高等学校教材

软件技术基础

徐士良 龚元明 赵鸿德 编著

高等教育出版社

高等学校教材

软件技术基础

徐士良　龚元明　赵鸿德　编著

高等教育出版社

(京)112号

内 容 简 介

本书是为非计算机专业学生学习计算机软件基本知识以及掌握基本软件的使用而编写的。主要内容包括：软件的基础知识，数据结构，操作系统，应用软件设计与开发，数据库技术，计算机网络。每章后面均附有习题。

本书内容新，实例丰富，语言通俗易懂，叙述深入浅出，可作为理工科非计算机专业本科生与研究生的“计算机软件技术基础”课程的教材，也可作为一般工程技术人员学习计算机软件基础知识的参考书，还可以作为使用基本软件的参考手册。

图书在版编目(CIP)数据

软件技术基础/徐士良等编著.-北京：高等教育出版社，1997.5
高等学校教材
ISBN 7-04-005950-9

Ⅰ.软… Ⅱ.徐… Ⅲ.软件-基础理论-高等学校-教材
Ⅳ.TP31

中国版本图书馆CIP数据核字（97）第01759号

*

高等教育出版社出版
北京沙滩后街55号
邮政编码：100009　传真：64014048　电话：64054588
新华书店总店北京发行所发行
高等教育出版社印刷厂印装

开本 787×1092　1/16　印张 24　字数 590 000
1997年7月第1版　1997年7月第1次印刷
印数 0 001-4 094

定价 19.10 元

前　言

随着计算机技术的深入发展，计算机技术的应用已经渗透到各个领域，特别是计算机软件的设计与应用，已经不只是计算机专业人员的事情了。现在，越来越多的软件需要非计算机专业人员来设计与开发；很多系统软件与应用软件由非计算机专业人员来使用，并需要在此基础上进行二次开发。本书就是为适应这种形势的需要而编写的。

本书的特点是强调基础，强调实用，突出一个“新”字。这也是非计算机专业人员学习计算机软件技术的三个主要方面。

全书共分六章，每章后面均有一定数量的习题。

第一章主要介绍了有关计算机处理方面的基本概念和基础知识。包括集合、笛卡尔积、二元关系、算法、计算机软件系统、计算机语言与处理系统，最后还介绍了计算机病毒方面的知识。

第二章介绍了几种常用的数据结构以及基本的查找、排序技术。其中数据结构包括线性结构、树、图。本章介绍的查找与排序技术都是最基本的，也是最常用的。另外，还介绍了 Hash 表技术。

第三章除了简单介绍操作系统的基本概念以及操作系统的主要功能与任务外，还主要介绍了 DOS 6.2、Windows 与 UNIX 操作系统。特别是对于前两种操作系统的使用作了较详细的叙述。

第四章介绍了软件工程的基本概念，特别是详细叙述了结构化的分析与设计方法以及程序的设计、测试与调试方法，最后还简单介绍了软件开发的几种新技术。

第五章介绍了数据库的基本概念以及数据库设计的基本理论，并重点介绍了微机数据库系统 FOXBASE$^+$，对 FOXPRO for Windows 和 Access 也作了简单介绍。

第六章介绍了计算机网络的基本概念，最后还对 Internet 网作了综述性的介绍。

特别要指出的是，本书所介绍的常用基本软件都是新的，并都详细叙述了实际使用操作的方法。

本书的第一章由赵鸿德编写，第二章以及第三章的 3.1、3.2、3.5 节由徐士良编写，第三章的 3.3、3.4 节以及第四、五、六章由龚元明编写。徐士良对全书进行了统稿。

