

计算机专业大平台必修课



计算机科学导论

武汉大学 计算机学院



考试信息



- **形式**：开卷，可携带书、打印或手写等非电子的纸质材料，无需计算器，不能携带任何电子设备
- **范围**：全书
- **时间**：2018年12月23日（周日）18:30-20:30
- **地点**：计算机学院圆顶教室203



考试题型



- **简答题**：7道题，每题10分
- **论述题**：4道题中任选2题，每题15分



考试题型示范



- **简答题**
- 例1：计算机的发展趋势是什么？
- 例2：现代计算机能做什么，不能做什么？
- 例3：矢量图和位图图像，各自有何优缺点？



考试题型示范



- 简答题：计算存储一幅 400×300 的静态24位真彩图像需要多少字节？如果采用该真彩图像进行视频存储，计算20秒视频（每秒30帧）需要多少字节？

存储时，记录每一个像素点的RGB值，对真彩色来讲用3B，因此该图像需要的存储空间为：

$$400 \times 300 \times 24 / 8 = 360000 \text{ 字节}$$

计算20秒视频（每秒30帧）所需的存储空间为：

$$360000 \times 30 \times 20 = 216000000 \text{ 字节}$$



考试题型示范



- **论述题**
- 例1：操作系统？层次化存储体系？
- 例2：OSI模型与TCP/IP模型的比较？IP地址？子网掩码？
- 例3：具体问题求解？算法描述？算法评价？



考试题型示范



- 论述题：？

存储时，记录每一个像素点的**RGB**值，对真彩色来讲用**3B**，因此该图像需要的存储空间为：

$$400 \times 300 \times 24/8 = 360000 \text{ 字节}$$

计算**20**秒视频（每秒**30**帧）所需的存储空间为：

$$360000 \times 30 \times 20 = 216000000 \text{ 字节}$$



第1章

计算机与计算思维基础

武汉大学 计算机学院

第一章 计算机与计算思维



- 主题1、 关于计算思维的理解
- 主题2、 计算机理论模型--图灵机
- 主题3、 现代计算机结构--冯若依曼机
- 主题4、 计算机发展与学科体系



计算思维—定义



- 计算思维（Computational Thinking）
是运用计算机科学的基础概念进行问题求解、系统设计以及人类行为理解等涵盖计算机科学之广度的一系列思维活动。



计算思维—定义



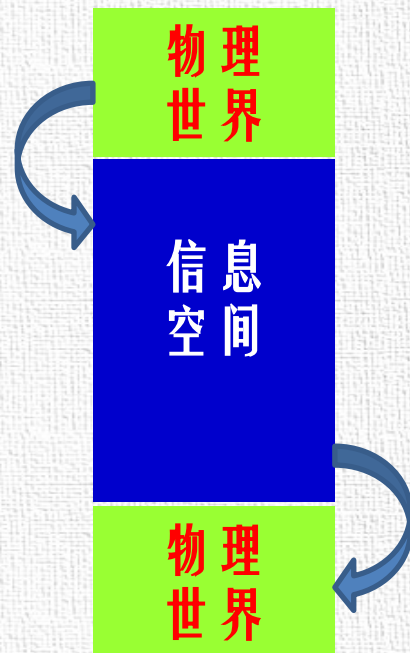
- 计算思维建立在计算过程的能力和限制之上，它是选择合适的方式去陈述一个问题，对一个问题的相关方面建模并用最有效的办法实现问题的求解，整个过程由人和机器协同配合执行。
- 计算方法和模型使我们敢于去处理那些原本无法由任何个人独自完成的问题求解和系统设计。



计算思维—表现



- **逻辑思维**：逻辑精准地描述计算过程
- **算法思维**：算法有效地构造计算过程
- **网络思维**：网络组合多个计算过程
- **系统思维**：计算过程在计算系统中运行
- **工程思维**：按生命周期运行与维护



计算思维—特征



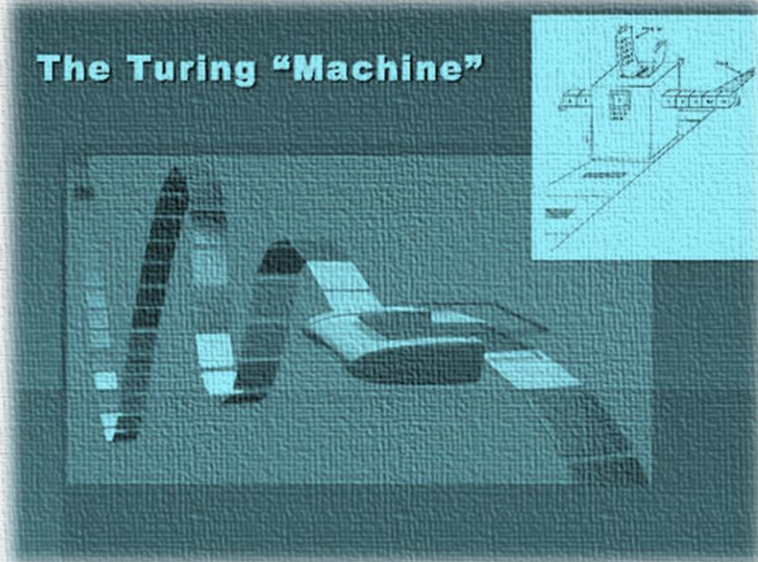
- 是概念化而不是程序化的。
- 是根本而不是刻板的技术。
- 是人而不是计算机的思维方式。
- 是数学和工程思维的互补与融合。
- 是思想而不是人造物。
- 是面向所有人和所有地方。



