线在微处理器和RAM之间运行的部分称为本地总线或内部总线。为了跟上微处理器的数据需求，是数据总线中最快的部分。

从RAM扩展到各种扩展端口的数据总线部分被称为扩展总线。

热插拔：在主机设备运行时断开外围设备连接的行为。

2.5.3 蓝牙

蓝牙：用于连接外围设备的通用无线技术。

2.5.4 设备驱动程序

设备驱动程序：帮助外围设备与主机设备建立通信的软件。

2.5.5 显示设备

GPU

一种称为集成图形的图形电路被内知道计算机的主板中；还有一种称为专用图形的图形电路，被安装在一个称为显卡（或视频卡）的小电路板上。

显卡包含一个图形处理单元（GPU）和特殊的视频内存。

组成总结

主板（芯片组），CPU，内存决定了系统的总体性能。

硬盘决定了系统存储性能。

显卡/显示器决定了系统显示性能。

向后兼容/向下兼容：新的能用在旧的上，分为软件和硬件。

ATX标准大板，PC机种尺寸最大的主板。

- 优点
 - 空间大，散热好；
 - 扩展槽多。



北桥一般内置于CPU中，负责CPU和内存间数据交换及与显卡的数据交换。

南桥控制各种外围设备。

芯片组和CPU要配对使用。



各部件中间的连接，称为总线。

内部各部件通过总线相连，外部设备通过接口电路和外部电路相连。

总线分为三种：

- 控制总线：用来控制各部分的时序；

- 地址总线：用来寻址；
- 数据总线：用来传输数据。

CPU与北桥之间，称为FSB（前端总线）。

CPU机器周期：

1. 读取操作，放入内存
2. 解码到处理器的控制单元
3. 携带指令到ALU执行
4. 存储到内存中

CPU的频率

- 主频：衡量性能的重要指标
- 时钟频率
- 外频
 - 一开始CPU与主板控制芯片、总线同频率；
 - 后来CPU与主板控制芯片频率不同
 - CPU时钟频率-主频
 - CPU与主控芯片-外频
- 前端总线频率
 - 早期，外频与此相等。
- 倍频系数
 - 从最初的2现在可达到40

$\text{CPU时钟频率} = \text{外频} * \text{倍频系数}$

- 一般倍频系数锁定，外频可以在主板设置；
- 超过规定的外频称为超频。
- 工程样品的CPU，倍频系数也可以修改；
- 超频会发热

FSB的速度可增加到外频的4倍。

CPU的速度并不单独取决于时钟频率，只有同系列才有比较意义。

摩尔定律：

提升CPU性能的技术：

- 多核心
- 超线程
- 多级缓存
- 64bit计算

多核的问题：

- 任务不能自动地并行处理
- 目前的多核CPU仅仅能同时运行多个程序，并没有大幅提升速度

缓存是一种高速内存，用以存储近期CPU运行所需的数据。

地址总线宽8位

程序的所有指令执行前都要调入RAM

屏幕上看到的所有信息都来自于内存

为何要有ROM？

- 很多地方我们需要软件不被修改：
 - bios
 - 智能设备
 - 单片机里的程序
- 可防止程序的非法修改

Chapter 6 软件

6.1 A部分：软件基础

我们从来都没有真正购买软件，只是获得使用它的许可证。

6.1.1 基本要素

软件（广义）：一个数字设备的所有非硬件组件，包括数据文件以及包含程序代码的文件。

在常见使用中，属于软件用于计算机程序，例如：操作系统和应用程序。

软件的种类：

- 系统软件：
 - 操作系统：Windows、Mac OS、Linux、Android、UNIX、Chrome；

- 设备驱动：用于数字设备彼此通信：打印机驱动、视频驱动；
- 实用程序：用于文件管理、安全、通信、备份、网络管理和系统监控；
- 开发软件：
 - 编程语言：用于编写程序：C、BASIC、Java、Fortan、C++、C#；
 - 脚本语言：用于编写脚本、创建网页和查询数据库：HTML、JavaScript、PHP、Python、Ruby、SQL；
 - 质量保证工具：用于测试软件：调试器、负载测试、安全测试。
- 应用软件：
 - 专业工具
 - 教育软件
 - 个人财务软件
 - 娱乐软件
 - 参考软件
 - 社交媒体软件
 - 商业软件
 - 生产力软件