在本书的编写过程中，得到了国家教育委员会工科计算机基础课委会许多同志的关心与支持，在此表示衷心的感谢。

由于作者水平有限，书中难免有错误或不妥之处，恳请读者批评指正。

作　者

1996 年 10 月

目　录

第一章 基础知识

1.1 信息、数据和数据处理

数据是指能够输入计算机并能被计算机处理的数字、字母和符号。

数据的表示形式是多种多样的，可以是数值、图形、图像、声音等。它是用来描述人们所看到的景象和听到的客观情况的某种物理符号的序列。

数据有两种形态。一种是人类可读形式的数据，简称人读数据。数据是由人类首先进行收集、整理、组织和使用的，从而形成了人类特有的语言、文字、数字及图像等。例如，图书、资料、声像制品等，都是特定的人群才能理解的数据。数据的另一种形态为机器可读形式的数据，简称机读数据。日常生活中，所购买的物品上印出的黑白相间粗细不同的条形码，就是一种机读数据。它通过扫描器阅读后，就会把有关物品的信息(如代号、价格等)输给计算机处理。

信息是事物状态及其运动方式的表现形式。信息和数据是两个不可分离又有区别的概念。数据是信息的载体，而信息是对数据的解释，是消化了的数据。信息不随载荷它的物理设备的改变而改变，而数据往往和所用的计算机系统有关。

信息是经过加工整理并对人类社会实践、生产及经营活动产生决策影响的数据。所以，只有通过对数据的去粗取精、去伪存真的加工整理，数据才能发生质的变化而成为信息，给人以新的知识和智慧，从而影响着人类的精神文明和物质文明活动。

数据处理是对各种类型的数据进行收集、存储、分类、加工、检索、优化和传输的过程。通常，人们将数据处理也称为信息处理。

数据处理的数学预备知识有如下几个方面。

1.1.1 集合及其运算

一、集合的概念

所谓集合，是指若干个或无穷多个具有相同属性的元(元素)的集体。

通常，集合名称用大写字母表示，而集合中的元素用小写字母表示。

如果集合 M 由 n 个元素 $a_1,a_2,\cdots,a_n$ 组成($n \geqslant 0$)，则称集合 M 为有限集。例如，大于 1 而小于 100 的所有整数构成的集合 A 为有限集。如果一个集合中有无穷多个元素，则称此集合为无限集。例如，所有整数构成的集合 Z，所有实数构成的集合 R，大于 0 而小于 1 的所有实数构成的集合 B 等均为无限集。不包含任何元素的集合称为空集。例如，大于1而小于2的整数构成的集合为空集。空集通常用 $\varnothing$ 表示。

如果 M 是一个集合，a 是集合 M 中的一个元素，则记作 $a \in M$，称元素 a 属于集合 M；如果 a 不是集合 M 中的元素，则记作 $a \notin M$，称元素 a 不属于集合 M。

对于一个集合，通常用以下两种方法表示。

(1) 列举法

用列举法表示一个集合，是将此集合中的元素全部列出来，或者列出若干项但能根据规律知其所有的元素。例如，上述举出的几个集合的例子可以用如下列举法表示出来：

大于1而小于100的所有整数的集合 A 表示为

$$A=\{2,3,4,\cdots,99\},\qquad \text{有限集}$$

所有整数构成的集合 Z 表示为

$$Z=\{0,\pm 1,\pm 2,\pm 3,\cdots\},\qquad \text{无限集}$$

空集表示为

$$\varnothing=\{\ \},\qquad \text{空集}$$

(2) 性质叙述法

用性质叙述法表示一个集合，是将集合中的元素所具有的属性描述出来。例如：

大于1而小于100的所有整数的集合 A 表示为

$$A=\{a\,|\,1<a<100\text{ 的所有整数}\}$$

所有整数构成的集合 Z 表示为

$$Z=\{z\,|\,z\text{ 为一切整数}\}$$

大于0而小于1的所有实数构成的集合 B 表示为

$$B=\{b\,|\,0<b<1\text{ 的所有实数}\}$$

所有实数构成的集合 R 表示为

$$R=\{r\,|\,r\text{ 为一切实数}\}$$

设 M 与 N 为两个集合。如果集合 M 中的每一个元素也都为集合 N 的元素，则称集合 M 为 N 的子集，记作 $M\subseteq N$ 或 $N\supseteq M$。如果 $M\subseteq N$，且 N 中至少有一个元素 $a\notin M$，则称 M 是 N 的真子集，记作 $M\subset N$ 或 $N\supset M$。如果 $M\subseteq N$ 且 $N\subseteq M$，则称集合 M 和集合 N 相等，记作 $M=N$。

