第一章数据结构与算法

1.1算法

算法是指解题方案的准确而完整的描述.

算法不等于程序或计算机方法,特征包括： ⑴可行性；(2)确定性；(3)有穷性；(4)拥有足够的情报.

算法的基本要素:一是对数据对象的运算和操作;二是算法的控制结构.

指令系统:一个计算机系统能执行的所有指令的集合.

基本运算和操作包括:算术运算，逻辑运算，关系运算，数据传输.

算法的控制结构:顺序结构，选择结构，循环结构.

算法基本设计方法:列举法,归纳法，递推，递归,减斗递推技术,回溯法.

算法复杂度:算法时间复杂度和算法空间复杂度.

算法时间复杂度是指执行算法所需要的计算工作量.(1)平均性态分析(2)最坏情况复杂性

算法空间复杂度是指执行这个算法所需要的内存空间.

1.2数据结构

数据结构研究的三个方面： (1)数据的逻辑结构； (2)数据的存储结构； (3)对数据结构进行的运算.

数据结构是指相互有关联的数据元素的集合.

数据的逻辑结构包含： (1)表示数据元素的信息； (2)表示各数据元素之间的前后件关系.

数据的存储结构有顺序，链接,索引等.

线性结构条件： (1)有且只有一个根结点 (2)每一个结点最多有一个前件，也最多有一个后件.

非线性结构:不满足线性结构条件的数据结构.

1.3线性表及其顺序存储结构

线性表由一组数据元素构成，元素位置只取决于自己的序号，元素之间的相对位置是线性的.

由若干项数据元素组成的数据元素称为记录，而由多个记录构成的线性表又称为文件.

非空线性表的结构特征： (1)有且只有一个根结点，无前件； (2) 有且只有一个终端结，无后件； (3)除根结点与终端结点外，其他所有结点有且只有一个前件,也有且只有一个后件.

结点个数n称为线性表的长度，当n=0时，称为空表.

线性表两个基本特点： (1)线性表的存储空间是连续的； (2)线性表的元素是按逻辑顺序依次存放的.

顺序表的运算:插入，删除，查找，排序，分解，复制和逆转等操作.

1.4栈和队列

栈是限定在一端进行插入与删除的线性表，栈遵循**先进后出**原则

用top表示栈顶位置，允许插入与删除，用bottom表示栈底，不允许插入与删除

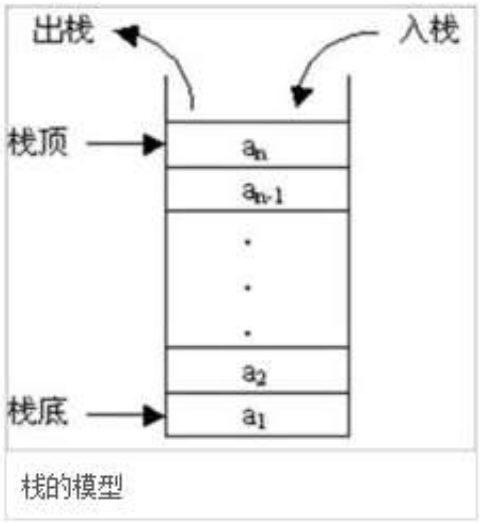
栈的基本运算:(1)入栈;(2)退栈;(3)读栈顶元素（**指针无变化**）

队列是指允许一端插入，另一端(队头)进行删除的线性表.队列是**先进先出**原则

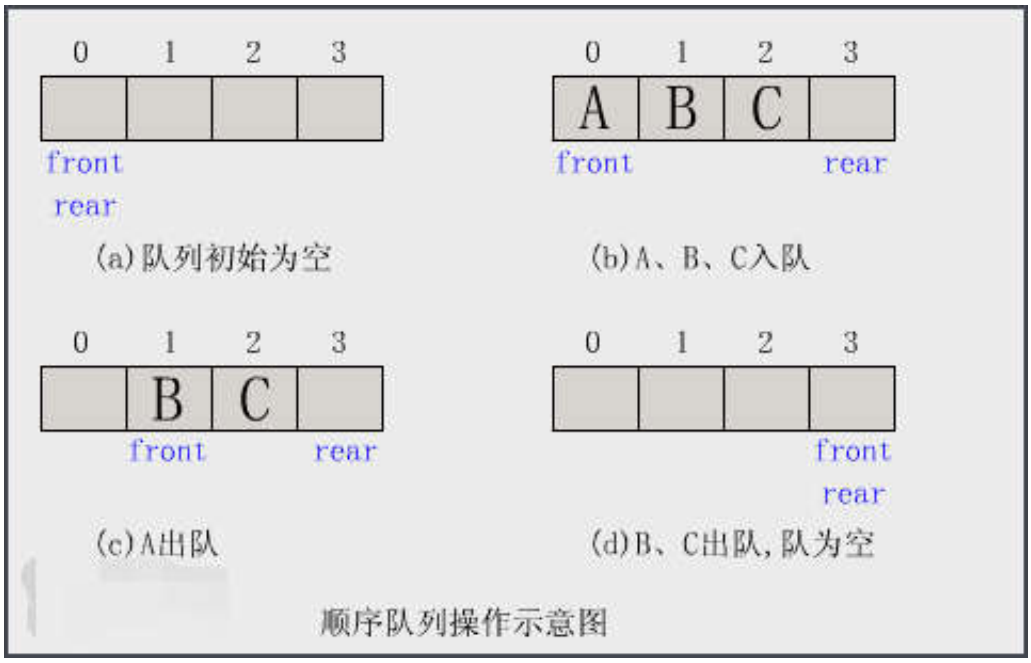
Rear指针指向队尾，front指针指向队头.

队列运算包括(1)入队；(2)退队

循环队列:s=0表示队列空,s=1且front=rear表示队列满



栈的模型



顺序队列操作示意图

1.5线性链表

数据结构中的每一个结点对应于一个存储单元,这种存储单元称为存储结点.