软件并不是通用的。

大多数数字设备都预装了应用程序，这些应用是因为软件开发人员已经与设备供应商达成协议。

台式电脑和笔记本电脑需要带有文件管理器的操作系统，以便用户可以查看和操作数据文件。

- 一套基本的系统实用程序包括安全和防病毒软件、网络管理工具、浏览器以及用于鼠标、键盘、显示设备、打印机和其它外围设备的设备驱动程序。
- 应用软件：生产力软件

移动设备

- 操作系统
- 实用程序

6.12 分发

典型的软件应用的组件：

- 可执行文件
 - 包含一个可以在微处理器内逐步执行的计算机程序。

- 为PC设计的可执行文件通常具有.exe扩展名，而Mac的可执行文件具有.app扩展名。
- 或是许多可以一起工作的独立文件

准备分发软件时，通常会将分发文件压缩或打包成一个可以通过因特网轻松下载的单位。

软件的最佳来源：仅从可信来源下载软件。

- 付费软件支付时尽量不要直接使用银行卡

系统要求：系统要求指定了软件产品正常工作所需的操作系统和最低硬件需求。

软件升级：软件发行商会定期发布软件产品的新版本。

- 每个版本都带有版本号。

软件更新：一小部分程序代码，用于替换当前安装的软件的一部分。

服务包：通常适用于操作系统更新，指一组更新。

- 更新和服务包只在纠正问题并解决安全漏洞，并通过修订编号进行区分。

可用的付款方式：

- 一次性购买：
 - 优点：
 - 没有额外的费用；
 - 稳定，不会有突然的变化。
- 订阅：
 - 定期支付费用。
 - 优点：
 - 更新和升级通常包含在定价中
- 试用
- 免费增值模式：可以使用精简版，完整版需要付费。

6.13 软件许可证

软件许可证（许可证协议）：一个定义计算机程序使用方式的法定合同。（EULA）软件被许可的原因与版权有关。

版权授予软件产品作者复制、分发、出售和修改该作品的专有权，购买者一般是没有这种权利。

从法律角度看，有两类软件：

- 公共域软件：不受版权保护，但不能申请版权
- 专有软件：受版权保护，基于许可权，分为以下三类：
 - 商业软件：严格遵守版权法、可能允许软件一次安装在多个设备上。
 - 大多数

Chapter 3 网络基础 & Chapter 4 Web基础

通信系统：源主机->发送器->信道->接收器->目标主机

带宽：通信信道的传输能力

- 数字信道：bps
- 模拟信道：Hz
- 宽带、窄带

交换：从两个通信实体直接通信扩展为网络的基础。

- 集中、转接
- 提高效率、降低成本
- 交换方式
 - 线路交换：数据传输期间在源与目标之间建立一条由目标节点构成的专用物理线路
 - 线路建立、数据传输、数据拆除
 - 实时性强，适用于交互式通信，预分配带宽
 - 对突发性通信不适应，效率低
 - 分组交换
 - 将待发送的数据消息分为小块逐块发送到网络上
 - 不同线路的分组共用同一物理线路
 - 适用于突发数据的处理
 - 分组携带地址，由中间节点路由器决定如何转发
 - 路由器根据路由表转发分组

网络设备

- 路由器
 - 最重要的网络设备

- 负责转发网络中的数据
- 用于连接不同的网络
- 集线器、交换机、网桥、无线接入点

通信协议：网络中数据传输的规则，定义了：

- 数据传输的格式
- 数据传输的顺序
- 传输过程中对数据的操作

协议栈：网络中使用协议的集合。

- 网络中通常使用多个协议，一起完成数据传输任务

网络分类

- 个域网（PAN）
 - 10m之内
- 局域网（LAN）
 - 200米以内
- 城域网（MAN）
 - 80km以内
- 广域网（WAN）
 - 多由多个网络组成