二、集合的基本运算

(1) 两个集合的并 (union)

设有两个集合 M 和 N，它们的并集记作 $M\cup N$，其定义如下：

$$M\cup N=\{a\,|\,a\in M\text{ 或 }a\in N\}$$

两个集合 M 与 N 的并集是指 M 与 N 中所有元素组成的集合。

(2) 两个集合的交 (intersection)

设有两个集合 M 和 N，它们的交集记作 $M\cap N$，其定义如下：

$$M\cap N=\{a\,|\,a\in M\text{ 且 }a\in N\}$$

两个集合 M 与 N 的交集是指 M 与 N 中所有共同元素组成的集合。

两个集合 M 与 N 的并、交均满足交换律，即

$$M\cup N=N\cup M,\qquad M\cap N=N\cap M$$

(3) 两个集合的差 (difference)

设有两个集合 M 和 N，M 和 N 的差集记作 $M-N$，其定义如下：

$$M-N=\{a\,|\,a\in M\text{ 但 }a\notin N\}$$

两个集合的差不满足交换律，即

$$M-N\neq N-M$$

例1 设集合

$$A=\{a,b,c,d,e\}$$
$$B=\{d,e,f,g,h\}$$

则

$$A\cup B=\{a,b,c,d,e,f,g,h\}$$
$$A\cap B=\{d,e\}$$
$$A-B=\{a,b,c\}$$
$$B-A=\{f,g,h\}$$

对于集合的并、交、差有以下几个基本性质：

(1) 结合律

$$(A\cap B)\cap C=A\cap(B\cap C)$$
$$(A\cup B)\cup C=A\cup(B\cup C)$$

(2) 分配律

$$A\cap(B\cup C)=(A\cap B)\cup(A\cap C)$$
$$A\cup(B\cap C)=(A\cup B)\cap(A\cup C)$$

(3)　$$(A-B)\cup(B-A)=(A\cup B)-(A\cap B)$$

(4)　$$B\cap(A-B)=\varnothing$$
$$(A\cap B)\cup(A-B)=A$$

三、映射

定义 1.1　设 A、B 是两个非空集，如果根据一定的法则 f，对于每一个 $x\in A$，在 B 中都有唯一确定的元素 y 与之对应，则称 f 是定义在 A 上而在 B 中取值的映射，记作 $f:A\to B$。并将 x 与 y 的关系记作 $y=f(x)$。x 称为自变元，y 称为在 f 作用下 x 的像。

集合 A 称为 f 的定义域，$f(A)=\{f(x)|x\in A\}$ 称为 f 的值域。

定义 1.2　设给定映射 $f:A\to B$，且 $B=f(A)$（即 f 的像充满整个 B）。如果对于每个 $y\in B$，仅有唯一的 $x\in A$ 使 $f(x)=y$，则称 f 有逆映射 f^{-1}（它是定义在 $f(A)$ 上而取值于 A 的映射）。当映射 $f:A\to f(A)$ 有逆映射时，则称 f 是一一映射。

定义 1.3　若 A、B 两集合有一一映射 f 存在，使 $f(A)=B$，则称 A 与 B 为一一对应。

如果集合 A 与 B 为一一对应，则称它们互相对等，并记作 $A\sim B$。当两个集合互相对等时，称它们有相等的浓度（或元素个数）。

例 2　设两集合为

$$A=\{1,2,\cdots,10\}=\{x|1\leqslant x\leqslant 10\text{ 的整数}\}$$
$$B=\{1,2,\cdots,20\}=\{y|1\leqslant y\leqslant 20\text{ 的整数}\}$$

若映射 $f:A\to B$ 为

$$y=2x$$

其中定义域为 A，值域为

$$f(A)=\{2,4,6,\cdots,20\}$$

显然，映射 f 不是一一映射。

例 3　设两集合为

$$A=\{x|0\leqslant x\leqslant 4\text{ 的所有实数}\}$$
$$B=\{y|0\leqslant y\leqslant 2\text{ 的所有实数}\}$$

