

第六章

计算机软件基础

主讲人：李延红 讲师

目录

- 一 计算机软件概述
- 二 操作系统
- 三 数据结构
- 四 数据库系统
- 五 软件工程
- 六 人工智能

6.1 计算机软件概述

一、计算机软件的概念

软件：指为运行、维护、管理及应用计算机所编制的所有程序及其文档资料总和。

程序：计算任务的处理对象和处理规则的描述；

文档：为了便于了解程序所需的阐明性资料。

特性：

- ①软件是功能、性能相对完备的程序系统。程序是软件，软件不仅仅是程序；
- ②软件是具有使用性能的软设备；
- ③软件只有过时而无磨损的信息商品。

IEEE对软件的定义(1983):

软件是计算机程序、方法、规则、相关的文档资料以及在计算机上运行的程序时所必需的数据的集合。

计算机程序：计算机需要遵照执行的一系列指令，它作为一种具有逻辑结构的信息，精确而完整地描述计算任务中的处理对象和处理规则。

方法、规则：通常是在文档中说明并在程序中实现的。

文档：开发、使用和维护程序所需要的图文资料。

数据：向一个程序提供的输入。

计算机软件的发展历史

第1阶段 1946-1956	从第一台计算机上程序的出现到实用的高级程序设计语言出现以前。设计和编制程序采用个体工作方式，强调编程技巧，尚未出现软件一词。
第2阶段 1956-1968	从实用的高级程序设计语言出现到软件工程出现以前。出现软件一词，出现软件危机，设计和编制程序的工作方式逐渐转向合作方式。
第3阶段 1968年以后	软件工程出现以后迄今。对于大型软件的开发，只有采用工程方法才能适应。1968年提出软件工程。

二、计算机软件的分类

系统软件：居于计算机系统中最靠近硬件的一层，其他软件一般通过系统软件发挥作用，它与具体的应用领域无关，如操作系统、编译程序等。

二、计算机软件的分类

支撑软件：支撑软件的开发、维护与运行的软件工具，如数据库管理系统、网络软件、软件开发环境和中间件等。

应用软件：特定应用领域专用的软件。

三、计算机软件与硬件的关系

硬件：构成计算机系统的物理装置。

软件：用计算机指令和算法语言编写的程序以及运行程序所需的文档、数据。

关系：硬件是软件运行的基础

软件是对硬件功能的扩充和完善

软件的运行最终被转换为对硬件的操作

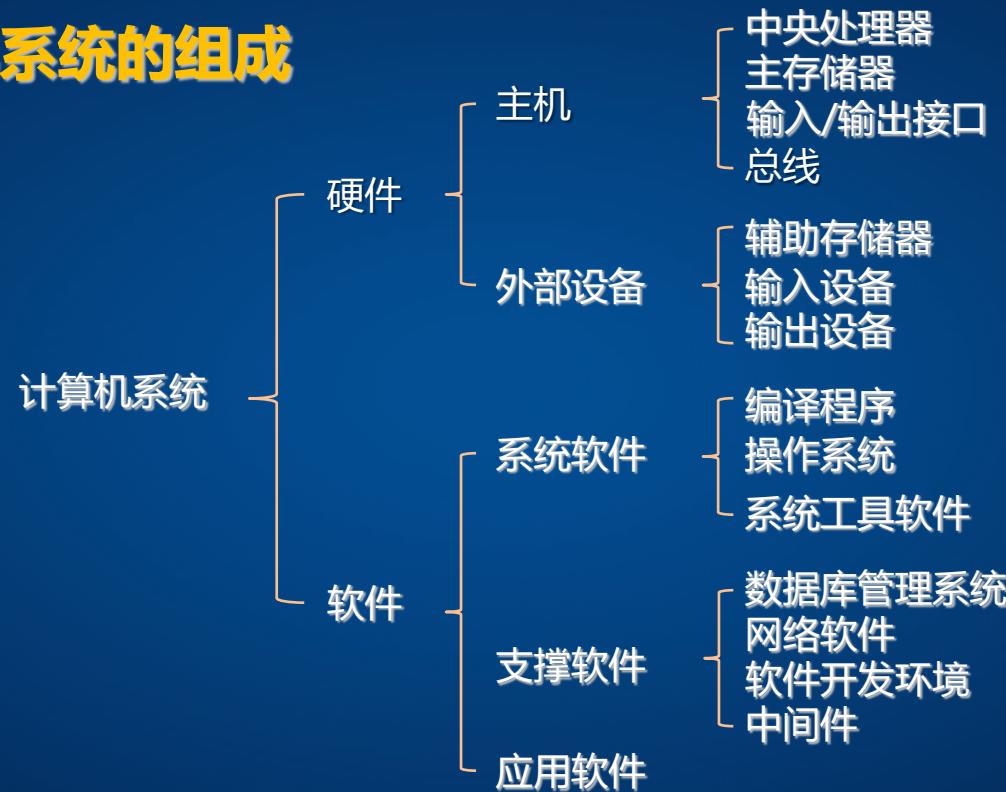
三、计算机软件与硬件的关系

软件的运行最终被转换为对硬件的操作

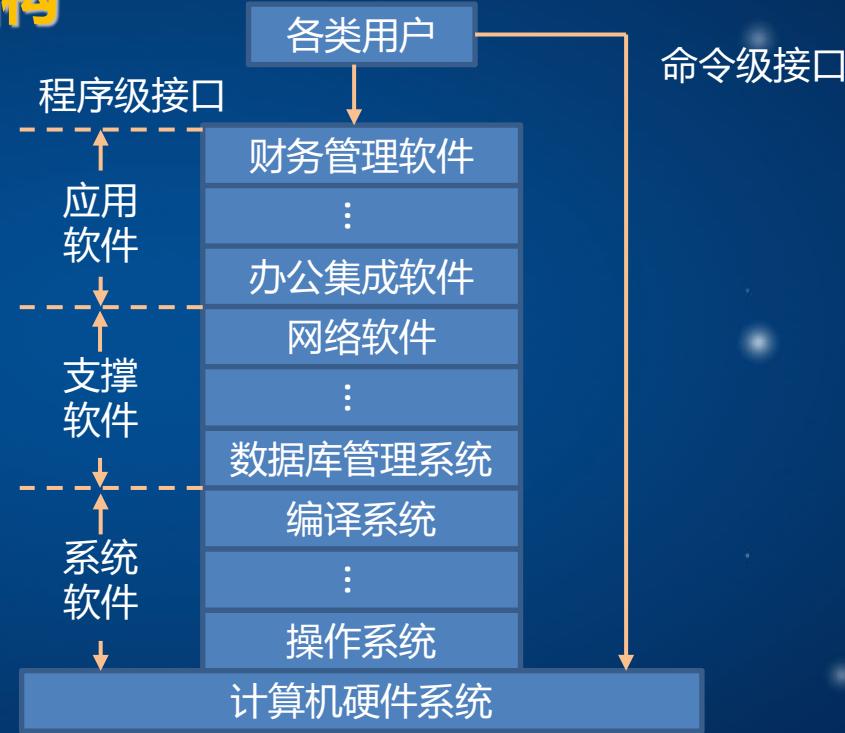
计算机：硬件、软件→人工生命

人：躯体、灵魂→自然生命

计算机系统的组成



计算机系统的层次结构



6.2 操作系统

操作系统是所有从事计算机应用、开发和研究的人经常使用的系统软件。

一、操作系统的定义

管理硬件资源、控制程序运行、改善人机界面、为应用软件提供支持的系统软件。

注意：

- (1) 操作系统是软件，而且是系统软件。
- (2) 操作系统的基本职能是控制管理系统内各种资源，有效地组织多道程序的运行；
- (3) 操作系统用户与计算机之间的接口。

二、操作系统的功能

1、主要作用

- ① 各种软硬件资源管理；
- ② 提供良好的用户界面。

二、操作系统的功能

2、基本功能

五大功能：处理机管理、存储管理

设备管理、文件管理

用户接口

6.2 操作系统

处理机管理：通过进程管理协调多道程序间的关系，解决对处理机实施分配调度策略、进行分配和回收等。

6.2 操作系统

存储管理：管理内存资源。

主要包括内存分配、地址映射、内存保护和内存扩充。

6.2 操作系统

虚拟内存→系统属性/高级→
性能设置/高级→虚拟内存/
更改



6.2 操作系统

设备管理：对硬件设备进行管理。

主要包括缓冲区管理、设备分配、设备驱动和设备无关性。

6.2 操作系统

文件管理：对信息资源的管理，操作系统将这些资源以文件的形式存储在外存上。

文件系统的重要概念：

- 文件
- 磁盘分区
- 目录结构

6.2 操作系统

文件

按一定格式建立在存储设备上的一批相关信息的有序集合。

文件名：每个文件必须起一个名字。

主文件名.扩展名/文件类型

文件扩展名及其意义

文件类型	扩展名	说明
批处理文件	bat	将一批系统操作命令存储在一起，可供用户连续执行
可执行文件	exe, com	可执行程序文件
源程序文件	c, cpp, bas, asm	程序设计语言的源程序文件
Office文件	doc, xls, ppt	Office中Word、Excel、PowerPoint创建的文档
网页文件	htm, asp	前者是静态的，后者是动态的
音频文件	wav, mp3, mid	声音文件，不同的扩展名表示不同格式的音频文件
图像文件	bmp, jpg, gif	图像文件，不同的扩展名表示不同格式的图像文件
压缩文件	zip, rar	压缩文件