计算机的理论模型—图灵机

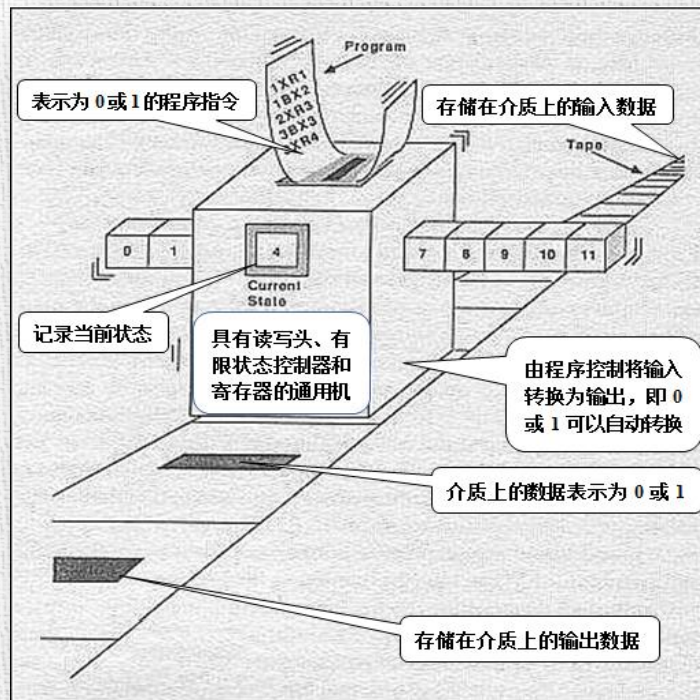


- 图灵机（Turing Machine），是一种抽象的计算思想模型，而不是一种具体的机器，但是这种思想模型可制造一种十分简单但运算能力极强的计算装置，用来计算所有能想象得到的可计算函数。
- 图灵认为这样的一台机器就能模拟人类所能进行的任何计算过程。



计算机的理论模型-图灵机

- 为了模拟人的这种运算过程，图灵构造出一台假想的机器，如图所示，该机器由以下几个部分组成：
 - 一条无限长的纸带TAPE。
 - 一个读写头HEAD。
 - 一套控制规则TABLE。
 - 一个状态寄存器。



图灵机蕴含的深邃意义



- 图灵机的产生蕴含的意义

- 一方面奠定了现代数字计算机的基础（要知道后来冯诺依曼就是根据图灵的设想才设计出第一台计算机的）。
- 另一方面，根据图灵机这一基本简洁的概念，我们还可以看到可计算的极限是什么。（强调必须是在有限输入的有限状态下完成有限的输出）



图灵机蕴含的深邃意义



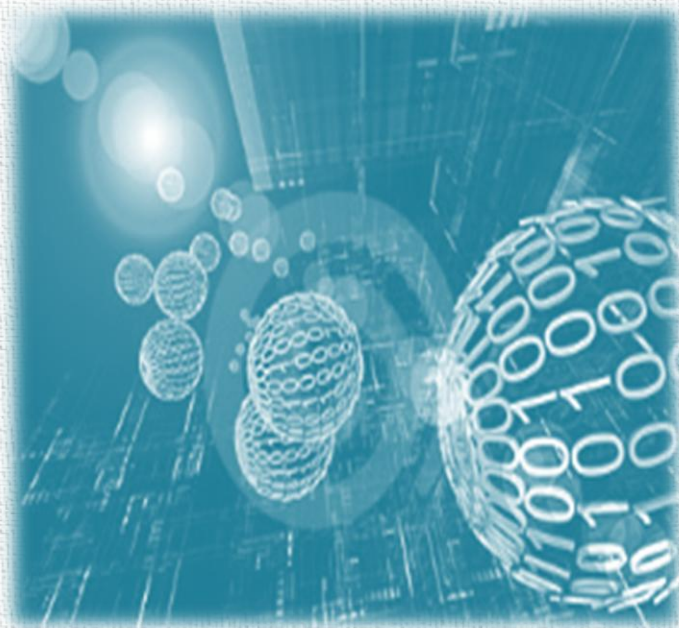
- 图灵给“可计算性”下了一个严格的数学定义，即：
 - 凡是能用计算算法解决的问题，也一定能用图灵机解决；
 - 凡是图灵机解决不了的问题，任何算法也解决不了。
- 图灵机解决的是一个可计算的问题，即：
 - 对于有限的输入数据；
 - 在有限步骤的算法指令的控制下；
 - 可以输出有限的结果。



计算机的理论奠基人—冯·诺依曼



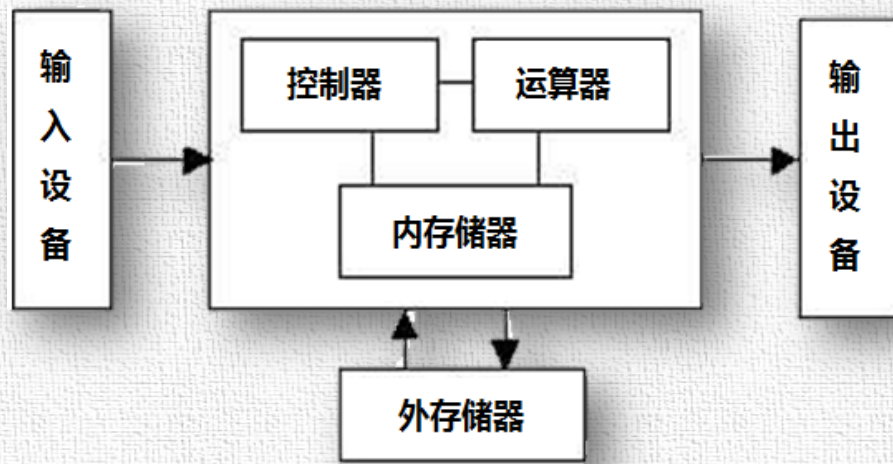
- 这两篇报告的综合设计思想，便是著名的“冯·诺依曼结构（Von Neumann Architecture）”。报告明确指出：
 - 采用二进制，不但数据采用二进制，指令也采用二进制；
 - 程序由指令组成并和数据一起存放在存储器中，机器按程序指定的逻辑顺序，把指令从存储器中读出来并逐条执行，从而自动完成程序描述的处理工作。（即，“存储程序”思想）