考虑映射 $f:A \to B$ 为

$$y = \sqrt{x}$$

其中定义域为 A，值域为 $f(A) = B$。

显然，f 为一一映射，集合 A 与 B 互相对等，即 $A \sim B$。

集合的对等满足以下性质：

(1) 自反性，即 $A \sim A$；

(2) 对称性，即若 $A \sim B$，则 $B \sim A$；

(3) 传递性，即若 $A \sim B$ 且 $B \sim C$，则 $A \sim C$。

1.1.2　自然数集与数学归纳法

由所有自然数所组成的集合 $\{1,2,3,\cdots\}$ 称为自然数集。自然数集是一个无限集。

由自然数组成的集合均是自然数集的子集。自然数集的子集可以是有限集，也可以是无限集。

与自然数集对等(即具有相等浓度)的集合称为可列集(或可数集)。任一可列集中的元素排列时可标以正整数下标，即任意可列集 M 均可写成

$$M = \{a_1, a_2, \cdots, a_n, \cdots\}$$

关于自然数集及其子集有以下两个命题成立。

定理 1.1　在自然数集的任一非空子集 M 中，必定有一个最小数。即在集合 M 中有不大于其它任意数的数。

证明　因为 M 非空，所以在 M 中可取得一自然数 n。

显然，M 中所有不大于 n 的自然数形成的非空集 N 包含在 M 中，即 $N \subset M$。

如果 N 中有最小数，则此最小数就是 M 的最小数。

而在 N 中最多有 n 个自然数(1 到 n)，因此 N 中有一个最小数。

综上所述，在自然数集的任一非空子集 M 中必定有一个最小数。定理得证。

定理 1.2　假定 M 是由自然数形成的集合，如果它含有 $1,2,\cdots,k$，并且当它含有数 $n-1$，$n-2,\cdots,n-k(n>k)$ 时，也含有数 n，那么它含有所有的自然数，即 M 是自然数集。

证明　设 N 是所有不属于 M 的自然数形成的集合，则 $1,2,\cdots,k \notin N$。

现假设 N 不是空集，则由定理 1.1 可知：在 N 中必定有一个最小数。设此最小数为 c。

由于 c 是 N 中的最小数，即 $c \in N$，因此 $c \notin M$，且 $c \neq 1,2,\cdots,k$，同时，$c-1, c-2,\cdots$，$c-k$ 均为自然数。又由于 c 是 N 中的最小数，所以自然数 $c-1, c-2,\cdots,c-k \notin N$，即 $c-1, c-2,\cdots,c-k \in M$，而根据定理中的条件有 $c \in M$。

由上所述，一方面有 $c \notin M$，另一方面 $c \in M$，这就导致矛盾。这个矛盾是由于一开始假定 N 不是空集所造成的。因此 N 只能为空集，即所有自然数均在 M 中，M 为自然数集。定理得证。

上述定理是数学归纳法的基础。通常，为了证明一个命题对于所有的自然数是真，采用数学归纳法证明的步骤如下：

(1) 证明命题对于自然数 $1,2,\cdots,k$ 是真的；

(2) 假设命题对于自然数 $n-k, n-k+1,\cdots,n-1(n>k)$ 是真的(这一步称为归纳假设)；

(3) 证明命题对于自然数 n 也是真的。

上述步骤(1) 中 k 值的选取决定于在步骤(3) 的证明过程中要用到归纳假设的最小自然数。在步骤(3) 中，为了证明命题对于自然数 n 是真的，要用到归纳假设的最小自然数，如果它为 $n-k$，则在步骤(1) 中要对 1 到 k 中的所有自然数证明命题为真。