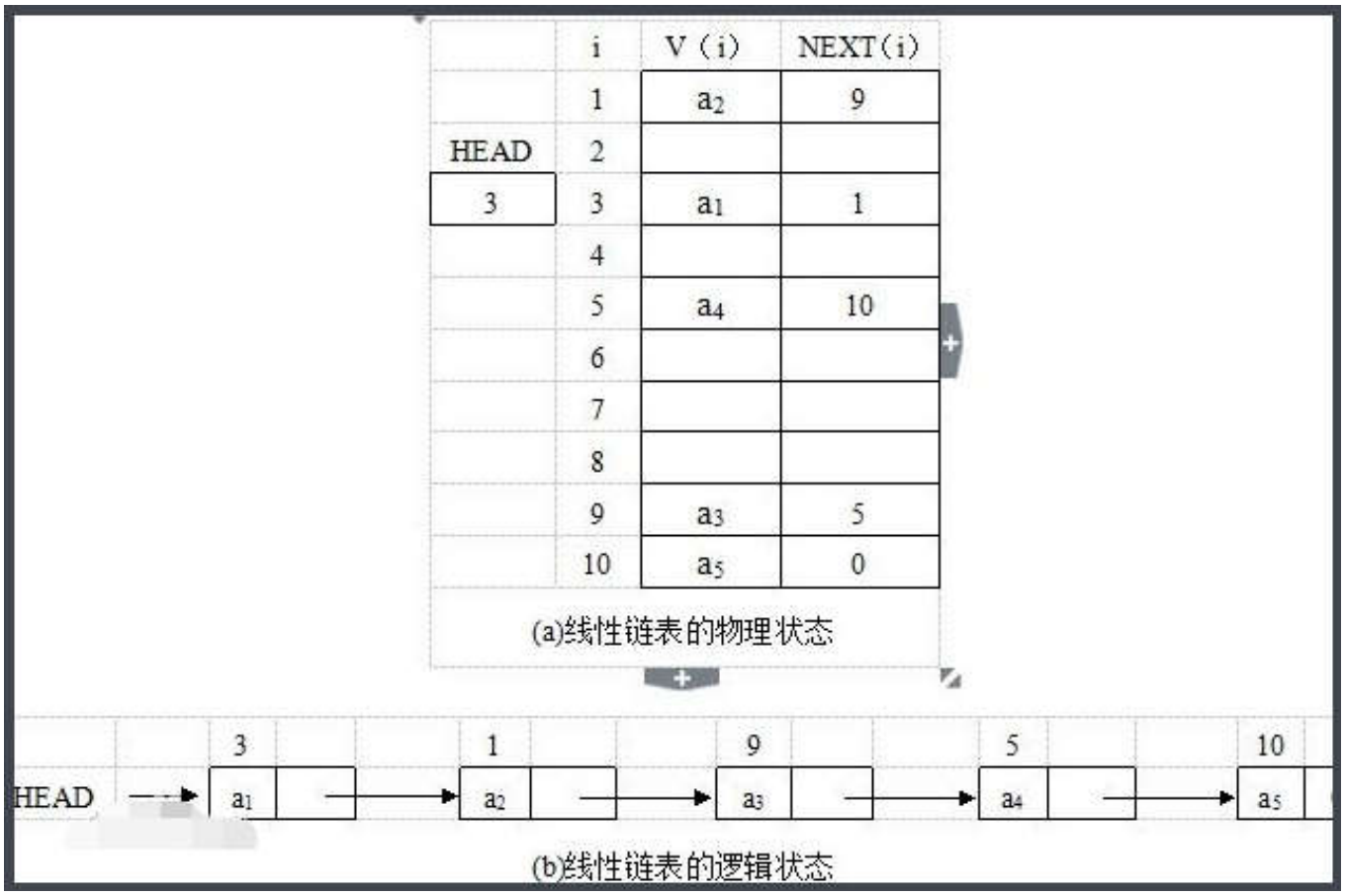
结点由两部分组成:(1)用于存储数据元素值,称为数据域;(2)用于存放指针，称为指针域

在链式存储结构中，存储数据结构的存储空间可以不连续，逻辑关系是由指针域来确定的.

链式存储方式即可用于表示线性结构，也可用于表示非线性结构.

线性链表,HEAD称为头指针,HEAD=NULL(或0)称为空表

线性链表的基本运算:查找，插入，删除.



	i	V (i)	NEXT(i)
	1	a_2	9
HEAD	2		
3	3	a_1	1
	4		
	5	a_4	10
	6		
	7		
	8		
	9	a_3	5
	10	a_5	0

(a)线性链表的物理状态

(b)线性链表的逻辑状态

1.6树与二叉树

树是一种简单的非线性结构,所有元素之间具有明显的层次特性.

每一个结点只有一个前件，称为父结点

没有前件的结点只有一个，称为树的根结点.

每一个结点可以有多个后件,称为该结点的子结点.

没有后件的结点称为叶子结点.

在树结构中，一个结点所拥有的后件的个数称为该结点的度，

二叉树的深度：从根节点到叶子节点最长路径的长度为树的深度。

二叉树的宽度：节点数最多的那一层的节点数为树的宽度。

二叉树的特点:(1)非空二叉树只有一个根结点;(2)每一个结点最多有两棵子树.