因特网的基本结构

- 主干网由高性能路由器和高速光纤链路构成，由不同提供商维护，通过网络接入点连接
- 网络设备：路由器、通信设备、电子邮件服务器、Web服务器、DNS服务器
- NSP将各个ISP的网络连接起来

因特网协议、地址和域名

- TCP/IP协议
 - 负责因特网上消息传输的主协议组
 - TCP：传输控制协议
 - 将消息或文件分成数据段
 - IP：因特网协议
 - 给各种数据段加上地址
- IP地址

- 标注网络中的主机和路由器
- 32位二进制数 (IPv4)
- IPv6为128位
- 通常写成四个八位组
- 子网：同一子网内部所有节点的网络接口直接连通
 - 处于不同子网的计算机间数据由路由器转发
 - 处于同一子网的计算机间数据直接传输
- 一个IP地址分为两端：网络部分（高位）和主机段（低位）
- 子网掩码：确定IP地址中网络号长度
 - 32位，1标识网络部分，0标识主机部分
- IP地址 & 子网掩码 = 网络号（按位与）
 - 节点发送数据时，将目标IP地址与节点自身IP地址分别与子网掩码按位与
 - 一样就直接传输（同一子网）
 - 不一样传输给默认路由器（不同子网）
- 获取IP地址
 - 固定分配的静态IP地址
 - 服务器必须使用静态IP
 - 临时分配的动态IP地址
 - 大多数因特网用户动态的就行
 - 动态IP地址分配：DHCP（动态主机配置协议）
- 域名
 - 完全限定域名 或称 域名
 - 域名是网页地址和电子邮件地址的关键部分
 - scu.edu.com
 - 顶级域名
 - 因特网域名的最后部分
 - com、org、gov、edu、cn、ca
 - 每个域名都必须对应着一个或多个的IP地址
 - 域名系统（DNS）：存储域名和响应IP地址对应关系的数据库系统
 - 域名服务器：装有DNS系统，负责将域名映射为IP地址的服务器

ISP：互联网服务提供商

接入网络技术

- 连接

- 数据通过因特网传输的速度很快

- 往返延迟：从A到B再到A用的时间

- ping：因特网包探测器

- 检查两台计算机是否连通，检测往返时延

- 工作原理：向目标主机发送一个分组要求回复，根据收到时间确定时延

- 语法：ping 域名、ping IP地址

- Tracert：用于确定IP数据包访问目标所采取的路径

- 用数据分组的存在时间字段和错误消息来确定从一个主机到网络上其它主机的路由

- 路径上的每个路由器在转发包前至少将数据分组上的TTL减1，为0时路由器发送错误信息到主机

- 通过错误消息指导哪个路由器把包弄丢了（IP地址）

- 速度

- 上行速度：上传数据的速度

- 下行速度：将数据下载的速度

- !=：非对称因特网连接

- =：对称因特网连接

- 固定因特网连接

- 基于有线电话网

- 拨号连接：使用语音频带调制解调器和电话线在用户和其ISP之间进行数据传输的固定因特网连接

- 设备：语音频带调制解调器：将计算机信号转换成电话线上的音频模拟信号

- DSL（数字用户线路）：一种高速、数字化、持续在线的因特网接入技术，使用标准电话线系统

- 传输介质：标准铜制电话线缆

- 允许语音和数据信号同时传输，在电话公司的本地交换站分离，路由到电话系统和因特网ISP

- 变体：非对称（A-）、对称（S-）、高速（H-）.....