例 4 证明任意一笔大于 7 元的整数付款均可用 3 元及 5 元的票面的钞票支付。

这个问题相当于要证明下列命题：

对于任意的自然数 n，存在一对非负整数 (i,j) 有

$$7+n=3i+5j$$

下面用数学归纳法证明这个命题。

证明

(1) 当 $n=1$ 时，有 $7+1=3+5$，即 $i=1,j=1$，命题成立。

当 $n=2$ 时，有 $7+2=3\times 3+5\times 0$，即 $i=3,j=0$，命题成立。

当 $n=3$ 时，有 $7+3=3\times 0+5\times 2$，即 $i=0,j=2$，命题成立。

(2) 假设命题对于自然数 $n-1,n-2,n-3(n>3)$ 成立(归纳假设)。

(3) 考虑自然数 n，有

$$7+n=[7+(n-3)]+3$$

根据归纳假设，对于自然数 $n-3$ 命题成立，设存在一对非负整数 i_1,j_1 有

$$7+(n-3)=3i_1+5j_1$$

则有

$$7+n=[7+(n-3)]+3=3(i_1+1)+5j_1=3i+5j$$

其中 $i=i_1+1,j=j_1$ 均为非负整数。即对于自然数 n 命题也成立。

由此得出结论，对于所有的自然数 n 命题成立。

在这个例子中，由于步骤(3) 的证明过程中要用到归纳假设的最小自然数为 $n-3$，因此在步骤(1) 中取 $k=3$。

1.1.3 笛卡尔积

在 1.1.1 节中介绍了两个集合的并、交、差运算。对于集合，还有一种很重要的运算，即笛卡尔积(Cartesian product)。

设有 n 个集合 $D_1,D_2,\cdots,D_n$，此 n 个集合的笛卡尔积定义为

$$D_1\times D_2\times\cdots\times D_n=\{(d_1,d_2,\cdots,d_n)\mid d_i\in D_i,i=1,2,\cdots,n\}$$

其中 $(d_1,d_2,\cdots,d_n)$ 称为 n 元组(n-tuple)，d_i 称为 n 元组的第 i 个分量。

由笛卡尔积的定义可以看出，n 个集合的笛卡尔积是以 n 元组为元素的集合，而每一个 n 元组中的第 i 个分量取自于第 i 个集合 D_i。

例 5 设有三个集合

$$A=\{a_1,a_2,a_3\},\quad B=\{b_1,b_2\},\quad C=\{c_1,c_2\}$$

则它们的笛卡尔积为

$$\begin{aligned}A\times B\times C=\{&(a_1,b_1,c_1),(a_1,b_1,c_2),(a_1,b_2,c_1),(a_1,b_2,c_2),\\&(a_2,b_1,c_1),(a_2,b_1,c_2),(a_2,b_2,c_1),(a_2,b_2,c_2),\\&(a_3,b_1,c_1),(a_3,b_1,c_2),(a_3,b_2,c_1),(a_3,b_2,c_2)\}\end{aligned}$$

如果 n 个集合 $D_1,D_2,\cdots,D_n$ 中的元素个数分别为 $m_1,m_2,\cdots,m_n$，则其笛卡尔积中共有 m_1

$\times m_2 \times \cdots \times m_n$ 个 n 元组。即 n 个集合的笛卡尔积是所有 n 元组组成的集合。

1.1.4 二元关系

定义 1.4 设 M 和 N 是两个集合，则其笛卡尔积

$$M \times N = \{(x,y) | x \in M \text{且} y \in N\}$$

的每一个子集称为在 $M \times N$ 上的一个二元关系。

如果 $M = N$，则其笛卡尔积

$$M \times N = \{(x,y) | x,y \in M\}$$

的每一个子集称为在集合 M 上的一个二元关系，简称为集合 M 上的一个关系。

例 6 设集合 M 为

$$M = \{a,b,c,d,e,f\}$$

则下列各二元组的集合为在集合 M 上的一个关系：

$$R_1 = \{(a,b),(b,c),(c,d),(d,e),(e,f)\}$$

$$R_2 = \{(a,e),(a,a),(c,f),(d,b),(e,a),(f,c),(b,d)\}$$

$$R_3 = \{(a,a),(b,b),(c,c),(d,d),(e,e),(f,f),(c,f),(e,a)\}$$

$$R_4 = \{(a,b),(b,e),(c,d),(d,f),(a,e),(c,f)\}$$

集合 M 上的一个关系实际上反映了 M 中各元素之间的联系。

定义 1.5 设 R 是集合 M 上的一个关系。

(1) 如果 $(a,b) \in R$，则称 a 是 b 的关于 R 的前件 (predecessor)，b 是 a 的关于 R 的后件 (successor)。