计算机的理论奠基人—冯·诺依曼



- 冯·诺依曼体系结构的计算机具有以下特点：
 - 必须有一个存储器，用于存储数据和程序；数据与程序以二进制形式存储；
 - 必须有一个控制器，用于实现程序的控制；
 - 必须有一个运算器，用于完成算术和逻辑运算；
 - 必须有输入和输出设备，用于进行人机通信。



现代计算机-定义



• 计算机的定义

- 计算机是由高科技电子元器件、线路和机械装置等部件或设备构成的，在计算机软件（程序及文档）的控制下，依照存储程序和程序控制的工作原理，能够高速、有效地完成人们指定的各种操作的自动化综合系统。



计算机能做什么？-应用



回眸与展望——趋势



- **由大到巨**
 - 追求高速度、高容量、高性能
- **由小到微**
 - 追求微型化，包括台式、便携式、笔记本式乃至掌上型，使用方便，价格低廉
- **网络化**
 - 整个网络是一个巨大的磁盘驱动器，可通过网络下载大多数乃至全部应用软件。PC使用者可不再为PC机的软硬件配置和文件的保存煞费苦心。因无论是数据还是应用软件，用户总能获得最新的版本。云存储、云计算、大数据和物联网是随着计算机网络的发展而应运而生的新技术
- **智能化**
 - 计算机的智能化就是让计算机来模拟人的感觉、行为、思维过程的机理，使计算机具备逻辑推理、学习等能力。也是第五代计算机要实现的目标。



第二章 信息表达与逻辑思维



- 主题1、 为什么是0与1
- 主题2、 布尔逻辑与逻辑思维
- 主题3、 0与1呈现的数值世界
- 主题4、 0与1呈现的文字世界
- 主题5、 0与1呈现的声色世界





2.1 为什么是0与1 ——二进制历史追溯



计算机选择了 0与1

- 容易被物理器件所实现
- 可靠性高
- 运算规则简单
- 与逻辑量相吻合



2.2 布尔逻辑与逻辑思维 — 计数制



- 数码

- 每种数制固定的符号集。例如，十进制有0、1、…、9 共10个数码，二进制采用0、1 共2 个数码。

- 基

- 计数制所使用的数码个数，称为“基”。十进制的基为10，逢十进一；二进制的基为2，逢二进一。

- 权

- 处于不同位置的数符所代表的值不同，与它所在的位置的权值有关。
- 例如，十进制数5555.555 可表示为：

$$5555.555 = 5 \times 10^3 + 5 \times 10^2 + 5 \times 10^1 + 5 \times 10^0 + 5 \times 10^{-1} + 5 \times 10^{-2} + 5 \times 10^{-3}$$



2.3 0与1的数值世界——整数表达



- 整数可以分为无符号整数和有符号整数两类。
 - 无符号整数的所有二进制位全部用来表示数值的大小。
 - 有符号整数用最高位表示数的正负号，而其他位表示数值的大小。
- 计算机中对有符号数的不同运算采用不同的编码方法，主要有原码、反码和补码3种。



2.3 0与1的数值世界——减法变加法



变
减
法
为
加
法

例如，在**五位**加法器中实现 **12-7** 的运算：

$$\because [-7]_{\text{原}} = 1\ 0\ 1\ 1\ 1 \quad [-7]_{\text{补}} = 1\ 1\ 0\ 0\ 1$$

$$[12]_{\text{补}} =$$

$$[-7]_{\text{补}} =$$

0	1	1	0	0
1	1	0	0	1
1	0	0	1	0
0	0	1	0	1

+

丢失

1

$$\therefore 12-7=12+25=5 \quad (\text{丢失模 } 32)$$

2.4 0与1的文字世界 —ASCII



- 为了信息交换中的统一性，人们已经建立了一些字符编码标准，目前国际上广泛使用的是ASCII码。
- 标准ASCII码的编码，用一个字节中的低7位对应一个字符的编码，可以为128种不同的符号编码。
- 计算机以一个字节存储ASCII码表中各字符的编码信息，最高位补0。



2.4 0与1的文字世界——汉字编码



- 在我国应用计算机，当然需要计算机处理汉字，中文数据也采用二进制编码来表示。汉字由于数量大，用8位二进制编码方式无法表示全部汉字，一般采用多个字节表示。
- 据统计，常用的汉字有四五千个。汉字字符集是一个很大的集合，要编码常用的汉字至少需要两个字节作为编码的形式。事实上，两个字节可以表示 $2^{16}=65536$ 种不同的符号，但考虑到汉字编码与其他国际通用编码（如ASCII码）的关系，我国国家标准局采用了加以修正的两字节汉字编码方案，即只用了两个字节的低7位。这个方案可以容纳 $2^{14}=16384$ 种不同的符号。



2.4 0与1的文字世界 --Unicode编码



Unicode是一个很大的集合，现在的规模可以容纳100多万个符号。每个符号的编码都不一样，比如，U+0639表示阿拉伯字母Ain，U+0041表示英语的大写字母A，U+4E00表示汉字“一”。具体的符号对应表，可以查询 unicode.org，或者专门的汉字对应表。Unicode 16编码里面已经包含了GB18030里面的所有汉字（27484个字），Unicode标准准备把康熙字典的所有汉字放入到Unicode 32bit编码中。



2.5 0与1的声色世界——音频数字化



2、声音数字化过程

- 采样：每隔一定的时间测量一次声音信号的幅值，把时间连续的模拟信号转换成每隔一定时间间隔的信号样本值序列。
- 量化：用有限个幅度近似表示原来在时间上连续变化的幅度值，把模拟信号的连续幅度变为有限数量、有一定时间间隔的离散值。
- 编码：经过采样和量化以后的数字信号以一定的方式进行编码，形成计算机内部运行的数据。经过编码后的声音信号就是数字音频信号。