二叉树的基本性质： (1)在二叉树的第k层上，最多有2k-1个结点；

(2)深度为m的二叉树最多有2m-1个结点；

(3)度为0的结点总是比度为2的结点多一个;N=N0+N1+N2 ;N0=N2+1

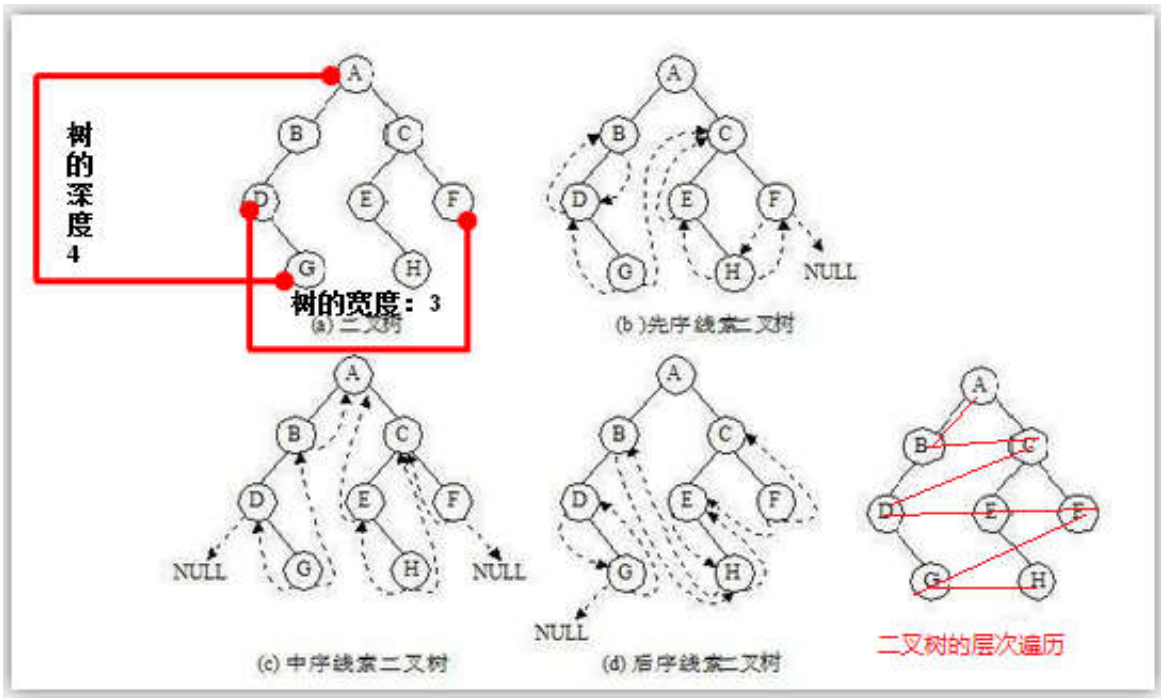
二叉树存储结构采用链式存储结构.

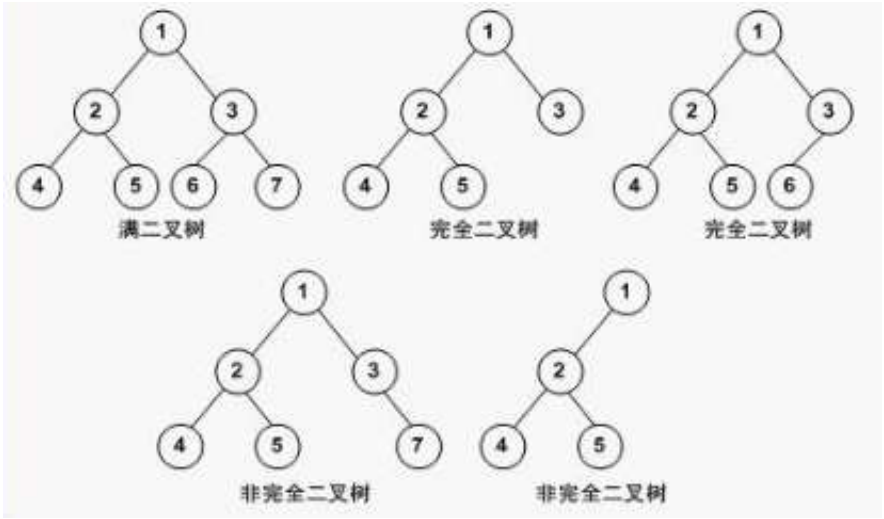
二叉树的遍历：

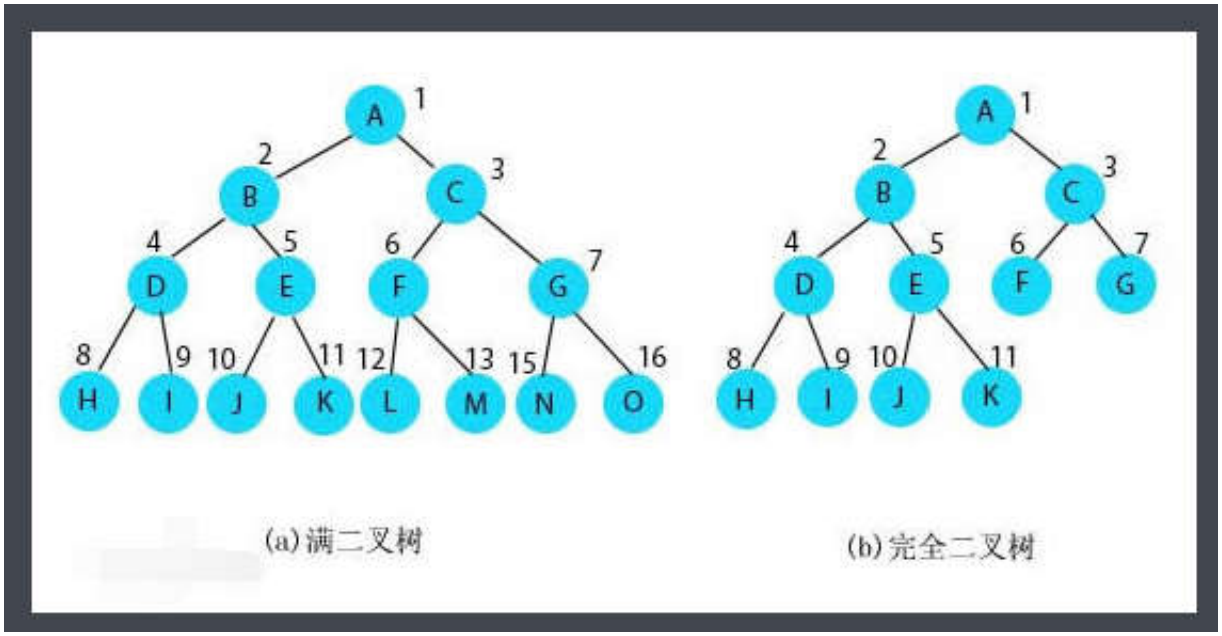
(1)前序遍历(DLR),首先访问根结点，然后遍历左子树，最后遍历右子树；

(2)中序遍历(LDR),首先遍历左子树，然后访问根结点，最后遍历右子树；

(3)后序遍历(LRD)首先遍历左子树,然后访问遍历右子树，最后访问根结点.