6.2 操作系统

磁盘分区

把一个磁盘驱动器划分成几个逻辑上独立的驱动器，这些分区被称为卷。

目的：

- 磁盘容量很大，为便于管理；
- 安装不同的系统，如Windows和Linux等。

6.2 操作系统

Windows操作系统的分区：

- 一个硬盘可以分为磁盘主分区和磁盘扩展分区
- 扩展分区还可以细分为几个逻辑分区。
- 每一个主分区或逻辑分区就是一个逻辑驱动器，它们各有盘符。

计算机管理

文件 (F) 操作 (A) 查看 (V) 窗口 (W) 帮助 (H)



计算机管理 (本地)

- 系统工具
 - 事件查看器
 - 共享文件夹
 - 本地用户和组
 - 性能日志和警报
- 设备管理器
- 存储
 - 可移动存储
 - 磁盘碎片整理程序
 - 磁盘管理
- 服务和应用程序

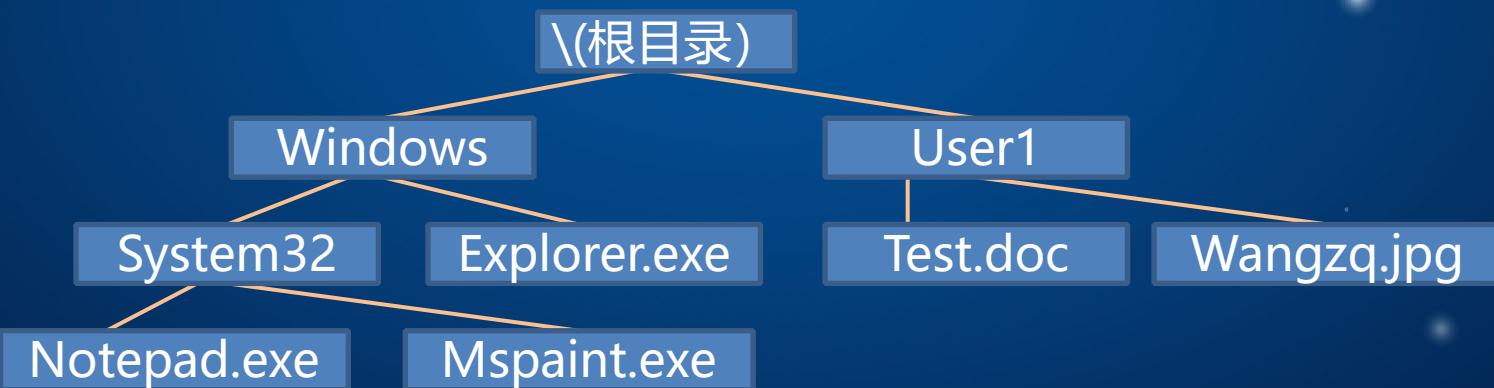
卷	布局	类型	文件系统	状态	容量	空闲空间	% 空闲
WLD20070602 (E:)	磁盘分区	基本	NTFS	状态良好	33.66 GB	24.87 GB	73 %
WWW20070602 (D:)	磁盘分区	基本	NTFS	状态良好	29.29 GB	21.25 GB	72 %
WZQ20070602 (C:)	磁盘分区	基本	NTFS	状态良好 (系统)	48.83 GB	38.68 GB	79 %

卷	布局	类型	文件系统	状态
磁盘 0	基本 111.78 GB 联机	WZQ20070602 (C:)	48.83 GB NTFS 状态良好 (系统)	WWW20070602 (D:)
		WWW20070602 (D:)	29.29 GB NTFS 状态良好	WLD20070602 (E:)
CD-ROM 0	DVD (F:)		无媒体	

■ 主要磁盘分区 ■ 扩展磁盘分区 ■ 逻辑驱动器

多级目录结构：

为了便于管理计算机中成千上万的文件，大多数的文件系统允许用户在根目录下建立子目录，在子目录下再建立子目录，也就是将目录结构构建构成树状结构。



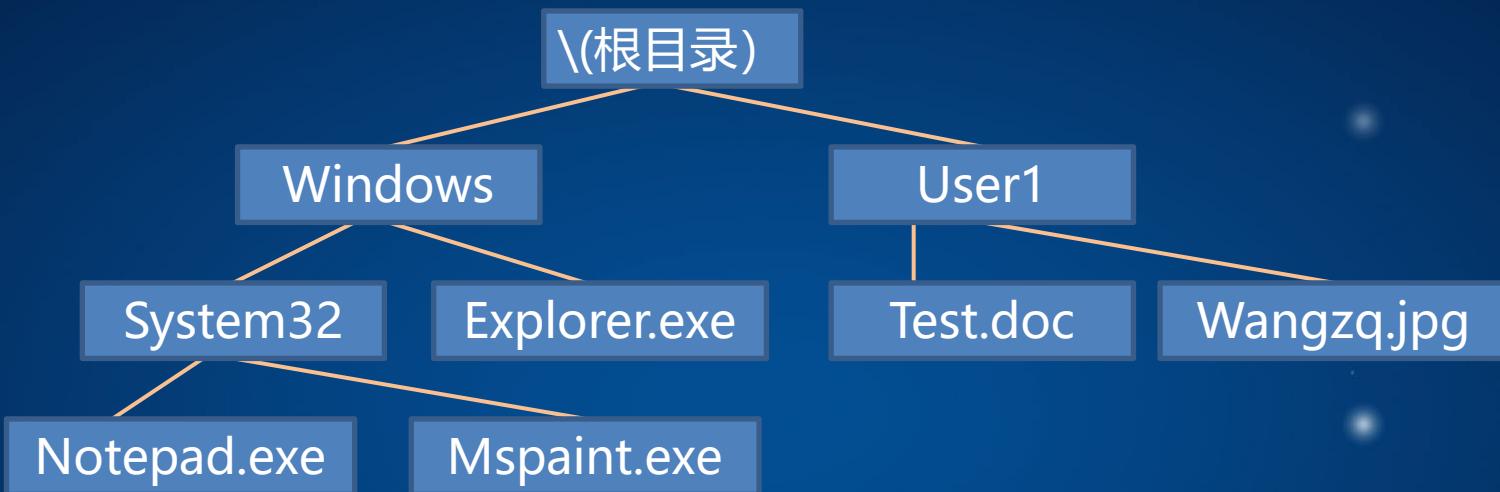
名称	大小	类型
sample-Body-search.vbs	2 KB	超级解霸.
sample-Browser-Type.vbs	9 KB	超级解霸.
sample-computerGroup...	1 KB	XML 文档
sample-Duwamish.vbs	23 KB	超级解霸.
sample-Fitch-Mather...	16 KB	超级解霸.
sample-Form-data.vbs	3 KB	超级解霸.
sample-perfCounters...	1 KB	XML 文档
sample-perfCounters...	1 KB	XML 文档
sample-perfCounters...	1 KB	XML 文档
sample-perfCounters...	1 KB	XML 文档