2.5 0与1的声色世界——音频数字化



3、数字音频的技术指标

- (1) 采样频率
- (2) 量化位数
- (3) 声道数
- (4) 编码算法



2.5 0与1的声色世界——图形图像



1、图像的种类

- 矢量图
- 位图



(a) 矢量图局部放大



(b) 位图局部放大



第三章 计算机系统与系统思维



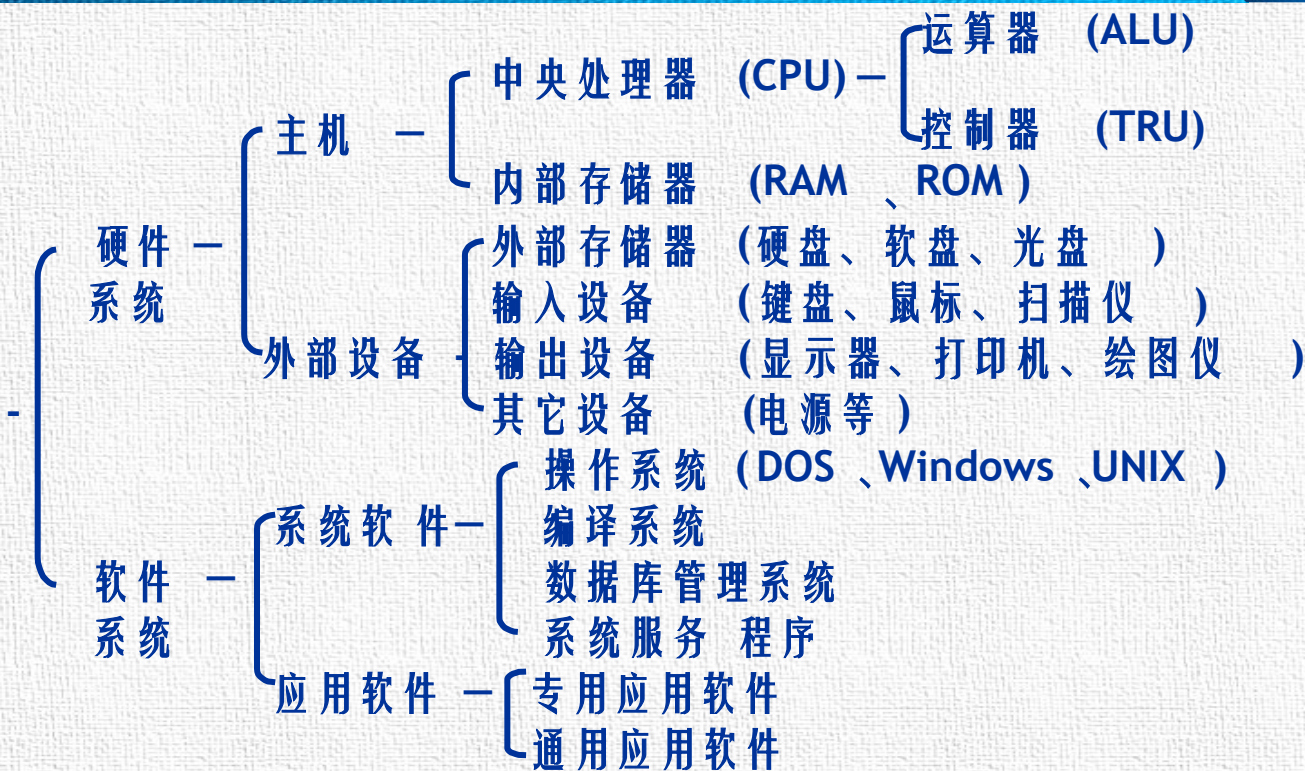
- 主题1、 计算机组成与系统思维
- 主题2、 计算机躯体——硬件系统
- 主题3、 层次化信息存储体系
- 主题4、 计算机灵魂——软件系统
- 主题5、 数据库管理系统



3.1 计算机组成



计算机系统



3.1 计算机组成与系统思维

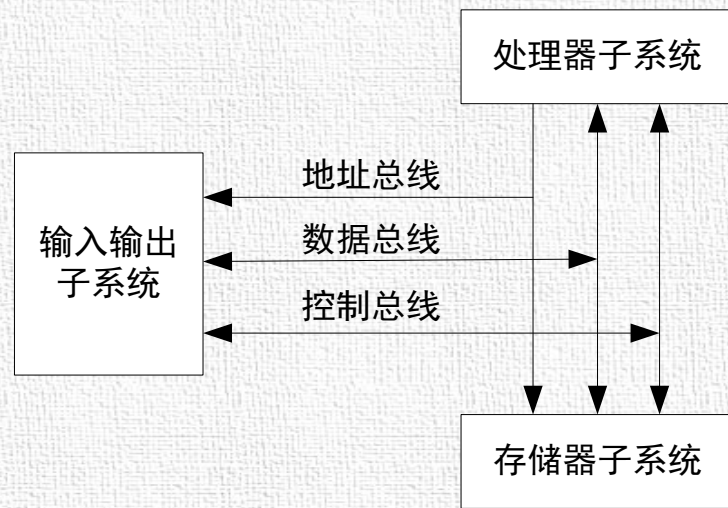


- 要点1：通过**抽象**，将系统**模块化**，**无缝**执行计算过程
- 要点2：**一套**方法支持**万千**应用
- 要点3：应对**复杂性**挑战