1.7查找技术

顺序查找的使用情况：

(1)线性表为无序表；

(2)链表采用链式存储结构.

二分法查找只适用于顺序存储的有序表，对于长度为n的有序线性表，最坏情况只需比较log2^t

1.8排序技术

排序是指将一个无序序列整理成按值非递减顺序排列的有序序列.

交换类排序法: (1)冒泡排序法，需要比较的次数为n(n-1)/2； (2)快速排序法. 最坏为n*(n-1)/2

插入类排序法: (1)简单插入排序法,最坏为n(n-1)/2次比较； (2)希尔排序法,最坏为O(n^1.5)次比较

选择类排序法: (1)简单选择排序法，最坏为n(n-1)/2次比较； (2)堆排序法,最坏为O(nlog2n)次比较

第二章程序设计基础

2.1程序设计设计方法和风格

良好的程序设计风格如下 1源程序文档化;2数据说明;3语句结构;4输入和输出.

序言性注释和功能性注释，语句结构清晰第一，效率第二.

2.2结构化程序设计

结构化程序设计方法的四条原则是:1.自顶向下;2.逐步求精;3.模块化;4.限制使用goto语句.

结构化程序的基本结构和特点： (1)顺序结构； (2)选择结构（分支）； (3)重复结构（循环）；

2.3面向对象的程序设计

面向对象方法的优点：

(1)与人类思维习惯一致（2)稳定性好（3)可重用性好 (4) 易于开发大型软件产品 （5)可维护性好

对象是可以用来表示客观世界中的任何实体，对象是实体的抽象.

对象由一组表示其静态特征的属性和可执行的操作组成.

属性即对象所包含的信息,操作描述了对象执行的方法或服务.

对象的基本特点： (1)标识惟一性;(2)分类性;(3)多态性;(4)封装性;(5)模块独立性好.

类是指具有共同属性，共同方法的对象的集合.对象是类的一个实例.

消息是2个实例之间传递的信息. 消息的组成包括(1)接收消息的对象(2)消息标识符也称消息名(3)参数.

继承是指能够直接获得已有的性质和特征,而不必重复定义.

继承分单继承和多继承.单继承指一个类只允许有一个父类,多继承允许有多个父类.Java支持单继承

多态性是指同样的消息被不同的对象接受时，产生不同的行为.

第三章软件工程基础

3.1软件工程基本概念

计算机软件是包括程序,数据及相关文档的完整集合.

软件的特点包括：

(1)软件是一种逻辑实体； (2)软件无明显的制作过程； (3)软件无磨损老化问题；

(4)软件受计算机系统限制； (5)软件复杂度高成本昂贵； (6)软件开发涉及诸多因素.

软件按功能分为应用软件，系统软件，支撑软件(或工具软件).

软件危机主要表现在成本，质量,生产率等问题.

软件工程是指计算机软件的定义，开发和维护的整套方法.

软件工程包括3个要素:方法，工具和过程.

软件工程过程是把软件转化为输出的一组活动，包含4种基本活动：

(1)P——软件规格说明； (3)C一软件确认； （4)A——软件演进.（2)D——软件开发；

软件周期:软件产品从提出，实现，使用，维护到退役的过程.

软件生命周期三个阶段:软件定义，软件开发，运行维护

主要活动阶段：

(1)可行性研究与计划制定;(2)需求分析;(3)软件设计;(4)软件实现;(5)软件测试;(6)运行和维护.

软件工程的目标和与原则：

目标:在给定成本,进度的前提下，开发出满足用户需求的产品.

基本目标:付出较低的开发成本;及时交付使用;达到要求的软件功能.

软件工程的理论研究内容:软件开发技术和软件工程管理.

软件开发技术包括:开发方法，开发过程，开发工具和开发环境.

软件工程管理包括:软件管理学，软件工程经济学，软件心理学等内容.

软件管理学包括人员组织，进度安排，质量保证，配置管理，项目计划等.

软件工程原则：抽象，信息隐蔽,模块化,局部化,确定性,一致性，完备性和可验证性.

3.2结构化分析方法

结构化方法的核心和基础是结构化程序设计理论.

需求分析方法有(1)结构化需求分析方法;（2)面向对象的分析的方法.

从需求分析建立的模型的特性来分:静态分析和动态分析.

结构化分析方法的实质:着眼于数据流，自顶向下，逐层分解,建立系统的处理流程

结构化分析的常用工具(1)数据流图;(2)数据字典;(3)判定树;(4)判定表.

数据流图:描述数据处理过程的工具，是需求理解的逻辑模型的图形表示，它直接支持系统功能建模.

数据字典:对所有与系统相关的数据元素的一个有组织的列表，以及精确的，严格的定义

判定树:从问题定义的文字描述中分清哪些是判定的条件，哪些是判定的结论，得出从属关系

判定表:与判定树相似，当数据流图中的加工要依赖于多个逻辑条件的取值，使用判定表描述比较适宜

数据字典是结构化分析的核心.

软件需求规格说明书的特点：

⑴正确性;(2)无岐义性;(3)完整性;(4)可验证性;(5)—致性;(6)可理解性;(7)可追踪性.

3.3结构化设计方法

软件设计的基本目标是确定目标，如何完成预定的任务。

软件设计是确定系统的物理模型.

软件设计包括软件结构设计，数据设计，接口设计，过程设计.

结构设计:定义各主要部件之间的关系.

数据设计:将模型转化为数据结构的定义.

接口设计:描述软件内部如何通信.

过程设计:把系统结构部件转换成软件的过程.

从工程管理角度分为:概要设计和详细设计.

优秀的软件**高内聚,低耦合.**

软件概要设计的基本任务是：

(1)设计软件系统结构;（2)数据结构及数据库设计； (3)编写概要设计文档;（4)概要设计文档评审.

模块用一个矩形表示,箭头表示模块间的调用关系.

带实心圆的箭头表示传递的是控制信息，空心圆箭头表示传递的是数据.

结构图的基本形式:顺序形式，重复形式，选择形式.

结构图有四种模块类型:传入模块，传出模块,变换模块和协调模块.

典型的数据流类型有两种:变换型和事务型.

变换型系统结构图由输入，中心变换，输出三部分组成.