- 速度各有不同，受线路特性、与交换站距离影响

- 设备：

- DSL调制解调器：将计算机链接到电话线，完成信号转换

- DSL滤波器：防止语音频带信号干扰DSL信号
- 基于有线电视网
 - 在提供有线电视服务的基础设施上建立的宽带因特网接入
 - 电视信号和数据信号在同一电缆中传输
 - 设备：电缆调制解调器：将计算机信号转换成有线电视网络信号
 - 安全性：DOCSIS（有线电视接口数据规范）
- 基于卫星电视网
 - 使用个人蝶形卫星天线手法广播信号建立的持续在线的高速非对称因特网接入
 - 设备：
 - 蝶形卫星天线：转换频率、放大信号并传送到同步卫星
 - 卫星调制解调器：将计算机数据信号调制到能传输给蝶形卫星天线的频带
- 便携式和移动互联网接入
 - 因特网
 - 便携式因特网接入：方便地将服务从一个位置移动到另一个位置
 - 移动因特网接入：终端在移动时可以提供不间断的互联网连接
 - Wi-Fi热点
 - 便携式因特网接入方式，通常不提供可接受的移动因特网接入，用户只能在覆盖范围内连接
 - 蜂窝数据网络
 - WAP（无线接入协议）：为手机之类的手持设备提供因特网接入的通信协议

有线和无线技术

- 有线基础知识
 - 利用电缆连接网络设备
 - 快速、安全、容易配置
 - 移动性有限
- 以太网
 - 有线网络技术
 - 同时将数据包向所有的网络设备广播，但只有被寻址的设备才能接收
- 无线基础知识
 - 不通过电缆或电线，将数据从一个设备传输到另一个设备
 - 射频信号（无线电信号）
 - 微波
 - 红外线

- 可移动
- 慢，不安全
- 蓝牙
 - 一种可在两个设备间建立连接的短距离无线网络技术
 - 9.1m
- Wi-Fi
 - 无线点对点协议
 - 无线集中控制协议

网络安装与配置

- 安装概述
 - 插上路由器
 - 链接到计算机
 - 配置路由器
 - DHCP（动态主机配置协议）
 - 每个工作站（接入的设备）都必须有用于数据收发的唯一地址
 - 将路由器配置成DHCP服务器，自动分配地址
 - 无线加密
 - 为在无线设备间传输的数据加扰，只有在具有有效加密密钥的设备上才能解扰
 - 加密协议：WEP、WPA、PSK
 - 访问路由器安装实用程序
 - 设置新密码
 - 输入SSID
 - 激活WEP、WPA或PSK并创建加密密钥
 - 链接因特网接入设备
 - 安装无线工作站
- 文件共享
 - 访问：使用文件管理程序查看网络设备

Web技术

- Web（万维网）：通过HTTP协议在因特网上连接和访问的文档、图像、视频和声音文件的集合
 - 网页：一个或多个基于Web的文件按照类似于书页的格式现实出来的结果或输出
 - 网站：包含一系列经过组织和格式化的相关信息，用户能够使用浏览器访问这些信息
- Web服务器与浏览器
 - 客户端与服务器
 - 客户端发出请求
 - 服务器响应请求，提供服务
 - 客户端主机与服务器主机
 - Web服务器
 - 一个能够响应客户端程序发送的Web通信请求，返回网页的程序
 - 连接到因特网运行着Web服务器程序的计算机
 - Web客户端
 - 一种能够显示网页元素并处理页面间链接的客户端软件，也被称为Web浏览器
 - 运行Web浏览器的计算机
 - URL（统一资源定位符）
 - 每个网页都有一个唯一地址称为URL
- HTML（超文本标记语言）
 - HTML是指创建HTML文档需要遵循的一组规范，这些文档可以在浏览器中显示为网页
 - 标记语言：创建者可以通过插入特定的指令来标记文档的不同内容
 - 标签
 - 特定的指令
 - 区分文本各个组成部分的分界符
 - 用来说明浏览器如何显示文档
- Web浏览器
 - 如果浏览器本身不支持显示或播放某个网页元素所需的文件格式，就要下载必要的软件
 - 插件、加载项、播放器
 - 浏览器调用的用于处理额外文件格式的软件
 - 插件：用来扩展浏览器文件格式处理能力的软件
- Cookie（HTTP Cookie）
 - 由Web服务器生成后存储在本地文本文件的一块数据