3.2 硬件系统组成

- 按照功能划分，可以将计算机硬件系统划分为四个子系统
 - 处理器子系统
 - 存储器子系统
 - 输入/输出子系统
 - 三种类型的总线组成的总线子系统，



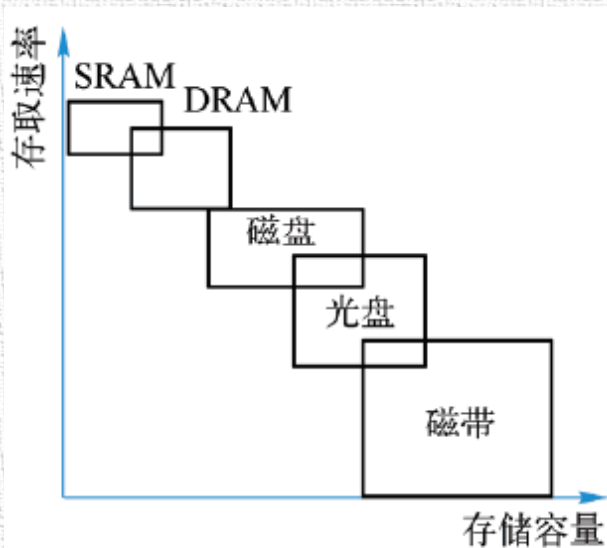
3.2 硬件系统组成 —— 存储



- 存储器分为两大类：内存储器和外存储器
- 内存储器也称主存，它直接与CPU相连接，存储容量小，速度快，用来存放当前运行的程序的指令和数据，直接与CPU交换信息
- 外存储器是内存的扩充；外存容量要比内存大得多，但它存取信息的速度比内存慢。通常外存不和计算机内其它装置交换数据，只和内存交换数据
- 常用的外存储器有磁带、磁盘和光盘等；外存用来永久存放“暂时不用”的程序和数据。



3.3 信息存储体系



(b) 存储器容量-速度系谱

- **主存储器**
 - CPU能够直接访问的存储器，用以存放当前运行的程序及其数据，也称为内存储器。
- **辅助存储器**
 - 为解决主存储器容量不足而设置的存储器，用以存放当前不参加运行的程序和数据。
- **高速缓冲存储器**
 - 一种介于主存与CPU之间，用于解决CPU与主存储器间速度匹配问题的高速小容量存储器。其存取速度接近CPU的工作速度，用来存放当前CPU经常使用到的指令和数据。
- **海量后备存储器**
 - 磁带存储器和光盘存储器的容量大、速度慢，主要用于信息的备份和脱机存档，因此被用作海量后备存储器



3.3 信息存储体系——分级存储



- 把几种存储技术结合起来、互相兼容，将各种存储器组织成统一结构。把不同的存取速度、容量、价格、性能等指标折中，按不同的层次存放程序和数据，按不同的存取速度、容量、价格、性能等指标折中，按不同的层次存放程序和数据。

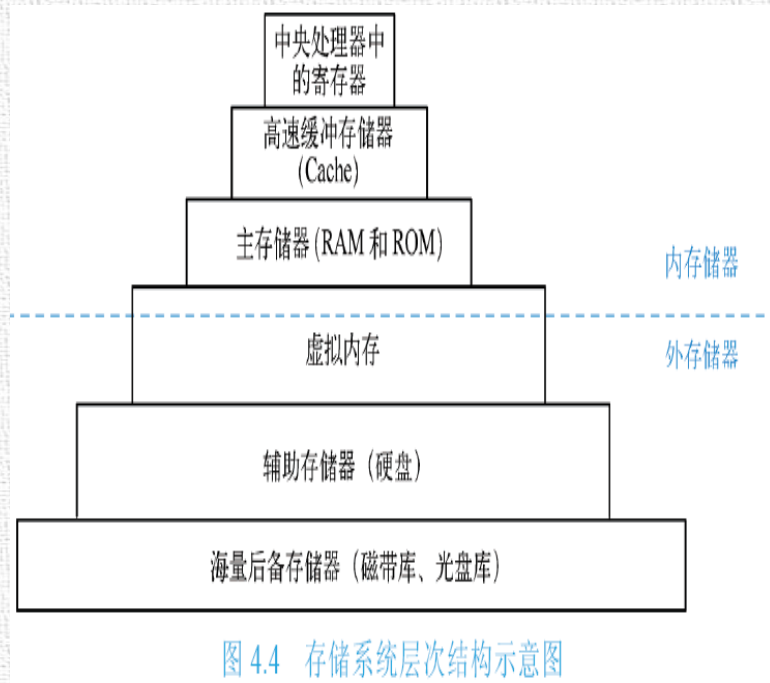


图 4.4 存储系统层次结构示意图

3.4 软件系统



- 计算机软件通常分为系统软件和应用软件两大类。
- 应用软件必须在系统软件的支持下才能运行。
- 没有系统软件，计算机无法运行；有系统软件而没有应用软件，计算机还是无法解决实际问题。



3.4.2 软件系统——操作系统



细分



- 操作系统是用户和计算机之间的界面。
- 机器层：硬件设备本身，它的对外界面由机器指令系统组成。
- 操作系统层：它的对内界面是管理和控制各种硬件资源，对外界面是为用户提供操作方便之外的一组系统软件、应用程序和计算机用户等）

3.4.2 操作系统——典型操作系统



- MS-DOS
- Windows
- Unix
- Linux
- Mac OS
- iOS
- Android
- Chrome OS



3.4.3 计算机语言



- 人和计算机交流信息使用的语言称为计算机语言或程序设计语言。
- 计算机语言通常分为机器语言、汇编语言和高级语言3类。



3.5.1 数据库——发展历程



- 人工管理阶段：数据管理任务都是针对每个具体应用，由编程人员单独设计解决，数据与程序是一个整体，数据无独立性，不能共享。
- 文件系统阶段：文件系统把计算机中的数据组织成相互独立的数据文件。应用程序和数据之间有了一定的独立性。但是数据仍然面向特定的应用程序，数据的共享性差，冗余度大，数据易产生不一致性，数据间联系弱，数据的管理和维护的代价也很大。
- 数据库系统阶段：数据库技术的特点是数据不再只针对某一特定应用，而是面向全组织，具有整体的结构性，共享性高，冗余度小，具有一定的程序与数据间的独立性，并且实现了对数据进行统一的控制。