事务型数据流的特点是:接受一项事务,分派一个适当的处理单元，然后给出结果.

详细设计:选定的表达工具来表示算法和数据结构的细节.

常见的过程设计工具有:

图形工具(程序流程图、N—S、PAD、HIPO),表格工具(判定表),语言工具(PDL过程设计语言).

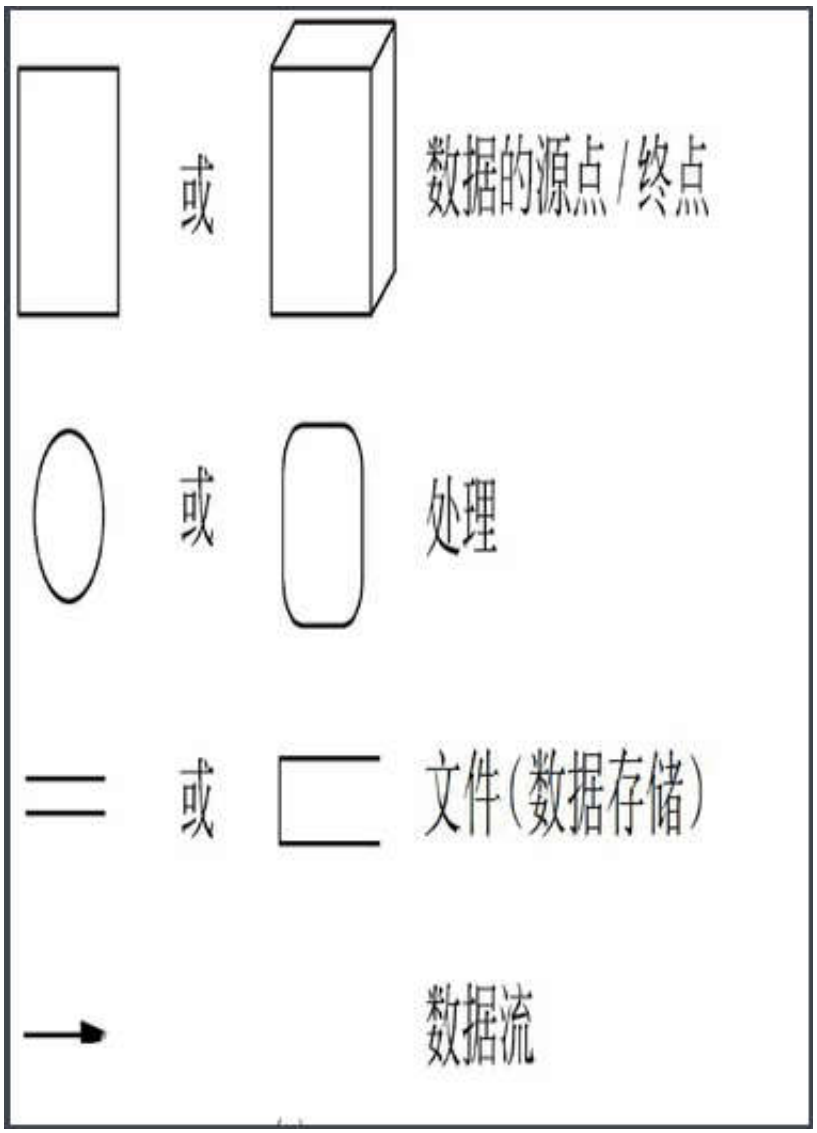
程序流程图是一种传统的、应用广泛的软件过程设计表示工具，也称程序框图。

数据流程图中有以下几种主要元素：