- 使服务器能够记录用户的行为
 - HTTP是一种无状态协议，不记得你干了啥
- 工作原理：浏览器受到Web服务器发出的HTTP“设置Cookie”消息，其中包含在一些浏览器存储在客户计算机硬盘上的信息
- 生成Cookie的服务器在客户下次访问网页的时候可以向用户请求该cookie
- 搜索引擎
 - Web搜索引擎：通过形成简单的关键字查询来帮助人们定位Web上的信息的程序
 - 工作原理
 - 爬网程序：爬虫
 - 索引器：处理爬网程序收集来的信息，将其转换成存储在数据库中的关键字和URL
 - 数据库：存储网页的索引引用
 - 查询处理器：允许用户通过输入关键字访问数据库，然后产生一个网页列表，其中包含与查询相关的内容
 - 查询结果的排序
 - 链接流行度：度量从一个网页到其它网页的链接数量和质量的标准
- 电子商务：在计算机网络上以电子形式进行的交易
 - 电子商务模式
 - B2C：企业对消费者
 - C2C：消费者对消费者
 - B2B：企业对企业
 - B2G：企业对政府
- 电子邮件
 - 电子邮件消息：一种可通过计算机网络传送的文档
 - 电子邮件系统
 - 提供电子邮件服务的计算机和软件
 - 电子邮件服务器
 - 为用户起到中央邮局作用的计算机，能将受到的消息传送到用户邮箱中，通过因特网将发出的邮件路由到其它电子邮件服务器
 - 消息头：消息头+消息正文 = 电子邮件消息
 - 字段：收件人、发件人、主题、日期、抄送地址、优先级、追踪信息
 - 访问电子邮件

- 电子邮件(客户端) 软件
 - 本地电子邮件
 - Web电子邮件
- 本地电子邮件：通过电子邮件客户端收发邮件信息
 - 使用本地文件夹作为收发件箱（用户启动客户端并开始接受邮件时从服务器下载到本地）
 - 发出的邮件可以临时存储在本地或直接发出去
 - 允许在离线状态下读写邮件
 - 协议
 - 管理接受的邮件：POP3、IMAP
 - 处理发出的邮件：SMTP
- Web电子邮件
 - 通过浏览器访问的电子邮件服务
 - 无需本地收发件箱
 - 读写时必须在线
- 电子邮件附件：任何随电子邮件一同发送的文件
 - MIME
 - 最初电子邮件是以简单的ASCII文本格式存储的

物联网

- 物与物、人与物、人与人的互联
 - 核心与基础仍是互联网
 - 用户终端扩展
- 受限的终端
 - 内存
 - 计算能力
 - 节点能量
 - 可访问性
- 受限的接入网
 - 低速传输
 - 占空比限制：低网络利用率
 - 丢包率
 - 异步链路
 - 小分组

Chapter 7 计算机安全

- 恶意软件
 - 病毒：在计算机程序中插入的破坏计算机功能或者数据的代码，能影响计算机使用，能自我复制的一组计算机指令或者程序代码
 - 病毒引导模块
 - 病毒传染模块
 - 激活传染条件的判断部分
 - 传染功能的实施部分
 - 病毒破坏表现模块
 - 触发破坏表现条件的判断部分
 - 破坏表现功能的实施部分
- Rootkits
 - 用对应技术编写的软件
 - 将自身及指定的文件保护起来使无法被发现、修改或删除这些文件
- 蠕虫
 - 一种可以自我复制的代码，通过网络传播，通常无需认为干预就能传播
- 木马：基于C/S（客户/服务器）结构的远程控制程序，隐藏在合法程序中的恶意代码，要么执行恶意行为，要么给非授权访问系统的特权功能提供后门
- 防病毒软件
 - 特征码扫描：将扫描信息与病毒数据库对照
 - 启发式扫描：分析行为，与危险行为样式库进行对比
- 在线入侵：
 - 勒索软件
 - 僵尸网络
- 拦截