3.5.2 数据库——定义



- 数据库管理系统（ DataBase Management System, DBMS ）：在操作系统支持下的操纵和管理数据库的系统软件，用于建立、使用和维护数据库。
- DBMS对数据库进行统一的管理和控制，以保证数据库的安全性和完整性。
- 用户DBMS通过访问数据库中的数据，数据库管理员也通过DBMS进行数据库的维护工作。



3.5.3 关系数据库——数据模型



- 在数据库系统中，人们通常按其数据结构类型来命名数据模型
 - 层次结构：对应的数据模型称为层次模型
 - 网状结构：对应的数据模型称为网状模型
 - 关系结构：对应的数据模型称为关系模型
- 层次模型和网状模型采用格式化的结构，在这类结构中实体用记录型表示，而记录型抽象为图的顶点。记录型之间的联系抽象为顶点间的连接弧。树形图的数据模型为层次模型；网状图的数据模型为网状模型。
- 关系模型为非格式化的结构，用单一的二维表的结构表示实体及实体之间的联系。满足一定条件的二维表，称为一个关系。



第四章 计算机网络与网络思维



- 主题1、 网络与网络思维
- 主题2、 计算机网络
- 主题3、 互连网
- 主题4、 网络安全
- 主题5、 物联网与云计算



4.1 网络与网络思维



网络思维要点

- ✓ 名字空间：利用一组规则对网络节点“取名”，在互联与通信时确认“对方”是谁。如E-Mail，域名，IP地址，微信名等。注意名字的唯一性，友好性，名字解析。
- ✓ 网络拓扑结构：用节点和连接线构成网络，可描述网络节点之间的连通性。
- ✓ 网络协议：网络消息的传递的过程依靠协议，可明确调用哪个层次的通信接口，如何解析消息数据。



4.2 计算机网络



- 定义

计算机网络是将地理位置不同并具有**独立功能**的多个计算机系统通过通信介质**互连**在一起，在网络软件管理下实现网络**资源共享和相互通信**的整个系统。

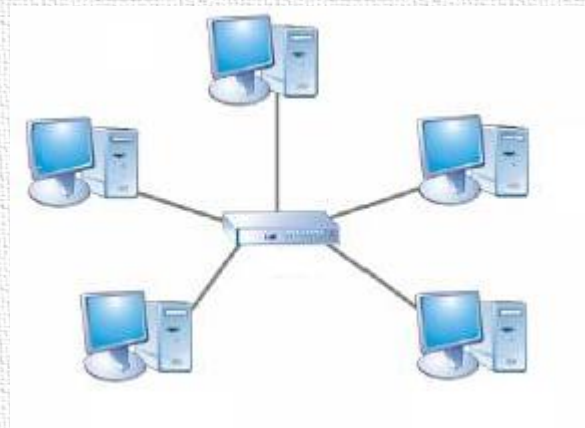


4.2.1 网络拓扑结构



- 拓扑结构

- 星型
- 总线型
- 环型
- 树型



4.2.3 网络设备



- **中继器**
主要作用是用来放大信号
- **集线器**
 - 工作在OSI模型的物理层
- **交换机**
 - 工作在数据链路层。网络邮递员。
- **路由器**
- **网关**