盘符:\子目录1\子目录2\...\子目录n\文件名

C:\Program Files\Microsoft Office\OFFICE11\winword.exe

目录路径：

- 绝对路径，从根目录开始，依序到该文件之前的名称。
- 相对路径，从当前目录开始到某个文件之前的名称。



Notepad.exe文件的**绝对路径**为

C:\Windows\System32\Notepad.exe。

如果当前目录为System32，则Test.doc文件的**相对路径**为

..\..\User1\Test.doc。

用户接口：

操作系统与用户之间的接口有：

① 程序接口

通过API调用例行程序实现既定的操作

② 命令接口

通过交互命令方式,对计算机进行操作

三、协调机器的活动

1、进程

程序的一次执行过程，是程序实体的运行过程，是系统进行资源分配和调度的一个独立单位，其属性会随着时间的推进而改变。

三、协调机器的活动

2、程序与进程

- 一个程序可以包含多个进程；
- 进程可以描述并发活动，程序则不明显；
- 进程执行需要处理机，程序存储需要介质；
- 进程有生命周期，程序是永存的。

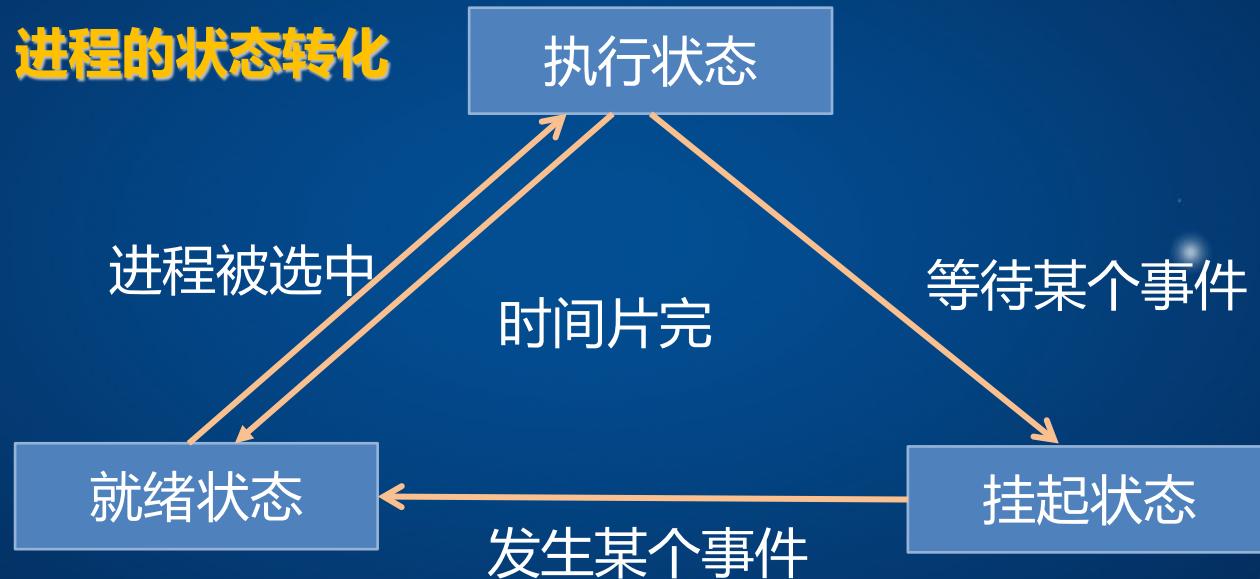
三、协调机器的活动

3、进程的状态

- **挂起状态**是指进程正在等待系统为其分配所需资源而暂未运行；
- **就绪状态**是指进程已获得所需资源并被调入内存，它具备了执行的条件但仍在等待获得处理机资源，以便投入运行；
- **运行状态**是指进程占有处理机且正在运行的状态。

三、协调机器的活动

4、进程的状态转化



四、竞争控制

操作系统的一个重要任务是将机器的各种资源合理分配给系统中的各个进程。

- 临界资源
- 信号量
- 死锁

四、竞争控制

临界资源

- 临界资源 (Critical Resource) 指计算机系统中在同一个时刻只能由一个进程使用的资源。
- 如硬件资源中的打印机、磁带机等，软件资源中的变量、表格、队列等。
- 对临界资源的使用采用互斥方式即一个进程使用完之后，另一个进程才能使用。

四、竞争控制

信号量

- 解决临界资源访问的另一种方法是使用测试并置位指令，它在许多机器语言里可用。这条指令要求CPU检索一个标志的值，记住它，然后置位该标志，所有工作都在一条机器指令内完成。
- 刚才描述的一个正确实现的标志称为信号量 (Semaphores)

四、竞争控制

死锁

指多个进程因竞争资源而造成的一种僵局，若无外力干预，这些进程都将永远不能继续运行。

例如，一个进程可能已有对打印机的访问权，同时它还在等待访问CD播放机，而另一个进程已经获得CD播放机的访问权，却在等待访问打印机，那么这两个进程就会陷入死锁，无限制等待对方手上的资源。

四、竞争控制

死锁



四、竞争控制

死锁的必要条件

- 互斥条件：存在对不可共享资源的竞争。
- 请求和保持条件：一个进程接受了某些资源后，稍后还将请求其他的资源。
- 不剥夺条件：进程已获得的资源，在使用完之前，不被外力剥夺。
- 环路等待条件：进程推进顺序不当，出现互相等待其他进程已获得资源。

四、竞争控制

解决锁的方法：

避免任何一个必要条件成立。

五、操作系统中的计算思维

- 时间管理
- 空间管理
- 设计哲学

时间管理举例1

如何合理安排期末复习？

解决方法：

首先对这个问题进行抽象：

- 1) 每个课程的复习可以定义为一个实时进程 P_i ；
- 2) 课程所需复习时间为 T_i ；
- 3) 课程复习必需在考试时间之前完成，考试时间既为进程截止完成时间 E_i ；
- 4) 将复习时间可以切分为单位时间片，在每一个时间片里学生只复习一个科目。

如何合理安排期末复习？

解决方法：

其次采用CPU调度中的实时调度策略采用最低松弛度（任务的紧急程度）优先算法来解决。

- 1) 在 t 时刻开始的时间片，计算出各个科目的松弛度 $LF_i = E_i - T_i - t$,
- 2) 按照最低松弛度优先的原则自动选择复习科目即可得到了合理的复习时间表。

时间管理举例1

如何用最短的时间把两个菜准备好？

解决方法：

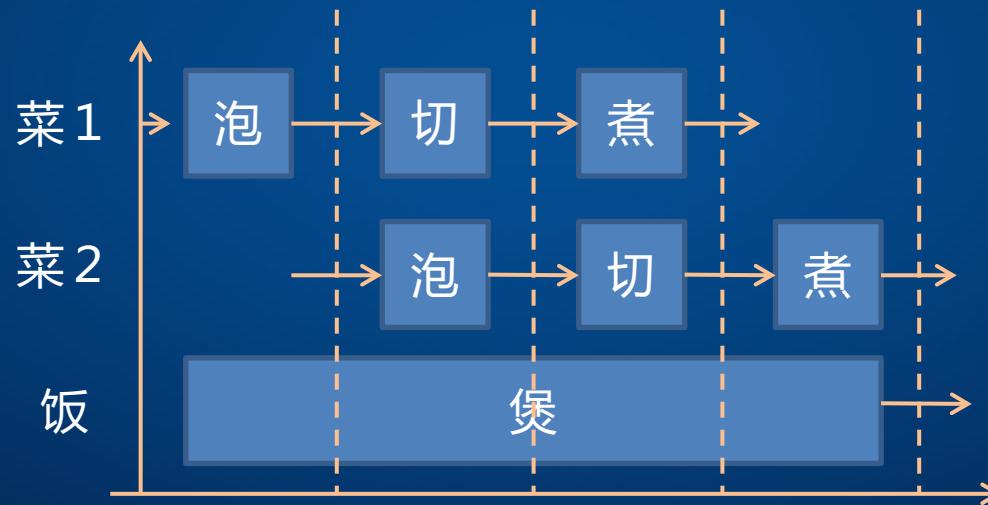
首先对这个问题进行抽象：

- 1) 假设两个菜分别需要三个步骤：泡洗、刀切、烹煮，
- 2) 白饭单独使用电饭煲煲制。

如何用最短的时间把两个菜准备好?

解决方法:

其次按照计算机执行指令的流水线作业方法完成



空间管理举例2

如何存储复杂而又大量的信息？

解决方法：



- 1) 人们最重要的信息比如家庭地址、银行账号密码、报警电话等好比操作系统内核必需常驻“内存”，
- 2) 其它并不重要或者暂时不用的信息可以采用“虚存”比如笔记本、手机、电脑、甚至网络等外部工具进行存储。