(2) 如果对于每一个 $a \in M$，都有 $(a,a) \in R$，则称关系 R 是自反的 (reflexive)；如果对于任何 $a \in M$，$(a,a) \in R$ 均不成立，则称关系 R 是非自反的 (antireflexive)。

(3) 如果 $(a,b) \in R$ 时必有 $(b,a) \in R$，则称关系 R 是对称的 (symmetric)。

(4) 如果当 $(a,b) \in R$ 且 $(b,c) \in R$ 时必有 $(a,c) \in R$，则称关系 R 是传递的 (transitive)。

在例 6 中，关系 R_1 是非自反的，但不是对称的，也不是传递的；关系 R_2 是对称的，但不是自反的，也不是非自反的，也不是传递的；关系 R_3 是自反的，但不是对称的，也不是传递的；关系 R_4 是传递的，且是非自反的，但不是对称的。

由此可以看出，集合 M 中的各元素之间的逻辑关系可以由集合 M 上的一个关系来描述。

定义 1.6 设 R 是 M 上的一个传递关系，且 $T \subseteq R$。若对于任何 $(x,y) \in R$，在 M 中有元素 $x_0,x_1,x_2,\cdots,x_n (n \geqslant 1)$ 满足：(1) $x_0 = x$，(2) $x_n = y$，(3) $(x_{i-1},x_i) \in T \quad (i = 1,2,\cdots,n)$。则称关系 T 是关系 R 的基 (basis)，又称关系 R 是关系 T 的传递体 (transitive hull)。

例 7 设集合 M 为

$$M = \{1,2,3,4,5\}$$

集合 M 上的一个关系为

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

可以验证，关系 R 是非自反的，且是传递的。现考虑集合 M 上的另一个关系

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

由于 $T \subseteq R$，且对于关系 R 中的每一个二元组 (x,y)，在 M 中存在元素 $x_0,x_1,\cdots,x_n$，满足定义 1.6 中的三个条件。验证过程如下：

R	$x_0,x_1,\cdots,x_n$	
(1,2)	1,2	$(1,2)\in T$
(2,3)	2,3	$(2,3)\in T$
(3,5)	3,5	$(3,5)\in T$
(3,4)	3,4	$(3,4)\in T$
(1,3)	1,2,3	$(1,2),(2,3)\in T$
(1,4)	1,2,3,4	$(1,2),(2,3),(3,4)\in T$
(2,4)	2,3,4	$(2,3),(3,4)\in T$
(2,5)	2,3,5	$(2,3),(3,5)\in T$
(1,5)	1,2,3,5	$(1,2),(2,3),(3,5)\in T$

因此,T 是 R 的具有 4 个元素(即二元组)的基。

同样还可以验证关系

$$T_1=\{(1,2),(1,4),(2,3),(3,5),(3,4)\}$$

是 R 的具有 5 个元素的基。但关系

$$T_2=\{(1,2),(2,3),(3,4),(4,5)\}$$

不是 R 的基,因为$(4,5)\notin R$。

1.2 算 法

1.2.1 算法的概念

算法是指为在有限步内解决一个具体问题而规定的意义明确的解题步骤的有限集合。概括地说,算法是指解题方案的准确而完整的描述。从程序角度,也可以说算法是一个有限条指令的集合,这些指令确定了解决某一特定类型问题的运算序列。

一、算法的特征

作为一个算法,应具有以下四个特征。

(1) 可行性

算法的可行性包括两个方面:一是算法中的每一个步骤必须是能实现的,例如,在算法中,不允许出现分母为零的情况,在实数范围内不能求一个负数的平方根等;二是算法执行的结果要能达到预期的目的。