4.3.1 TCP/IP协议

• 什么是TCP/IP

- 传输控制协议/因特网互联协议
- 是一个协议族, 包括TCP、IP、UDP、ICMP、TELNET、FTP、SMTP、ARP等Internet基本协议

OSMISO 模型

TCP/IP 协议

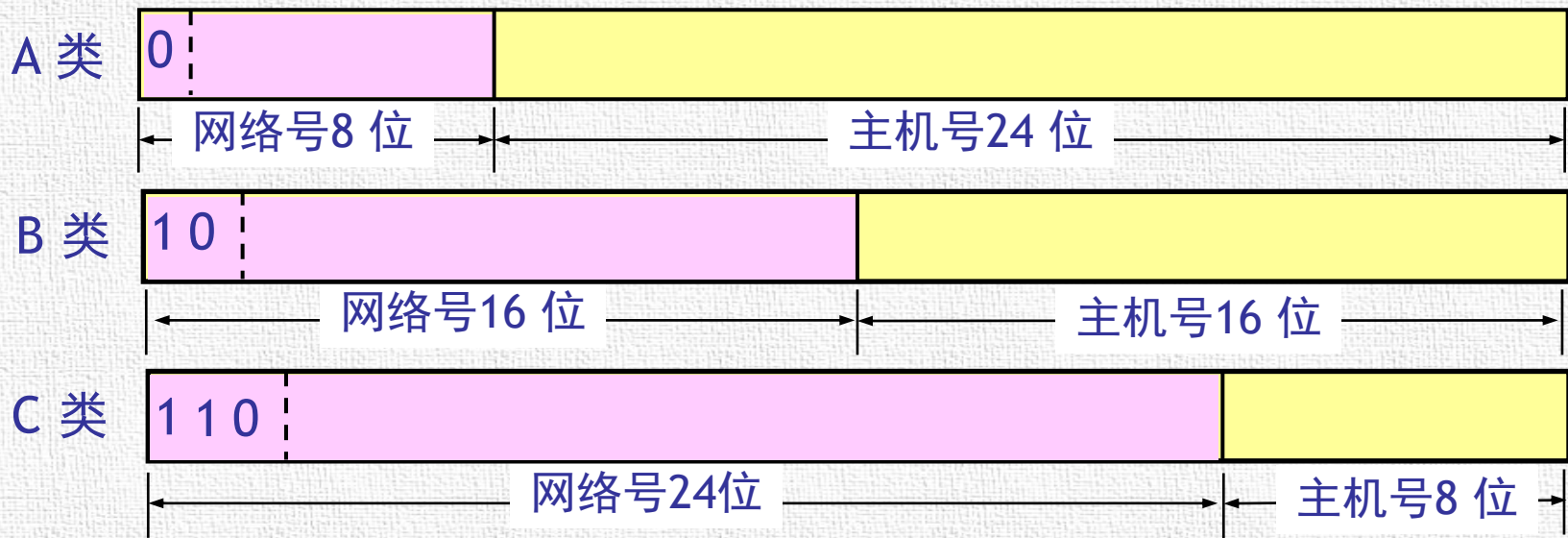
TCP/IP 模型

应用层	文件传输	远程登录	电子邮件	网络文件	网络管理	
表示层	协议	协议	协议	服务	协议	应用层
会话层	FTP	Telnet	SMTP	协议	SNMP	
				NFS		
传输层	TCP				UDP	传输层
网络层	IP		ICMP	ARP RARP		网际层
数据链路层	Ethernet	FDDI	Token-Ring/ IEEE 802.5	ARCnet	PPP/SLIP	网络接口层
	IEEE 802.3					
物理层						硬件层

4.3.1 TCP/IP协议



- IP地址分类

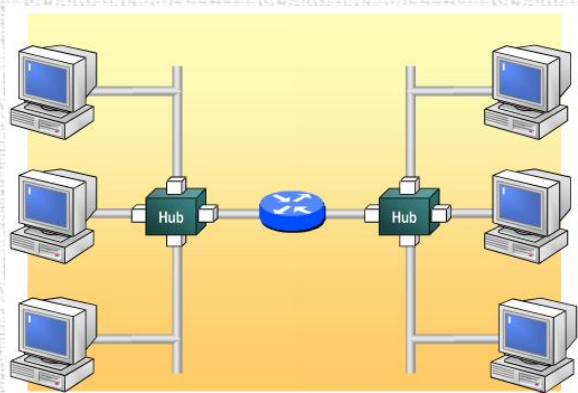


4.3.1 TCP/IP协议



- 子网掩码

- 一个规模较大的网络可被划分为若干个子网络
- 子网掩码可以让若干子网络共享同一网络地址



子网与子网掩码



于是我们可以知道：

- A类地址的缺省子网掩码为 255.0.0.0
- B类为255.255.0.0
- C类为255.255.255.0



子网与子网掩码



一个B类网络，但它分成了几十个子网
子网掩码为255.255.252.0

11111111. 11111111. 11111100. 00000000

主机地址为166.111.4.5的计算机与主机166.111.5.1是否处于同一子网？

10100110. 01101111. 00000100. 00000101
10100110. 01101111. 00000101. 00000001



4.4 网络安全

- **定义：** 网络安全是指网络系统的硬件、软件及其系统中的数据受到保护，不因偶然的或者恶意的原因而遭受到破坏、更改、泄露，系统连续可靠正常地运行，网络服务不中断



4.4 网络安全



- 网络安全的基本需求:

- **可用性:**网络信息可被授权实体访问并按需求使用的特性;
- **完整性:**网络信息未经授权不能进行改变的特性
- **保密性:**防止信息泄漏给非授权个人或实体;
- **不可抵赖性:**所有参与者都不可能否认或抵赖曾经完成的操作和承诺。



4.5 物联网

- 定义

通过RFID (射频识别)、红外感应器、全球定位系统、激光扫描器等信息传感设备，按约定的协议，把任何物品与互联网连接起来，进行信息交换和通讯，以实现智能化识别、定位、跟踪、监控和管理的一种网络



4.5 物联网

• 物联网的三层结构

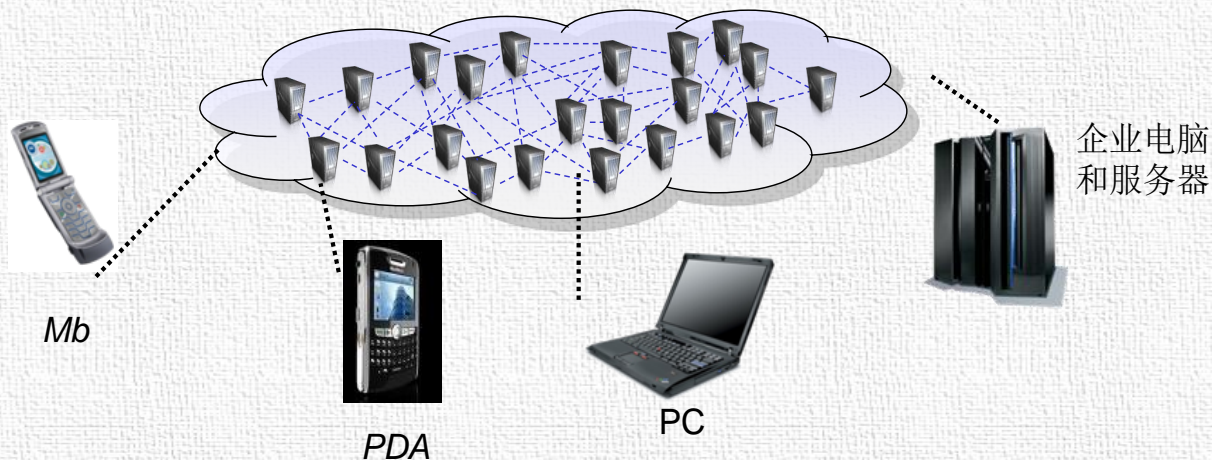
- 感知层利用各种传感设备感知各种物体的信息，实时搜集、获取、记录数据。
- 通过对物体的识别和通讯，将数据化的虚拟物体接入网络，实现传送与交互。
- 应用层利用各种智能计算技术，对海量的数据和信息进行分析 and 处理。