解决方法：

3) 在“内存”即大脑中保存一个“页表”记录这些信息的存储位置或者检索方法，当发生“缺页”即信息缺失时就可以随时根据“页表”把缺失信息从虚存调入实存。

设计哲学举例

如何确保住房作为人类生存的必需品，其分配过程能够兼顾效率和公平？

解决方法：

1) 房地产市场中个人按照经济能力和自身条件选购相应住房，实现资源合理分配；

多级反馈调度算法中，进程按照优先级进行排队分时使用CPU

设计哲学举例

如何确保住房作为人类生存的必需品，其分配过程能够兼顾效率和公平？

解决方法：

2) 银行对首次购房者提供相对优惠的贷款政策，帮助无房者满足基本住房要求；

新提交进程排在最高优先级的队列中，保证所有进程能够得到快速响应。

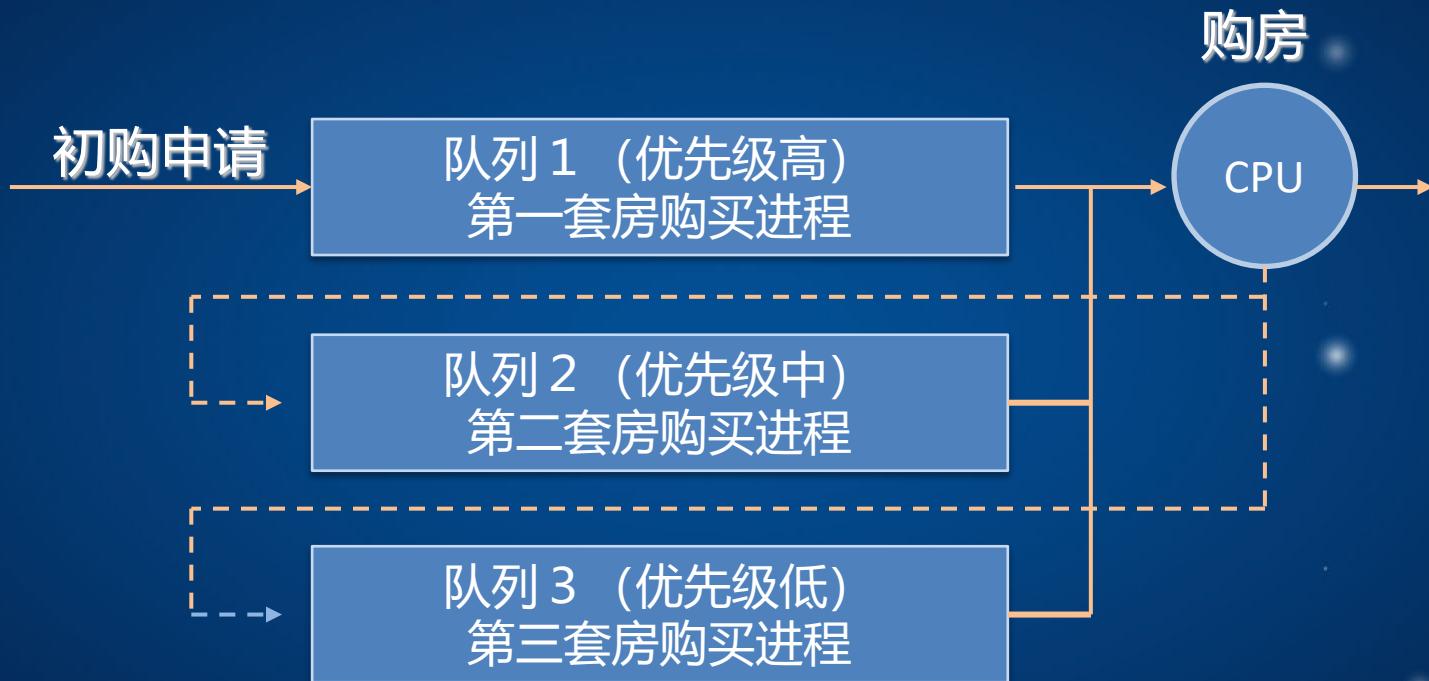
设计哲学举例

如何确保住房作为人类生存的必需品，其分配过程能够兼顾效率和公平？

解决方法：

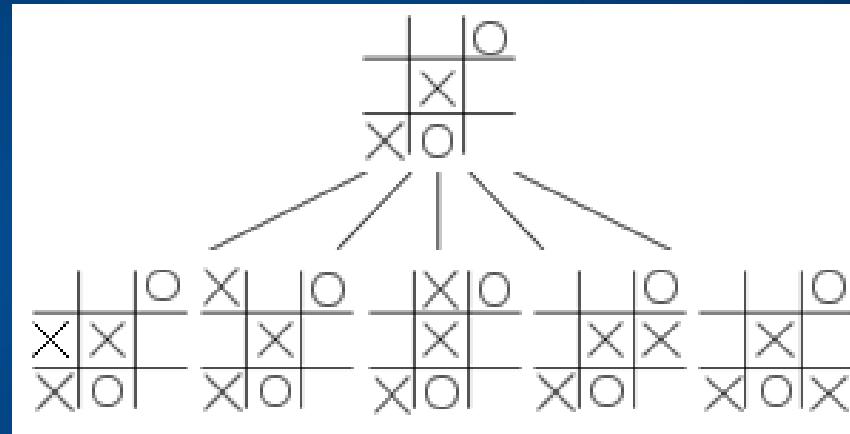
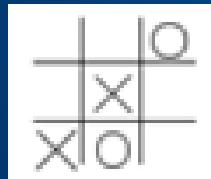
3) 政府调控中对购置多套房产的限购措施，可以让更多人有机会购房，防止恶意炒作，提高住房资源分配的公平性。

当进程用完CPU一个单位计算资源后，会被降级到更低优先级队列，避免少数组程序长期霸占CPU，让更多的进程有机会参与使用CPU。



6.3 数据结构

问题的引入：



数据结构是一门研究非数值计算的程序设计问题中计算机的操作对象以及他们之间的关系和操作等的科学。

一、数据结构的概念

1、数据 (Data)

是信息的载体，是描述客观事物的数、字符、以及所有能输入到计算机中，被计算机程序识别和处理的**符号的集合**。

- 数值性数据，如整型、实型等
- 非数值性数据，如字符及声音、图像、视频等

一、数据结构的概念

2、数据元素 (Data Element)

数据的基本单位。在计算机程序中常作为一个整体进行考虑和处理。

有时一个数据元素可以由若干数据项(Data Item)组成 (此时数据元素被称为记录)

数据元素又称为元素、结点、记录

数据元素举例

例如：描述一个运动员的数据元素可以是

姓名	俱乐部名称	出生日期	参加日期	职务	业绩
----	-------	------	------	----	----

年 月 日

称之为组合项

一、数据结构的概念

3、数据对象 (Data Object)

具有相同性质的数据元素的集合，是数据的一个子集。

- 整数数据对象

$$N = \{0, \pm 1, \pm 2, \dots\}$$

- 字母字符数据对象

$$C = \{ 'A', 'B', 'C', \dots, 'F' \}$$

一、数据结构的概念

4、结构 (Structure)

结构是元素之间的关系

数据结构就是相互之间存在一种或多种特定关系的数据元素的集合。

- 逻辑结构
- 存储结构

一、数据结构的概念

逻辑结构

数据对象中数据元素之间的相互关系

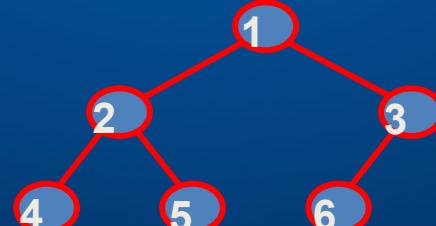
○集合结构



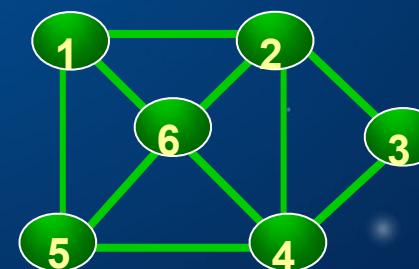
○线性结构



○树形结构



○图形结构

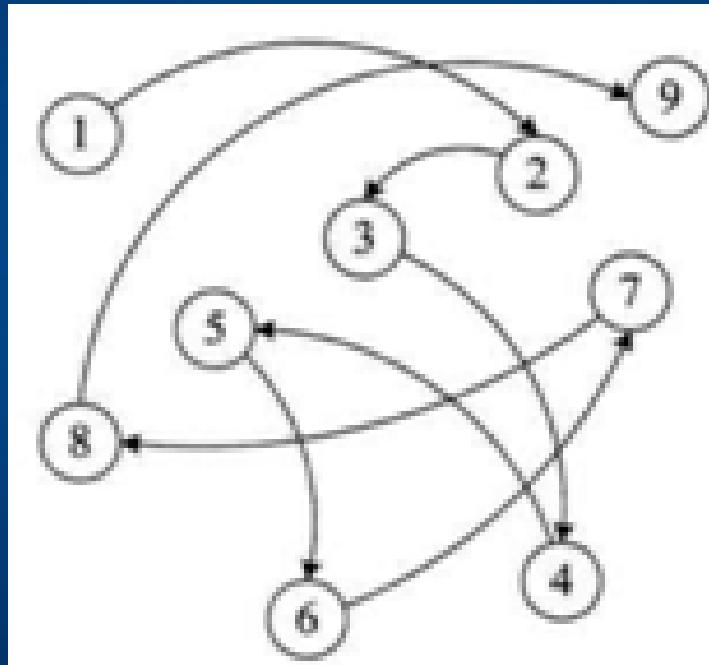


一、数据结构的概念

物理结构

数据结构在物理中的表示称为**物理结构** (Physical Structure) , 又称为**存储结构** (Storage Structure) 。

- **顺序存储**: 数据元素存放在地址连续的存储单元里
- **链式存储**: 把数据元素存放在任意的存储单元里, 这组存储单元可以是连续的, 也可以是不连续的