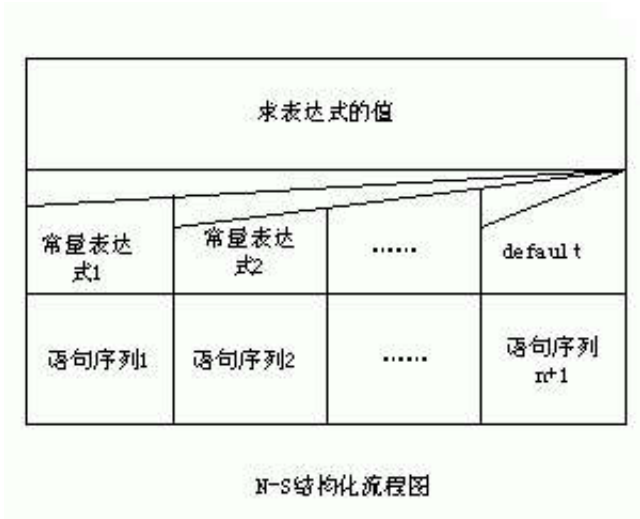
数据流是数据在系统内传播的路径。

数据源代表系统之外的实体，如人。

数据存储表示信息的静态存储。

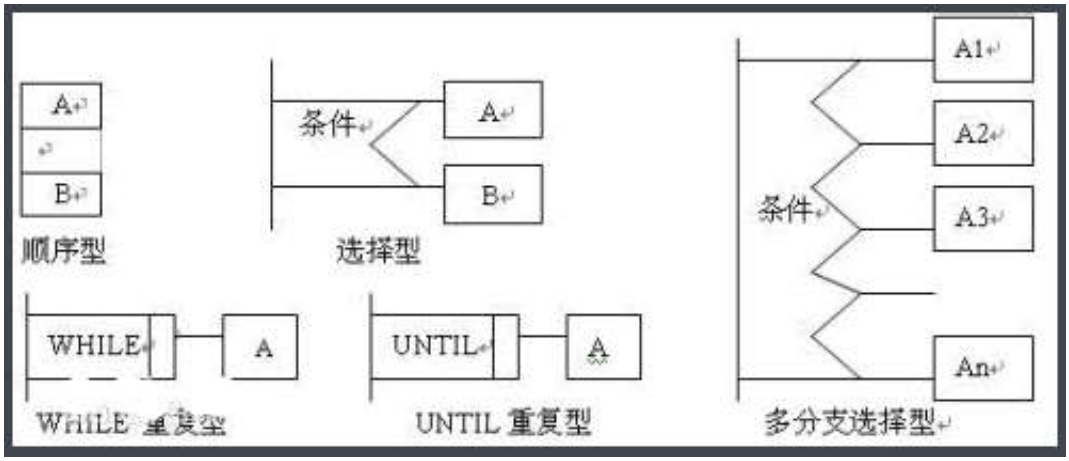


N—S图：方框图。



N-S结构化流程图

PAD图



HIPO

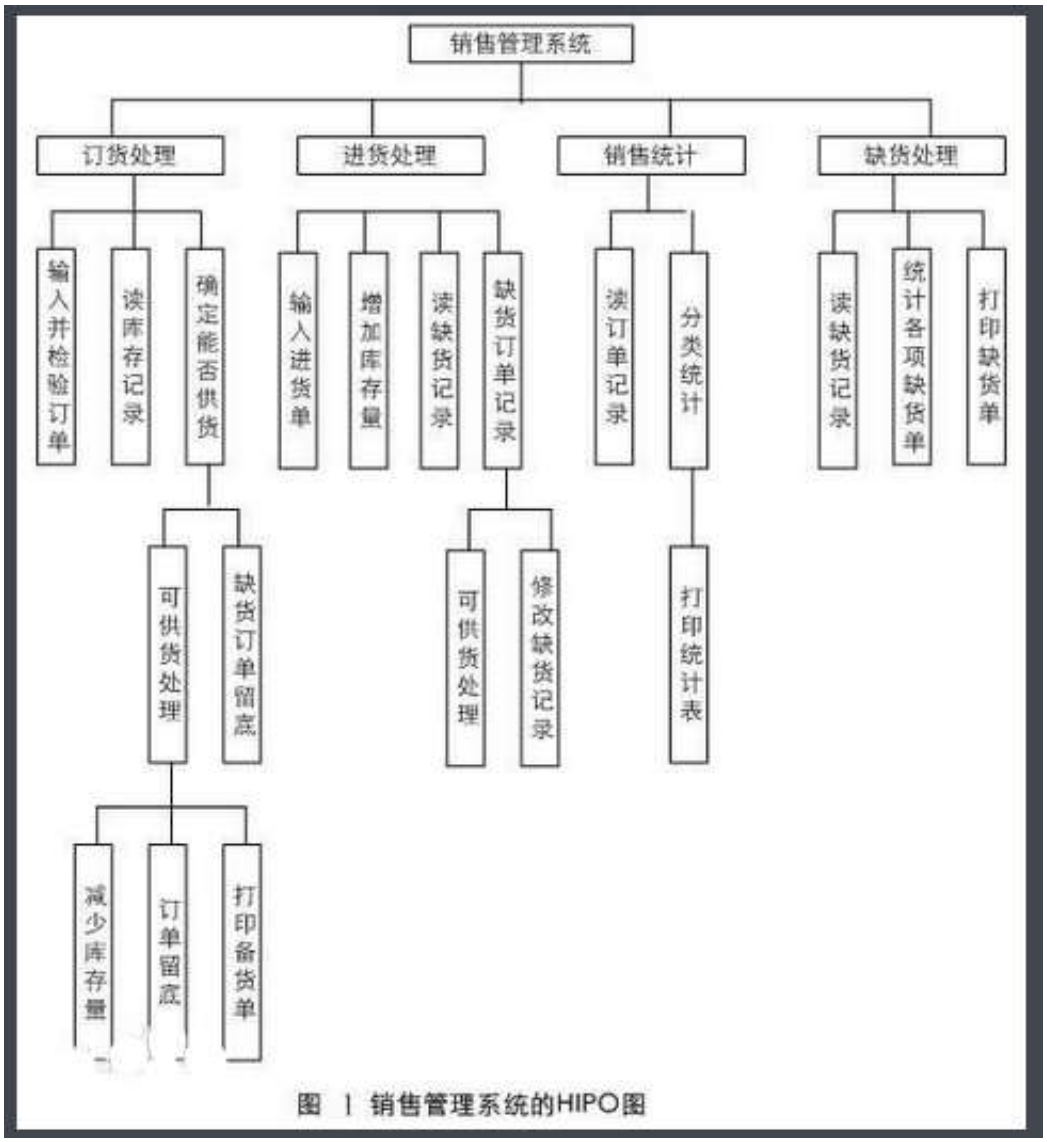
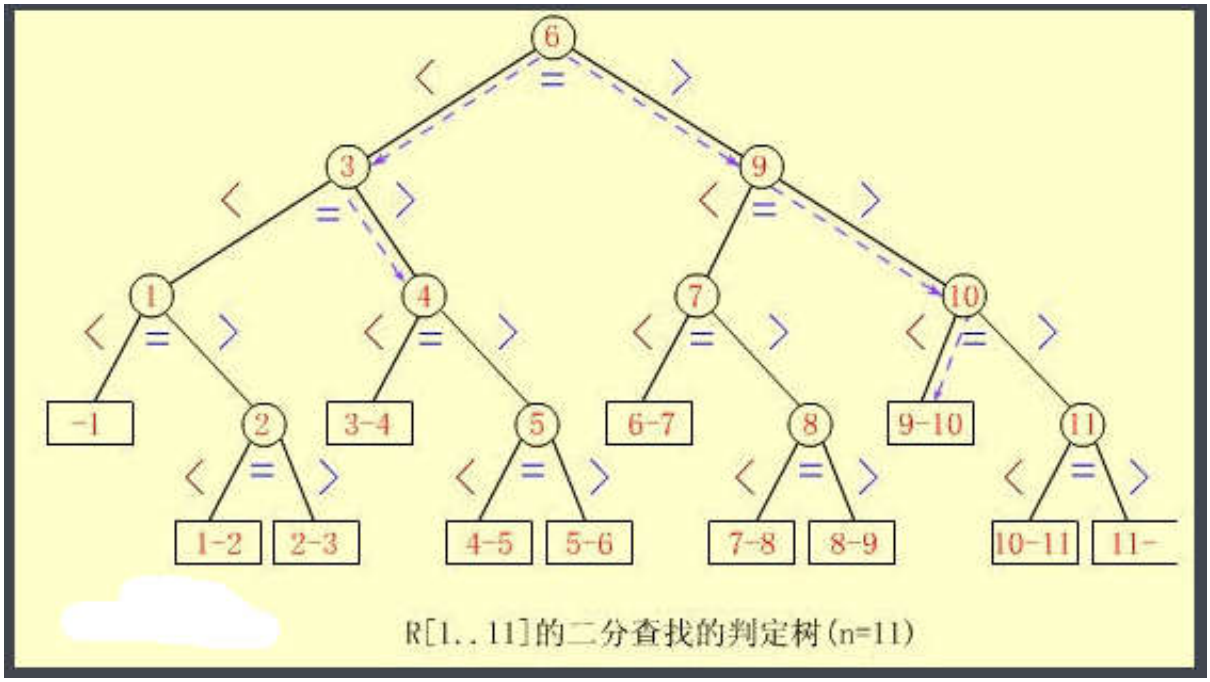