Chapter 9 数据库

数据库与文件系统的区别：数据库系统数据与对数据的操作（程序）是相对独立的，而文件系统是不独立的。

使用数据库系统的优势（比较于文件系统）：

- 收集和存储大量、多类型数据

- 提供数据完整性约束和数据操纵
- 支持数据的查询、统计
- 具有并发控制功能，支持分布式并行处理
- 具有数据恢复的能力
- 丰富的数据接口，便于应用程序访问
- 提供编程语言SQL，支持软件开发

关系数据库-基表-字段-关键字

- 关系数据库-关键字（建立数据之间的约束关系）
 - 主键：数据库表中的一个或几个字段，该字段的取值可以唯一确定表中的一个记录。
 - 主键保证数据不重复，避免数据的冗余。
 - 外键：其它表中的主键，在当前表中可以定义为外键，用于限定两个表之间的数据关联
 - 保证数据之间的一致性（对应关系），避免错误数据

数据库设计

- 数据需求分析：分析数据之间的关系
 - E-R图：实体集合关系图表示数据和数据之间的关系
 - 数据类图：数据对象及其数据对象之间的关系
- 创建数据库
 - 数据库组成：
 - 数据库
 - 数据库基表
 - 数据库视图
 - 数据库索引
 - 数据库触发器（程序）
 - 数据库存储过程（程序）
- 创建基表
- 定义基表的字段（属性）
- 定义基表的关键字
- 加载数据

Chapter 8 信息系统

信息系统基本概念：

- 作为大型软件：是一个可以进行信息的收集、组织、存储和交流的计算机系统
- 作为一个学科，研究人和组织收集、过滤、处理、创建、分发数据的方法、技术。工具等

信息系统的分类：

- 事务处理系统（TPS）：
 - 事务：一个交易过程，被记录和存储在计算机系统中。
 - 例：网购一个商品，预订一张机票，ATM上取一次现金等。
 - 事务处理系统（TPS）：提供收集、处理、存储、显示、修改或取消的事务处理功能的系统。
 - 例：工资管理系统、财务系统、机票预订系统、仓储管理系统、销售点信息系统、移动电话收费系统。
 - 批处理和在线处理
 - OLTP联机事务处理系统
- 管理信息系统（MIS，管理人员使用）
 - 很多都是多用户系统（TPS+MIS）
 - 例：四川大学学生综合教务系统、四川大学教师综合教务系统等
- 决策支持系统（DSS）
 - 帮助人们直接做决定
 - 执行信息系统（EIS）
 - DSS设计决策模型，进行决策查询
- 专家系统（ES）
 - 专家系统基于事实和规则，分析数据，产生推荐、诊断或决策。
 - 例：动物、植物识别专家系统、东海渔业资源评估专家系统

信息系统的应用领域：

- 垂直市场：为特定行业、企业设计的信息系统。
- 水平市场：为共同元素设计的信息系统。

SCM：供应链管理

CRM：客户关系管理

ERP：企业资源规划

信息系统的开发：

- 需求分析

- 软件设计
- 软件构造（构建数据库）
- 软件测试
- 软件测试
- 软件维护

PIECES框架：

性能、信息、经济、控制、效率、服务

Chapter 5 软件开发方法

软件和软件工程：

- 软件 = 程序 + 数据 + 文档 + 服务
- 软件工程：用系统的、科学的、可度量的方法开发软件
- 技术层次：
 - 1. 工具
 - 2. 方法
 - 3. 过程
 - 4. 质量

软件过程模型

- 软件开发过程：
 - 线性
 - 迭代
 - 并行

软件系统开发

- 需求分析：明确系统做什么
 - 主要活动：
 - 需求的获取
 - 分析需求（数据需求、功能需求、行为需求）
 - 分析建模
 - 编写软件需求规格说明书
 - 需求评审
 - 需求分析模型-UML模型