二、常用的数据结构

常用的数据结构

- 线性表
- 栈和队列
- 数组
- 树
- 图

二、常用的数据结构

线性表



在数据元素的非空有限集中

- 1、存在惟一的一个被称作“第一个”的数据元素(如“1”)
- 2、存在惟一的一个被称作“最后一个”的数据元素(如“6”)
- 3、除第一个元素外，每个数据元素均只有一个前驱
- 4、除最后一个元素外，每个数据元素均只有一个后继(next)，
如“1”的next是“2”，“2”的next是“3”)

二、常用的数据结构

线性表举例

1、26个字母的字母表: (A,B,C,D,...,Z)

2、一个学校的学生健康情况登记表:

姓名	学号	性别	年龄	班级	健康状况
王小林	790631	男	18	计91	健康
陈红	790632	女	20	计91	一般
刘建平	790633	男	21	计91	健康
张立立	790634	男	17	计91	神经衰弱
⋮	⋮	⋮	⋮	⋮	⋮

二、常用的数据结构

线性表的主要操作

- 插入一个元素：可以在任意位置插入元素
- 删除一个元素：可以在任意位置删除元素

二、常用的数据结构

栈和队列

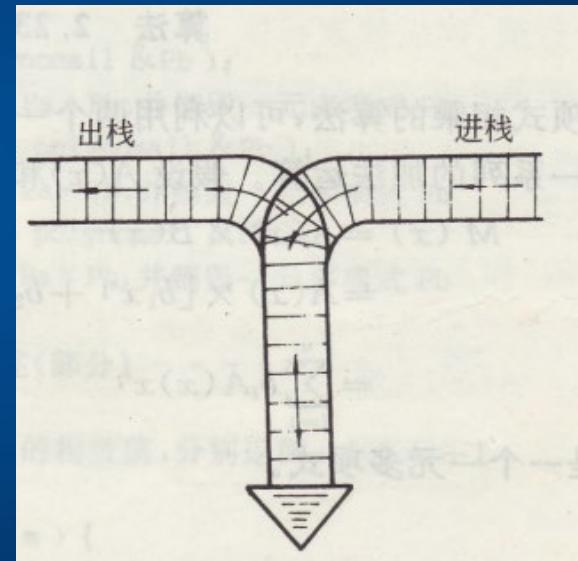
栈和队列的逻辑结构均为线性结构，是两种特殊的线性表

限定仅在表的两端进行插入或删除操作的线性表。

二、常用的数据结构

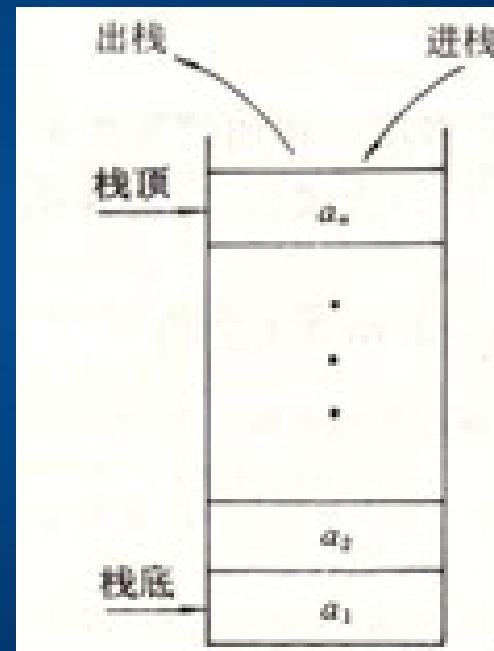
栈

- 限定仅在表尾进行插入或删除操作的线性表。
- 通常称表尾为栈顶，称表头为栈底。
- 不含元素的空表称为空栈。
- 栈也被称为后进先出表或LIFO (Last-In, First-Out) 表



二、常用的数据结构

例如，设栈 $S = a_1, a_2, \dots, a_n$ ($n \geq 0$)，则称 a_1 为栈底元素， a_n 为栈顶元素。



二、常用的数据结构

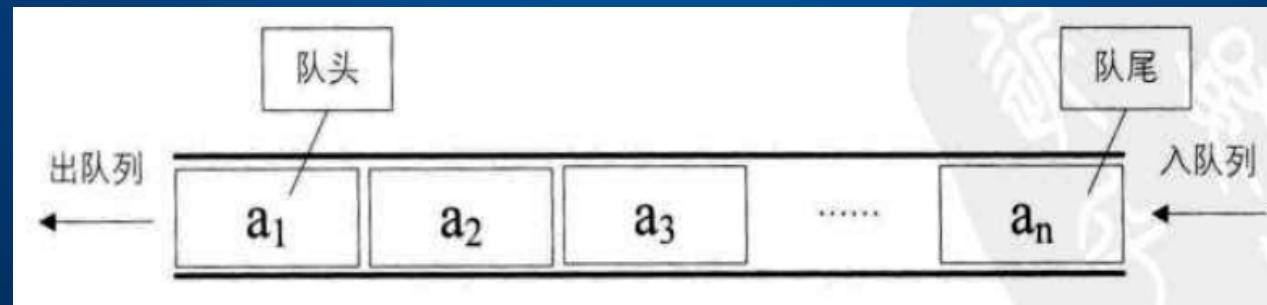
队列

- 插入限定在表的某一端进行，删除限定在表的另一端进行。
- 允许插入的一端为队尾，允许删除的一端为队头。
- 不含元素的空表称为空队列。
- 通常称队列为先进先出 (First In First Out, FIFO) 的线性表。

二、常用的数据结构

例如，设 $Q = a_1, a_2, \dots, a_n$ ($n \geq 0$)，那么 a_1 为队头元素， a_n 为队尾元素。

在程序设计中，一般采用循环队列，以便重复使用存储空间



二、常用的数据结构

数组

从逻辑结构上，数组可以看成是一般线性表的扩充。一维数组即为线性表。二维数组也可以看作为线性表，只不过线性表的数据元素亦是线性表。

使用数组便于处理成批的数据。例如矩阵运算、对信息进行排序等处理过程。

二、常用的数据结构

数组

例1, $A=[1, 2, 3, 4, 5]$ 是一个有5个元素的一维数组.

例2, 若数组B表示如下:

$$B= \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

则B是一个三行三列具有9个元素的二维数组, 也可以把B看成是一个线性表; $B=[b_1, b_2, b_3]$, 其中 b_i 本身也是一个线性表, 属于一维列向量,
 $b_1=[1, 2, 3]$, $b_2=[4, 5, 6]$, $b_3=[7, 8, 9]$.

二、常用的数据结构

树

- 树是有 $n (n \geq 0)$ 个结点的有限集合。
- 如果 $n=0$, 称为空树;
- 如果 $n>0$, 称为非空树, 对于非空树, 有且仅有一个特定的称为根 (Root) 的节点 (无直接前驱)

二、常用的数据结构

树

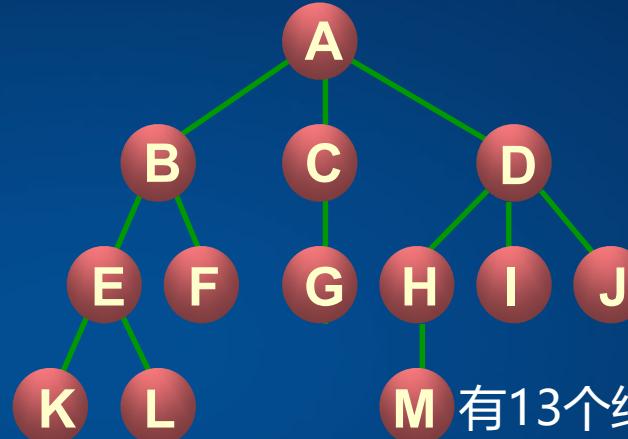
- 如果 $n > 1$, 则除根以外的其它结点划分为 m ($m > 0$) 个互不相交的有限集 T_1, T_2, \dots, T_m , 其中每个集合本身又是一棵树, 并且称为根的子树 (SubTree)。(此为递归定义)
- 每个结点都有唯一的直接前驱, 但可能有多个后继

二、常用的数据结构

树的定义(举例)