图 1 销售管理系统的HIPO图

判定表

		1	2	3
条件	赊欠情况	>60天	>60天	≤60天
	发货单金额	>$500	≤$500	——
操作	不发出批准书	√		
	发出批准书 发出发货单		√	√
	发出赊欠报告		√	

判定树



R[1..11]的二分查找的判定树(n=11)

3.4软件测试

软件测试定义:检验预期结果与实际结果之间的差别.

软件测试的目的:**发现错误**而执行程序的过程.

软件测试方法:静态测试和动态测试.

静态测试包括代码检查,静态结构分析,代码质量度量.

动态测试:是最基本的测试，主要包括白盒测试方法和黑盒测试方法.

白盒测试（结构测试或逻辑驱动测试）：在程序内部进行,主要用于完成软件内部操作的验证.

主要方法有逻辑覆盖,基本路径测试.

逻辑覆盖：指一系列以程序内部的逻辑结构为基础的测试用例设计技术。

1.语句覆盖：每个语句至少被执行一次。

2.路径覆盖：所有可能的路径至少经历一次。

3.判断覆盖：每个判断的每个取值分支至少经历一次

4.条件覆盖：每个判断的每个条件的可能取值至少执行一次。

5.判断一条件覆盖：每个条件的所有可能取值至少执行一次，同时每个判断分支至少执行一次。

基本路径测试：根据软件过程性描述中的控制流程确定程序的环路复杂性度量。

黑盒测试（功能测试或数据驱动测试）：是对软件已实现的功能是否满足需要进行测试和验证。

主要方法有等价类划分法，边界值分析法，错误推测法，因果图等.

等价类划分法：是将程序的所有可能的输入数据划分成若干部分，然后从每个等价类中选取数据作为测试用例。包括1有效等价类，2无效等价类

边界值分析法：对各种输入、输出范围的边界情况设计测试用例的方法。

程序错误最容易出现在输入或输出范围的边界处。

错误推测法：以经验和直觉推测程序中可能存在的各种错误，有针对性地编写检查这些错误的例子。

软件测试过程一般按4个步骤进行:单元测试，集成测试，验收测试(确认测试)和系统测试.

1单元测试是对软件设计的最小单位模块进行正确性检测的过程

依据：**详细设计说明书和源程序**。

2集成测试是测试和组装软件的过程。主要目的是发现与结构有关的错误。

依据：**概要计算机说明书**。

集成测试涉及的内容包括软件单元的接口测试、全局数据结构测试、边界条件和非法输入的测试等。

集成测试时将模块组成程序通常采用两种方式：非增量方式组装（一次性组装）与增量方式组装。

3验收测试任务是验证软件的功能和性能是否满足了规格说明书中确定的各种需求。

4系统测试是在实际运行环境下对计算机系统进行集成测试和确认测试。

系统测试包括：功能测试、性能测试、操作测试、配置测试、外部接口测试和安全性测试。

3.5程序的调试

程序调试的任务是改正程序中的错误.

程序调试分两部分：定位错误；排除错误。

程序调试的基本步骤： (1)定位错误;(2)修改代码;(3)回归测试

软件调试可分表静态调试和动态调试.

静态调试主要是指通过人的思维来分析源程序代码和排错

动态调试是辅助静态调试.主要调试方法有：(1)强行排错法;(2)回溯法;(3)原因排除法.

第四章 数据库设计基础

4.1数据库系统的基本概念

数据:描述事物的符号记录.

数据库:是数据的集合，存储在硬盘上.

数据库管理系统:一种**系统软件**，负责数据库中的数据增删改查.

数据库管理系统功能：

(1)数据定义(2)数据存取(3)数据操纵(4)数据完整性(5)数据库恢复(6)数据的服务

数据库管理系统提供以下的数据语言：

(1)数据定义语言DDL:负责数据的模式定义与数据的物理存取构建；

(2)数据操纵语言DML:负责数据的操纵，如增删查改等；

(3)数据控制语言DCL:负责数据完整性，故障恢复等.

数据语言按其使用方式具有两种结构形式:交互式命令与宿主型语言

数据库管理员DBA，数据库系统DBS，数据库应用系统DBAS，数据库管理系统DBMS

数据库系统发展三个阶段：

文件系统阶段:提供了简单的数据共享与数据管理能力，它无统一管理和数据共享能力.

层次数据库与网状数据库系统阶段：为统一与共享数据提供了有力支撑.

关系数据库系统阶段：

数据的集成性，高共享性与低冗余性，数据独立性(物理独立性与逻辑独立性),数据统一管理与控制.

物理独立性：存储结构、存取方式的改变，不影响数据库的逻辑结构。

逻辑独立性：逻辑结构的改变如修改数据、增加数据类型等，不需要相应修改应用程序。

数据的统一管理与控制 1数据的完整性检查2数据的安全性保护3并发控制

数据库系统的三级模式：

(1)概念模式:数据库系统中全局数据逻辑结构的描述，全体用户公共数据视图；

(2)外模式:也称子模式或用户模式.是用户的数据视图，也就是用户所见到的数据模式；

(3)内模式:又称物理模式，它给出了数据库物理存储结构与物理存取方法.

内模式处于最底层，反映了数据在计算机物理结构中的实际存储形式；

概念模式处于中层，反映了设计者的数据全局逻辑要求；

处模式处于最外层，反映了用户对数据的要求。

数据库系统的两级映射： (2)外模式到概念模式的映射. (1)概念模式到内模式的映射；

4.2数据模型

数据模型的概念:是数据特征的抽象，描述了数据结构,数据操作及数据约束.