针对实际问题设计的算法,人们总是希望得到满意的结果。算法总是与特定的计算工具有关。例如,某计算工具具有七位有效数字(如 FORTRAN 中的单精度运算),在计算下列三个量

$$A=10^{12},\quad B=1,\quad C=-10^{12}$$

的和时,如果采用不同的运算顺序,就会得到不同的结果,即

$$A+B+C=10^{12}+1+(-10^{12})=0$$

$$A+C+B=10^{12}+(-10^{12})+1=1$$

而在数学上,$A+B+C$ 与 $A+C+B$ 是完全等价的。因此,算法与计算公式是有差别的。在设计一个算法时,必须考虑它的可行性,否则是不会得到满意结果的。

(2) 确定性

算法的确定性是指算法中的每一个步骤都必须有明确定义，不允许有模棱两可的解释，也不允许有多义性。这一性质也反映了算法与数学公式的明显差异。在解决实际问题时，可能会出现这样的情况：针对某种特殊问题，数学公式是正确的，按此数学公式设计的算法从数学角度来说也是明确的，但是算法的程序实现其计算过程可能会使计算机无所适从。这是因为计算过程只考虑了正常使用的情况，而当出现异常情况时，此计算过程就不能适应了。例如，某计算工具规定：大于 100 的数认为是比 1 大很多，而小于 10 的数不能认为是比 1 大很多；且在正常情况下出现的数或是大于100，或是小于10。但指令“输入一个 X，若 X 比1大很多，则输出数字1，否则输出数字 0”是不确定的。这是因为，在正常的输入情况下，这一计算可以得到正确的结果，但在异常情况下(输入的 X 在 10 与 100 之间)，其输出结果就不确定了。

(3) 有穷性

算法的有穷性是指算法必须能在有限的时间内做完，即算法必须能在执行有限个步骤之后终止。数学中的无穷级数，在实际计算时只能取有限项，即计算无穷级数数值的过程只能是有穷的。因此，一个数的无穷级数表示只是一个计算公式，而根据精度要求确定的计算过程才是有穷的算法。

算法的有穷性还应包括合理的执行时间的含义。因为如果一个算法需要执行千万年，显然失去了实用价值。例如，克莱姆(Cramer)规则是求解线性代数方程组的一种数学方法，但不能以此设计为算法，这是因为，虽然总可以根据克莱姆规则设计出一个计算过程用于计算所有可能出现的行列式，但此计算过程所需的时间实际上是不能容忍的。

(4) 有足够的原始数据

一个算法是否有效，还取决于为算法所提供的数据是否足够。例如，对于指令：“如果小明是学生，则输出字母 Y，否则输出字母 N”。当算法执行过程中提供了小明不是学生的情报时，执行的结果将输出字母 N；当提供的情报中只有部分学生的名单，且小明恰在其中，执行的结果将输出字母 Y。但如果在提供的部分学生的名单中，找不到小明的名字，则在执行算法过程中无法服从此指令，因为在部分学生的名单中找不到小明的名字，既不能说明小明是学生，也不能说明小明不是学生。

通常，算法中的各种运算总是要施加到各个运算对象上，而这些运算对象又可能具有某种初始状态，这是算法执行的起点或依据。因此，一个算法执行的结果总是与输入的初始数据有关，不同的输入将会有不同的结果输出，当输入不够或输入错误时，算法本身就无法执行或执行有错。一般来说，当算法拥有足够的原始数据时，此算法才是有效的，而提供的数据不够时，算法并不是有效的。