A

只有根结点的树



有13个结点的树

其中：A是根；其余结点分成三个互不相交的子集，

$T_1 = \{B, E, F, K, L\}$ ； $T_2 = \{C, G\}$ ； $T_3 = \{D, H, I, J, M\}$ ，

T_1, T_2, T_3 都是根A的子树，且本身也是一棵树

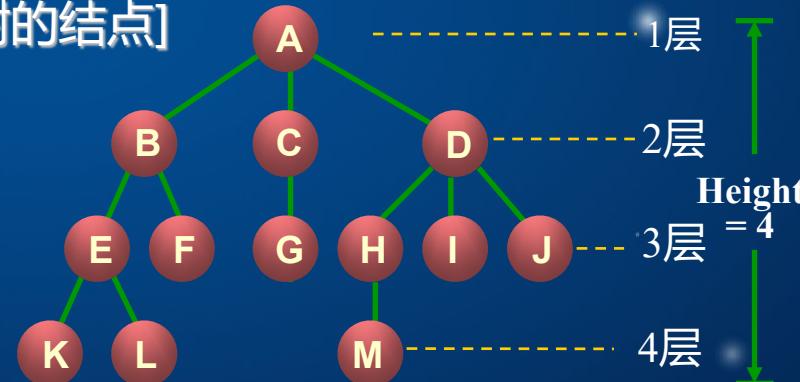
二、常用的数据结构

树的基本术语

- **结点**: 包含一个数据元素及若干指向其子树的分支
- **结点的度**: 结点拥有的子树数
- **叶结点**: 度为0的结点[没有子树的结点]
- **分支结点**: 度不为0的结点

[包括根结点], 也
称为非终端结点。

除根外称为内部结点



二、常用的数据结构

树的基本术语

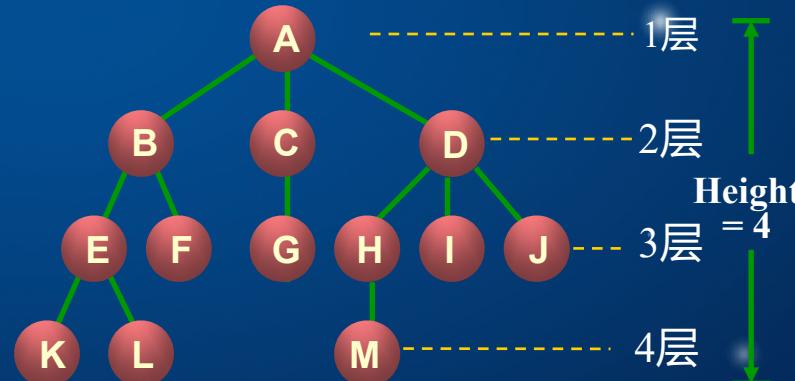
·**孩子**: 结点的子树的根[直接后继, 可能有多个]

·**双亲**: 孩子的直接前驱[最多只能有一个]

·**兄弟**: 同一双亲的孩子

·**子孙**: 以某结点为根的
树中的所有结点

·**祖先**: 从根到该结点所经
分支上的所有结点



二、常用的数据结构

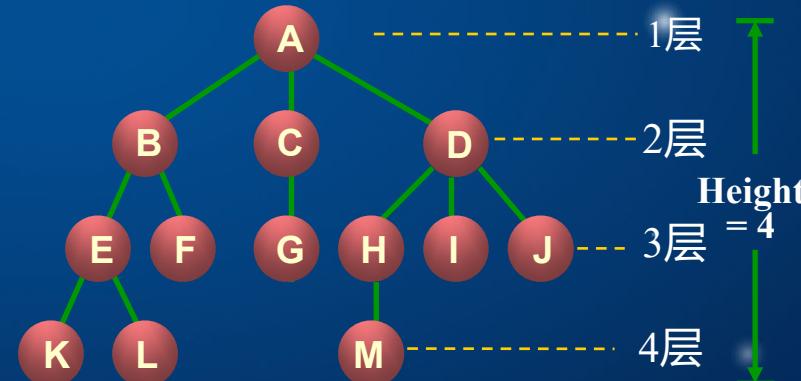
树的基本术语

·**层次**: 根结点为第一层, 其孩子为第二层, 依此类推

·**深度**: 树中结点的最大层次

·**有序树**: 子树之间存在确定的次序关系。

·**无序树**: 子树之间不存在确定的次序关系。



二、常用的数据结构

图

图 G 是由两个集合 $V(G)$ 和 $E(G)$ 组成的，记为 $G=(V, E)$

其中： $V(G)$ 是顶点的非空有限集合；

$E(G)$ 是边的有限集合，边是顶点的无序对或有序对。

二、常用的数据结构

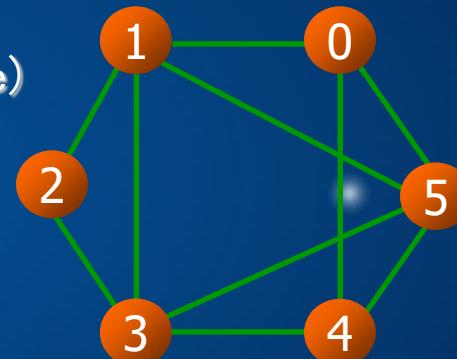
无向图

用 (x, y) 表示两个顶点 x, y 之间的一条边 (edge)

$N = \{V, E\}$,

$V = \{0, 1, 2, 3, 4, 5\}$,

$E = \{(0, 1), (0, 4), (0, 5), (1, 2), (1, 3), (1, 5), (2, 3), (3, 4), (3, 5), (4, 5)\}$



二、常用的数据结构

有向图

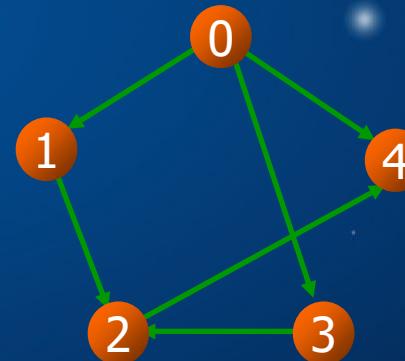
用 $\langle x, y \rangle$ 表示从 x 到 y 的一条弧(Arc)，且称 x 为弧尾， y 为弧头，

$$N = \{V, E\},$$

$$V = \{0, 1, 2, 3, 4\},$$

$$E = \{\langle 0, 1 \rangle, \langle 0, 3 \rangle, \langle 0, 4 \rangle, \langle 1, 2 \rangle,$$

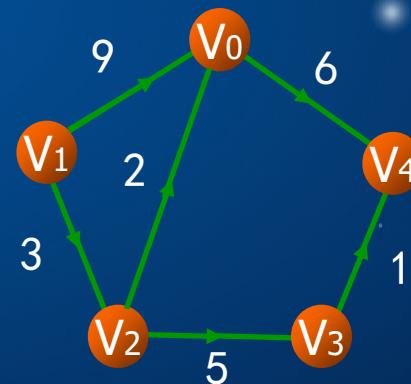
$$\langle 2, 4 \rangle, \langle 3, 2 \rangle\}$$



二、常用的数据结构

网

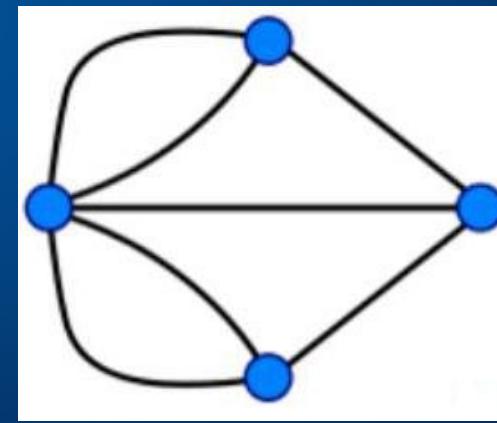
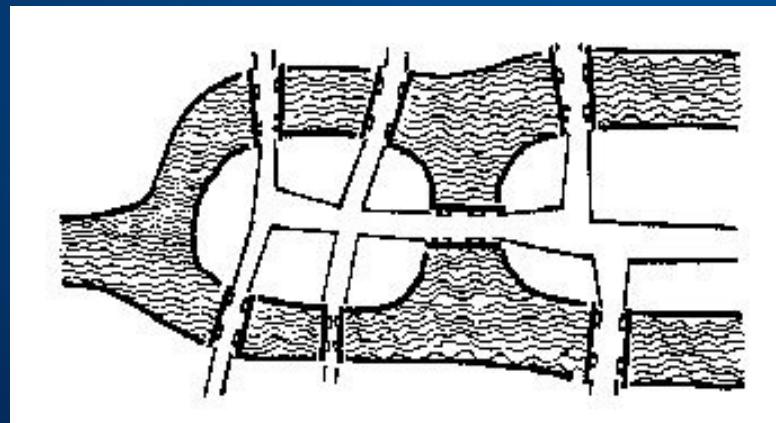
图的边具有与它相关的数值，这种与边相关的数叫做权。这些权可以表示从一个顶点到另一个顶点的距离或花费的代价。此类图又称为网。