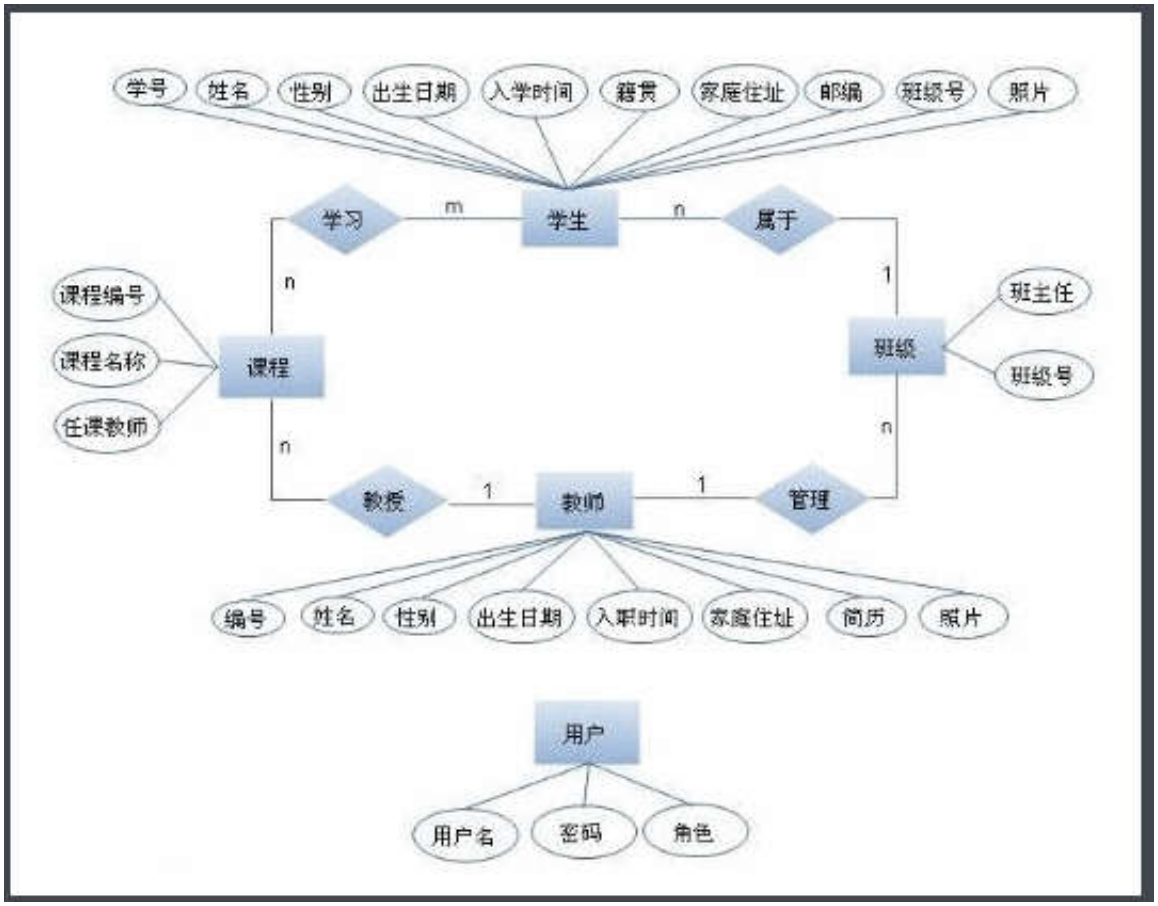
E-R模型的基本概念

矩形框：表示实体，在框中记入实体名。

菱形框：表示联系，在框中记入联系名。

椭圆形框：表示实体或联系的属性，将属性名记入框中。

连线：实体与属性之间；



E-R模型的图示法:(1)实体集表示法;（2)属性表法;（3)联系表示法.

层次模型的基本结构是树形结构,网状模型是一个不加任何条件限制的无向图.

一个二维表就是一个关系.

在二维表中凡能唯一标识元组的最小属性称为键

从所有侯选健中选取一个作为用户使用的键称主键.

表A中的某属性是某表B的键，则称该属性集为A的外键

关系中的数据约束：

(1)实体完整性约束:约束关系的主键中属性值不能为空值；

(2)参照完全性约束:是关系之间的基本约束；

(3)用户定义的完整性约束:它反映了具体应用中数据的语义要求.

4.3关系代数

关系数据库系统的特点：关系代数与关系演算.

关系模型的基本运算： (1)插入（2)删除（3)修改（4)查询(包括投影，选择,笛卡尔积运算）

4.4数据库设计与管理

数据库设计是数据应用的核心.

数据库设计的两种方法：

(1)面向数据:以信息需求为主，兼顾处理需求；

(2)面向过程:以处理需求为主,兼顾信息需求.

数据库的生命周期:

需求分析阶段,概念设计阶段,逻辑设计阶段,物理设计阶段，编码阶段,测试阶段,运行阶段，修改阶段.

需求分析常用结构析方法和面向对象的方法.

结构化分析(简称SA)方法用自顶向下，逐层分解的方式分析系统.用数据流图表达关系.

数据字典是各类数据描述的集合，包括5个部分:数据项，数据结构,数据流,数据存储，处理过程.

数据库概念设计方法： (1)集中式模式设计法 (2)视图集成设计法.

视图设计一般有三种设计次序:自顶向下，由底向上，由内向外.

视图集成的几种冲突:命名冲突，概念冲突，域冲突，约束冲突.

数据库的逻辑设计

1从E-R图向关系模式转换。 2逻辑模式规范化及调整、实现 3关系视图设计

关系视图设计:关系视图的设计又称外模式设计.

关系视图的主要作用：

(1)提供数据逻辑独立性； (2)能适应用户对数据的不同需求； (3)有一定数据保密功能.

数据库的物理设计目标是提高数据库访问速度，有效利用存储空间.如索引设计，分区设计.

数据库管理的内容：

(1)数据库的建立； (2)数据库的调整； (3)数据库的重组；

(4)数据库的控制； (5)数据库的故障恢复； (6)数据库监控.