- 需求分析工具：UML
- UML用例图：描述系统的用户、外部系统、外部设备、系统的功能和功能之间的关系，以及外部环境与系统之间的交互关系。
- 软件需求规格说明书：
 - 对系统做什么进行描述
 - ~是需求分析的结果，也是软件设计的一句
 - 需求的变更需要进行控制和管理
- 系统设计：
 - 大型系统 通常由许多个子系统组成
 - 回答怎么做的问题，即如何实现需求的功能
 - 概要设计：数据设计、架构设计、界面设计
 - 详细设计：构件级设计
 - 设计建模
 - 编写设计文档
 - 概要设计：
 - 架构设计：设计软件的组成和组成之间的关系
 - 常见的软件架构
 - 集中式，主/从程序，管道，批处理
 - 分布式：客户端/服务器（C/S），浏览器/服务器（B/S）
 - 详细设计：
 - 需求分析
 - 架构设计
 - 构建层设计
 - 类的设计（算法设计、算法选择）
 - 设计工具：
 - 建模工具：URL建模语言
- 系统实现和测试
 - 选择语言
 - 选择方法
 - 选择环境
 - 编码和单元测试
 - 软件测试是尽可能发现缺陷，而不是证明软件没有缺陷
 - 测试方法：

- 白盒测试：根据程序的结构进行的测试（源代码）
 - 黑盒测试：根据程序的输入和输出的测试（执行软件）
 - 灰盒测试：混起来
- 测试阶段划分
 - 单元测试（开发人员完成）
 - 集成测试（测试人员完成）
 - 系统测试（测试人员完成）
 - 验收测试（用户参与）
- 单元测试：保证软件模块运行的可靠和正确。
- 集成测试：检查所有软件模块之间的协调工作。
- 系统测试
- 验收测试：
 - α 测试：工厂验收测试，在开发环境下的用户参与的系统测试。
 - β 测试：现场验收测试，在真实的使用环境下的用户独立进行的系统测试。
- 测试文档编写：
 - 测试计划；
 - 测试程序；
 - 测试用例；
 - 测试记录；
 - 测试结果报告/缺陷报告
- 系统的技术文档和用户文档
 - 系统开发文档：开发计划、软件需求规格说明书、设计文档、测试文档
 - 用户文档：操作使用手册、系统安装、维护手册
- 运行和维护
 - 交付和发布
 - 系统迁移：将系统从旧系统平滑过渡到新系统。
 - 系统转换策略：
 - 直接使用新系统；
 - 新旧系统同时使用，逐渐过渡到新系统；
 - 阶段性过渡：分阶段过渡到新系统；
 - 示范性过渡：改革开放（让一部分人先富起来）。
 - 系统的生命周期
 - 维护阶段的主要工作：

- 日常运行的维护：修改缺陷、改进功能和性能，适应性调整、增加新功能、系统升级
- 提供技术支持
- 服务质量保证，系统的运行性能保证

Chapter 10 编程

编程：

- 编程基础：
 - 指令：只会计算机工作的指示和命令。
 - 计算机程序：一组按部就班执行的命令，告诉计算机如何解决问题或执行任务。
 - 代码：构成计算机程序的指令
- 编程计划：弄清问题
 - 问题陈述：定义了要达到的某个结果或目标而必须被操作的一些元素。
 - 好的问题陈述的三个特点：
 - 能指定所有能够定义问题范围的假设
 - 能清楚地指定已知信息
 - 能指定问题什么时候算是已经解决
- 编写程序
- 程序测试和文档
 - 程序错误：语法错误、运行错误、逻辑错误。
 - 发现错误：阅读查找、调试器（GDB等）。
 - 注释文档：在程序代码中加入的对代码说明的文字
 - 不同的编程语言由特定的符号标记注释
- 发布程序的条件：
 - 性能
 - 可用性
 - 安全