二、常用的数据结构

七桥问题

两个岛与陆地间有七座桥，能否不重复走遍所有道路。



6.4 数据库系统

一、数据库系统基础

1、数据(Data)

狭义的理解：数字，如78、99.5、\$100

广义的理解：文字、图形、图像、声音、学生的档案记录、
货物的运输情况等

定义：描述事物的符号记录。

(2011130925, 王晓丰, 男, 199306, 江苏南京, 计算机学院)

2、数据库(DataBase, 简称DB)

数据库→存放数据的仓库

数据库：是为了实现一定的目的、按某种规则组织起来的数据集合。更准确的说，数据库就是长期存储在计算机内、有组织的、可共享的数据集合。

特点：集成性、共享性、海量性、持久性

序号	姓名	性别	出生年月	所属院系
1	张三	男	1986.5	信息学院计算机系
⋮				

3.数据库管理系统(DBMS DataBase Management System)

DBMS:

数据库管理系统是由一组程序构成，其主要功能是完成对数据库中数据定义、数据操纵，提供给用户一个简明的应用接口，实现事务处理等。

常见的DBMS有Oracle、DB2、SQL Server、My SQL、FoxPro和Access等等。

建立、使用和维护数据库的软件。

其基本功能：

对象定义功能	数据对象的定义，如库、表、视图、索引等
数据操纵功能	数据对象基本操作，如插入、修改、查询等
运行管理功能	数据对象统一控制，如数据安全、并发控制
系统维护功能	数据对象的输入、转换、转储、通信等

二、关系模型(二维表)

关系模型：是用二维表结构来表示实体以及实体之间的联系。

关系模型是以关系数学理论为基础，其操作的对象和结果都是二维表，这种二维表就是关系。

学号	姓名	性别	出生日期	院别
2000011001	黄小东	男	22/06/87	经济学院
2000011002	蔡晓芬	女	04/01/87	经济学院
2000014011	刘远峰	男	11/09/87	信息工程学院

关系模型的基本概念：

关系：一个关系对应一张二维表，对关系的描述称为关系模式，一个关系模式对应一个关系结构

元组(记录)：表中的一行

属性(字段)：表中的一列

域：属性的取值范围

关系名(字段1, 字段2, …, 字段n)

学生(学号, 姓名, 性别, 出生年月, 院系)

关系模型的基本概念：

关键字：可唯一标识一个记录的属性或属性组

候选码(多个)、主码或关键字(一个)

外关键字：如果表中的一个字段不是本表的主关键字，而是另外一个表的主关键字和候选关键字，这个字段(属性)就称为外部关键字。

关系名(字段1, 字段2, …, 字段n)

学生(学号, 姓名, 性别, 出生年月, 院系)

关系模型的基本概念：

两个实体之间的联系可分为如下3种类型。

(1) 一对联系 (1:1)



(2) 一对多联系 (1:n)



(3) 多对多联系 (m:n)



1:1联系

1:n联系

m:n联系

关系模型的基本概念：

两个实体之间的联系可分为如下3种类型。

(1) 一对联系 (1:1)。如果对于实体集 A 中的每一个实体，实体集B中至多有一个实体与之联系，反之亦然，则称实体集A与实体集B具有一对联系，记为1 : 1。

例如，电影院中观众和座位之间、乘车旅客和车票都是一对一的联系。

关系模型的基本概念：

两个实体之间的联系可分为如下3种类型。

(2) 一对多联系 (1:n)。如果对于实体A中的每一个实体，实体集B中有n个实体 ($n \geq 0$) 与之联系，反之，对于实体B中的每一个实体，实体集A中至多只有一个与之相联系，则称实体集A与实体集B有一对多联系。记为1 : n。

例如，公司对部门之间、班级对学生之间都是一对多联系。

关系模型的基本概念：

两个实体之间的联系可分为如下3种类型。

(3) 多对多联系 ($m:n$)。如果对于实体A中的每一个实体，实体集B中有 n 个实体 ($n \geq 0$)与之联系，反之，对于实体集B中的每一个实体，实体集A中也有 m 个实体 ($m \geq 0$) 与之联系，则称实体集A与实体集B具有多对多联系。记为 $m:n$ 。

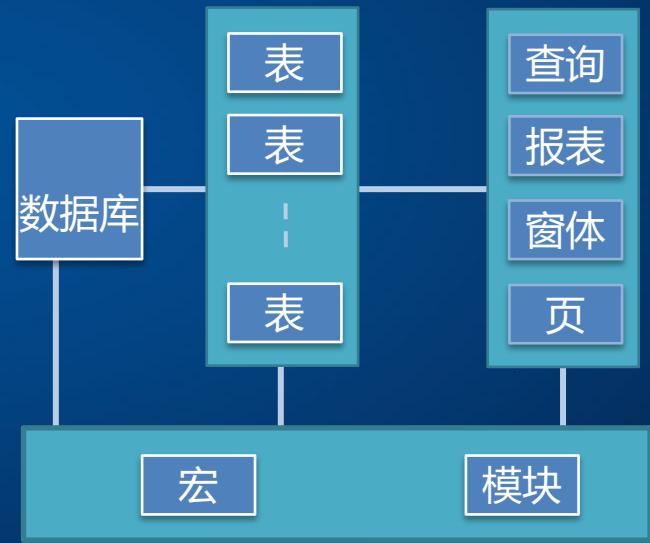
例如，课程与学生之间、图书与读者之间都是多对多的联系。

三、数据库管理系统

Access是一种关系型数据库管理系统。

Access中各对象之间的关系

不同的数据库对象在数据库中起着不同作用，其中表是数据库的核心与基础，存放数据库中的全部数据。报表、查询和窗体都是从表中获得数据信息，以实现用户的某一特定的需求，例如查找、计算统计和打印等。窗体可以提供一种良好的用户操作界面，通过它可以直接或间接地调用宏或模块，并执行查询、打印、预览、计算等功能，甚至可以对数据库进行编辑修改。



三、数据库管理系统

数据表

- 表是指一张满足关系模型的二维表，用于存储有关特定实体的数据集合。
- 它由表名、表中的字段、表的主关键字以及表中的具体数据组成。
- 表中的每一列称为一个“字段”。每个字段包含某一专题的信息。例如，在图书表中，“书名”和“书号”是表中所有行共有的属性，包含这两类信息的列分别称为“书名”字段和“书号”字段。
- 表中的每一行称为一个“记录”，每一个记录包含这行中的所有信息，就像在图书表中某本图书的全部信息。

学生登记表s

学号	姓名	性别	出生年月	系别
200510101	李丽	女	07/05/1987	计算机
200510102	赵勇	男	02/23/1988	法律
200510103	王凯	男	01/19/1987	管理

三、数据库管理系统

查询

查询是Access数据库的重要对象，是用户按照一定条件从Access数据库表或已建立的查询中检索需要数据的最主要方法。

- 选择查询
- 交叉表查询
- 参数查询
- 操作查询
- SQL查询

三、数据库管理系统

SQL查询

SQL查询是直接应用SQL语言执行查询任务的一种查询。SQL语言中的查询使用SELECT语句来执行的。

SELECT语句的一般格式是：

SELECT 字段名列表

FROM 基本表或视图

[WHERE 条件表达式]

[GROUP BY 列名1 [HAVING 内部函数表达式]]

[ORDER BY 列名2 [ASC/DESC]]

三、数据库管理系统

SQL查询

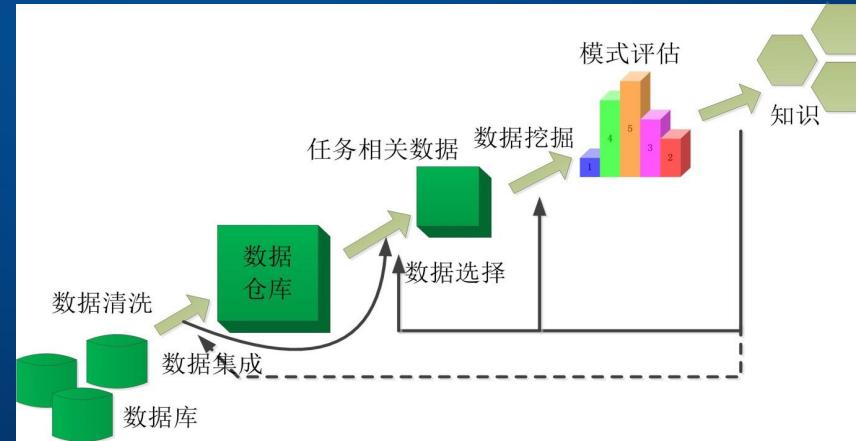
[例6-1]用SQL语句从“读者”表中查询“性别”为“男”的所有字段，并按借书证号升序排列。