4.5 云计算



- **定义：**云计算是一种基于互联网的计算方式，共享的软硬件资源和信息可以按需求提供给计算机和其他设备。



4.5 云计算



- 数据在云端：不怕丢失，不必备份,可以任意点的恢复；
- 软件在云端：不必下载自动升级；
- 无所不在的计算：在任何时间，任意地点，任何设备登录后就可以进行计算服务；
- 无限强大的计算：具有无限空间的，无限速度



第五章 问题求解与计算思维实训

916

- 主题1、 算法与算法思维
- 主题2、 算法策略
- 主题3、 问题求解框架
- 主题4、 问题求解经典案例分析

5.1 算法与算法思维



- 算法的定义

- 算法 (Algorithm) 是指解题方案的准确而完整的描述，是一系列解决问题的方法步骤或清晰指令的陈述。
- 算法代表着用系统的方法描述解决问题的策略机制。也就是说，能够对一定规范的输入，在有限时间内获得所要求的输出。
- 算法中的指令描述的是一个计算，当其运行时能从一个初始状态和（可能为空的）初始输入开始，经过一系列有限而清晰定义的状态，最终产生输出并停止于一个终态。实际上，算法能够完成的就是图灵对可计算性定义的范畴。



5.1.1 发现算法



- 用计算机实现一个问题的求解，通常包括两个步骤：
 - 一是发现潜在的算法；
 - 二是以程序的方法表示并实现算法。
- 算法发现的过程和一般问题的求解过程之间存在着紧密的联系，因此在计算机科学领域，人们把问题求解，简化为一种算法，但并不是所有的问题一定都能找到解决问题的算法。



5.1.2 描述算法



- 常用的描述算法的方法有四种：
 - 自然语言
 - 流程图
 - 伪代码
 - 计算机语言。
- 需要特别说明的是，根据分析问题所处的阶段，描述算法的步骤可粗可细，可以粗到只有一个大的解题框架，也可以细到接近程序设计语言的每条语句。



5.1.2 描述算法



用流程图描述算法

基本算法结构

- “结构化程序设计方法”的核心是规定了算法的三种基本结构：顺序结构、分支结构和循环结构



5.1.3 评价算法



- 一个算法的优劣可以用空间复杂度与时间复杂度来衡量。
- 解决方案的优劣，即评价算法的标准，基本包括：
 - 算法的时间复杂度
 - 算法的空间复杂度
 - 算法的正确性、可读性和健壮性



5.1.3 评价算法——时间复杂度



- 时间复杂度 (Time Complexity)
 - 算法的时间复杂度是指执行算法所需要的计算工作量。在计算机科学中，算法的时间复杂度是一个函数，它定量描述了该算法的运行时间。
 - 一般情况下，算法的基本操作重复执行的次数是模块N的某一个函数 $F(N)$ ，因此，算法的时间复杂度记做： $T(N)=O(F(N))$



5.1.3 评价算法——时间复杂度



求解算法的时间复杂度的具体步骤是：

(1) 找出算法中的基本语句

算法中执行次数最多的那条语句就是基本语句，通常是最内层循环的循环体

(2) 计算基本语句的执行次数的数量级

只需计算基本语句执行次数的数量级，这就意味着只要保证基本语句执行次数的函数中的最高次幂正确即可，可以忽略所有低次幂和最高次幂的系数

(3) 用大O记号表示算法的时间性能

将基本语句执行次数的数量级放入大O记号中

常见的算法时间复杂度由小到大依次为：

$$O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < \dots < O(2^n) < O(n!)$$



5.1.3 评价算法——时间复杂度



```
for (i=1; i<=n; i++)
```

```
    x++;
```

```
for (k=1; k<=m; k++)
```

```
    for (j=1; j<=m; j++)
```

第一个for循环的时间复杂度为 $O(n)$

第二个for循环的时间复杂度为 $O(n^2)$

整个算法的时间复杂度为 $O(n+n^2)=O(n^2)$

```
int sum = 0, n = 100;
printf("I love you.com\n");
printf("I love you.com\n");
printf("I love you.com\n");
printf("I love you.com\n");
printf("I love you.com\n");
printf("I love you.com\n");
sum = (1+n)*n/2;
```

```
int i, j, n = 100;
for( i=0; i < n; i++ )
{
    for( j=0; j < n; j++ )
    {
        printf("I love FishC.com\n");
    }
}
```

$O(n^2)$

5.1.3 评价算法——空间复杂度



- 空间复杂度 (Space Complexity)
 - 算法的空间复杂度是指算法需要消耗的内存空间。其计算和表示方法与时间复杂度类似，一般都用复杂度的渐近性来表示。同时间复杂度相比，空间复杂度的分析要简单得多。
 - 类似于时间复杂度的讨论，一个算法的空间复杂度 $S(N)$ 定义为该算法所耗费的存储空间，它也是问题规模 N 的函数。
 - 一个算法所需的存储空间用 $F(N)$ 表示，记为： $S(N)=O(F(N))$



5.2 算法策略



- 问题空间会随着问题解决的进程而逐渐得到丰富和扩展。而且，在解决某一特定问题时，不同个体的问题空间可能是有差别的。一个问题解决者对问题的解决过程，就是穿越其问题空间搜索一条通往问题目标状态的路径。事实上，对大多数问题来说，可以通过多条路径来达到问题的解决。
- 经典的算法策略主要包括：枚举算法、递推算法、递归算法、迭代算法、分治算法、贪心算法和回溯算法等。



5.3.2 问题求解框架



- 通过计算机解决一个具体问题时，大致需要经过下列几个步骤：
 - 首先要从具体问题中抽象出一个适当的数学模型，
 - 然后选择并确定合适的数据结构并设计一个解此数学模型的算法，
 - 最后编出程序、进行测试、调整直至问题得到最终解答。



5.3.2 问题求解框架



- 例18题：数字叠加问题
 - 计算 $S=a+aa+aaa+aa\dots a$





Thanks !