形式化方法（formal method）

威胁分析

安全编程（走查、简化、过滤）

- 编程语言分类
 - 按级别分类

- 低级编程语言：机器语言
- 高级编程语言：人能看懂

○ 按代别分类

- 第一代编程语言：机器语言（与硬件直接相关、不兼容）
- 第二代编程语言：汇编语言（加入助记符）
- 第三代编程语言：PASCAL/BASIC/C/FORTAN（易记的命令字）
- 第四代编程语言：SQL（数据库查询语言，更像人话）
- 第五代编程语言：PROLO/Python（基于声明式程序设计范式）
 - 应用场景：逻辑推理、人工智能、自然语言理解、专家系统

○ 按范式分类

- 编程范式：程序员编程时的表达方式
- 分类：
 - 过程化编程范式：计算机解决问题或完成任务，是按照程序员编写的指令，以顺序步骤执行。
 - 面向对象编程范式
 - 计算机解决问题或完成任务，是按照程序员编写的对象，以及对象之间关系的指令，不一定按照顺序步骤执行。
 - 声明性编程范式
 - 计算机解决问题或完成任务，是按照程序员编写的事实和规则指令，不一定按照顺序步骤进行。

○ 工具集：

- SDK：软件开发工具包

● 过程化编程

- 算法：可以写下来并能实现的用以执行任务的一系列步骤。
 - 表述方法：结构化英语、伪代码、流程图
- 最好在哪里用过程化范式？
 - 最适用于可以通过逐步算法解决的问题
- 过程化范式的优缺点：
 - 优点：
 - 倾向于生成快速运行的程序并充分利用系统资源，便于理解。
 - 允许用同一方法解决许多类问题，易于修改。
 - 缺点：

- 不适合非结构化或算法非常复杂的问题。
- 一些问题不是线性的。
- 面向对象编程
 - 对象：代表抽象或真实世界实体的数据单元。
 - 类是具有相似特征的一组对象的模板
 - 一个类由属性和方法定义。
 - 类属性定义了一组对象的特征
 - 继承：将某个类的某些特征传递给其它类。
 - 生成具有集成属性的新类的过程将创建一个包含超类和子类的类层次。任何类都可以从超类处继承属性，子类是能从超类处继承属性的类。
 - 多态：多态是在子类中重新定义方法的能力
 - 优势：提供了易于扩展的面向对象程序，有助于简单地编程控制结构
 - 优点：
 - 更容易将问题的解决方案可视化
 - 可以提高效率，因为封装允许对象在各种不同的程序中进行适应性调整和反复利用。封装指隐藏对象及其方法内部的细节。
 - 缺点：
 - 运行时效率不高，面向对象程序要比面向过程程序需要更多的内存和处理资源。
- 声明式编程
 - 优点：
 - 为涉及词、概念和复杂逻辑的问题提供了高效的编程环境
 - 缺点：
 - 现在不用，没人气。

sql语法基础

sql语句对大小写不敏感

以分号结尾

- SELECT 从数据库中提取数据
- UPDATE 更新数据库中的数据
- DELETE 从数据库中删除数据
- INSERT INTO 向数据库中新插入数据
- CREATE DATABASE 创建新数据库

- ALTER DATABASE 修改数据库
- CREATE TABLE 创建新表
- ALTER TABLE 变更（改变）数据库表
- DROP TABLE 删除表
- CREATE INDEX 创建索引（搜索键）
- DROP INDEX 删除索引

SELECT语句

- 从表中获取所有列：SELECT * FROM XX（表的名称）；
- 获取对应列：SELECT xxx, xxx（column, 列名）FROM xx（表的名称）；
 - WHERE子句用于过滤那些满足指定条件的记录
 - 例：从websites表中获取国家为CN的网站：
 - SELECT * FROM Websites WHERE country='CN';
 - SQL以单引号来环绕文本值，如果是数值字段请不要使用引号。