[解] `SELECT * FROM 读者 WHERE 性别="男" ORDER BY 借书证号。`

SELECT语句的含义是：根据WHERE子句中的条件表达式，从表或视图中找出满足条件的记录集，按SELECT子句中的目标列，选出记录集中的分量形成结果表。如果有ORDER子句，则结果表要根据指定的列按升序或降序排序。GROUP子句将结果按列名分组，每个组产生结果表中的一个记录集。

三、数据挖掘

从大量数据中挖掘或抽取出知识，又称为数据库中知识发现，是一个从大量数据中抽取挖掘出未知的、有价值的模式或规律等知识的过程。



三、数据挖掘

- ◆ 数据清洗，清除数据噪声和无关的数据；
- ◆ 数据集成，将多数据源中的数据组合到一起；
- ◆ 数据转换，将数据转换为易于挖掘的存储形式；
- ◆ 数据挖掘，利用智能方法挖掘数据模式或规律知识；
- ◆ 模式评估，挖掘结果筛选出有意义的模式知识；
- ◆ 知识表示，利用可视化技术向用户展示挖掘出相关知识

啤酒与尿布

“啤酒与尿布”的故事是营销界的神话，“啤酒”和“尿布”两个看上去风马牛不相及的商品摆放在一起进行销售、并获得了很好的销售收益，这看起来像是一个笑话，但是实际上这一直被商家所津津乐道的发生在美国沃尔玛连锁超市的真实案例。在美国，妇女通常在家照顾孩子，她们经常会嘱咐丈夫在下班回家路上去超市给孩子买尿布，而丈夫在买尿布的同时又会顺手购买自己爱喝的啤酒。发现这个有趣的现象后把两种商品摆放在一起，使尿布和啤酒的销量大幅增加。



6.5 软件工程

一、软件工程概述

软件危机是指在计算机软件的开发和维护过程中所遇到的一系列严重问题。

6.5 软件工程

软件危机的具体表现：

- ①软件开发的进度难以控制，完成期限一再拖延的现象
- ②软件成本严重超标
- ③软件需求在开发初期不明确，导致矛盾在后期集中暴露从而对整个开发过程带来灾难性的后果
- ④由于缺乏完整规范的资料，加之软件测试不充分，从而造成软件质量低下，运行中出现大量问题

6.5 软件工程

二、软件工程定义

定义：软件工程是为了经济地获得能够在实际机器上有效运行的可靠软件而建立和使用的一系列完善的工程化原则。

1983年，IEEE给出定义为：软件工程是开发、运行、维护和修复软件的系统方法。

软件工程的重要思想：强调在软件开发过程中需要应用工程化原则的重要性。

6.5 软件工程

三、软件生存周期

1. 制订计划
2. 需求分析
3. 软件设计
4. 程序编码
5. 软件测试
6. 运行和维护

是指软件产品从考虑其概念开始到该产品交付使用，直至最终退役为止的整个过程，一般包括计划、需求分析、软件设计、程序编码、软件测试、运行和维护等阶段。

6.5 软件工程

四、软件工程方法

- 结构化方法
- 模块化方法
- 面向数据结构方法
- 面向对象方法
- 基于构件的软件开发方法

面向数据结构开发的基本思想是基于构件的软件开发方法 (CBSD) 是一种基于预先开发好的软件构件，通过将其集成组装的方式来开发软件系统的方法。又称基于构件的软件工程，它是软件复用的实现方式之一。其根本目的仍然是为了提高软件开发的质量和效率。提供了一种自底向上的、基于预先定制包装好的类属元素 (构件) 来构作应用系统的途径。

6.6 人工智能

一、人工智能基础

人工智能(Artificial Intelligence): 关于人造物的智能行为, 而智能行为包括知觉、推理、学习、交流和在复杂环境中的行为。

研究途径:

- (1) 从大脑的神经元模型着手研究, 搞清大脑信息处理过程的机理。
- (2) 从模拟人脑功能的角度来实现人工智能, 通过计算机程序的运行, 达到和人类智能行为活动过程相类似作为研究目标。

6.6 人工智能

一、人工智能基础

人工智能的研究内容：

模式识别

问题求解

自然语言处理

自动定理证明

自动程序设计

专家系统

机器感知

机器人学

模式识别是人工智能最早的

トロント市議会議員選挙（2014年）

我们可以把机器人定义为一种可以编程序的多功能操作装置。机器人可以代替人从事有害环境中的危险工作，可以提高我们的工作质量和效率。

6.6 人工智能

二、人工神经网络

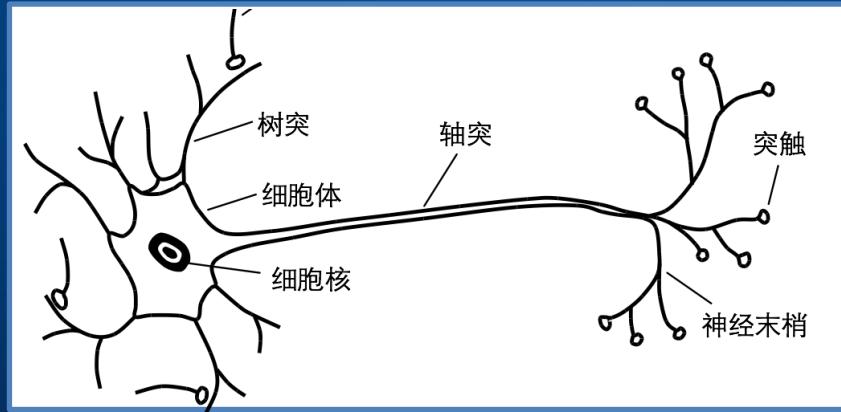
人工神经网络(Artificial Neural Network):通过模拟人脑感知行为出发，基于神经元间的连接来实现感知信息的大规模并行、分布式存储和处理，并提供自组织、自适应和自学习能力。

人工神经网络由许多单个的处理器以模仿活的生物体神经元网络的方式构成，这些处理器称为处理单元。

6.6 人工智能

二、人工神经网络

一个生物神经元是一个单个细胞，具有一些称为树突的输入触角和一个称作轴突的输出触角。



- 树突从其他细胞的轴突通过称为突触的小间隙采集信号。
- 轴突传递的信号反映了细胞是处于抑制状态还是兴奋状态。这种状态由细胞的树突接收到的信号的合成来决定。
- 这些一个突触的传导是由突触的化学成分控制的。

6.6 人工智能

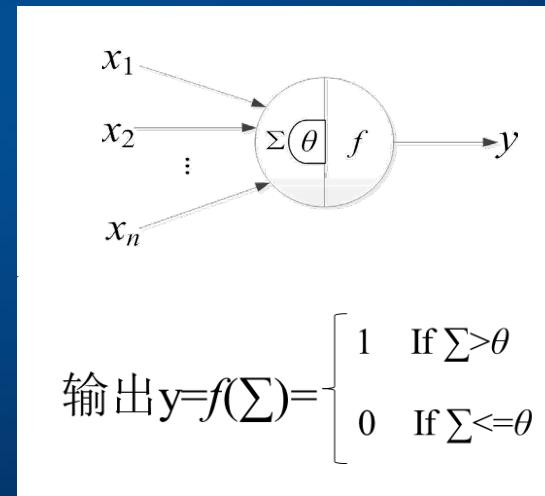
人工神经网络的一个处理单元是模仿对生物神经元这种基本了解的一个简单装置。这个装置是一个多输入单输出的非线性阈值器件。

输入: $x_1, x_2 \dots x_n$

权值: $w_1, w_2 \dots w_n$

$$\sum = x_1 w_1 + x_2 w_2 + \dots + x_n w_n$$

单元阈值: θ



6.6 人工智能

人工神经元与生物神经元作用对比

生物神经元	人工神经元	作用
树突	输入层	接收输入信号
细胞体	加权和	加工处理信号
轴突	阈值函数	控制输出
突触	输出层	输出结果

6.6 人工智能

三、机器人学

机器人学 (Robotics) 就是研究具有智能性并且物理上自主的智能体的一门学科

研究重点：

- (1) 移动性
- (2) 智能性



本章小结

- 计算机软件的概念、层次结构、软硬件关系
- 操作系统的概念、功能、进程、
- 数据结构的相关知识
- 数据库定义、关系模型、数据挖掘概念
- 软件工程概念、软件生存周期、软件工程方法
- 人工智能与机器人学