综上所述，所谓算法是一个过程，这一过程有一组严谨地定义运算顺序的规则，并且每一个规则都是有效的且是明确的，此顺序将在有限的次数下终止。

二、算法的要素

算法由操作和控制结构两要素组成。

(1) 算法的操作

每个算法实际上是按解题要求从所用解题环境能进行的操作中选择合适的操作所组成的指令序列。计算机算法就是计算机能执行的操作所组成的指令序列。

通常，计算机可以执行的基本操作是以指令的形式描述的。一台计算机能执行的全部指令

的集合，称为该机器的指令系统。程序就是按解题要求从计算机的指令系统中选择合适的指令所组成的指令序列。

计算机中基本的运算和操作有：

① 算术运算：加、减、乘、除；

② 逻辑运算："与"、"或"、"非"；

③ 关系运算：大于、小于、等于、不等于；

④ 数据传输：赋值、输入、输出。

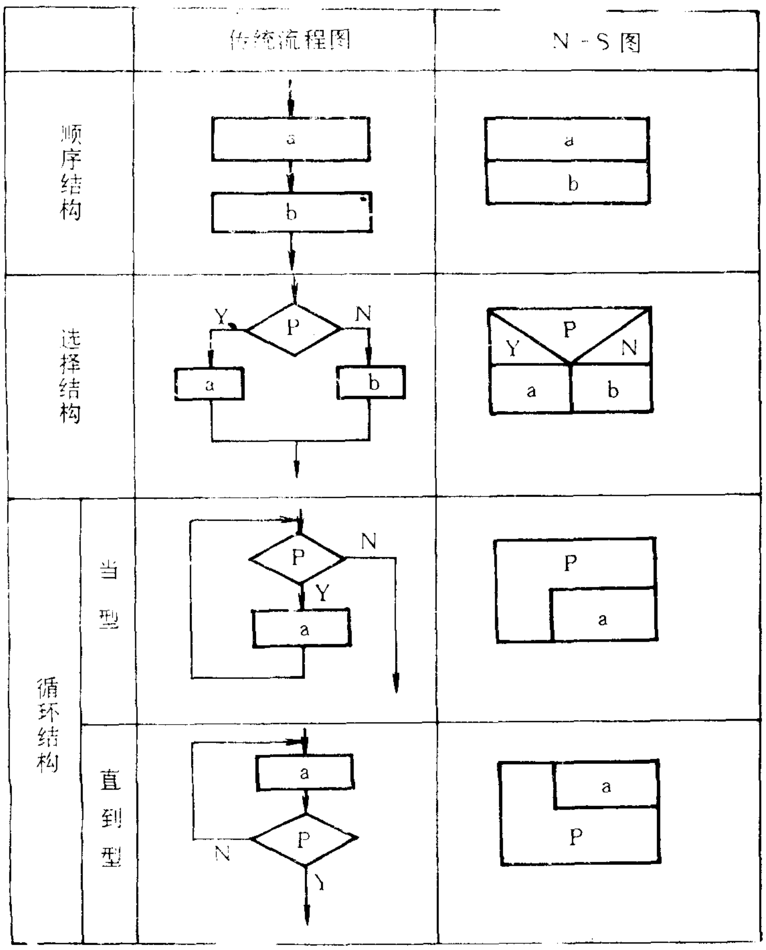

图 1.1　算法控制结构流程图

计算机算法设计应从上述四种基本功能操作考虑，按解题要求从这些基本操作中选择合适的操作组成解题的操作序列。其特征着重于动态的执行，这和传统的着重于静态描述的或按演绎方式求解问题的过程是有区别的。演绎数学以公理系统为基础，问题的求解过程是通过有限次推演完成的，每次推演都将对问题作进一步的描述。如此不断推演，直到直接将解描述出来。而算法则是使用一些最基本的操作，通过对已知条件进行一步一步的加工、变换，从而实现解题目标。这两种方法的解题思路是不同的。所以，初学程序设计的人一定要把自己的解题思路从演绎转到算法的思路上来。

(2) 算法的控制结构

一个算法的功能不仅取决于所选用的操作，而且还决定于操作之间的执行顺序，即算法的控制结构。算法的控制结构给出了算法的框架，决定了各操作的执行顺序。用流程图可以形象地表示出算法的控制结构。常用的流程图有两种，一是传统流程图，一是 N-S 结构化流程图（见图 1.1）。在图 1.1 中，a，b 为程序块，P 为条件。

1966 年，Bohm 和 Jacopini 证明了任何复杂的算法都可以用顺序、选择、循环三种基本控制结构组合而成。

1.2.2　算法设计的步骤

本节简单介绍算法设计的完整过程，以便建立一个较好的算法，求得问题的解。

1. 问题的描述

准确而完整地描述问题是解决问题的第一步。要做到这一点，必须注意以下一些问题：在未经加工的原始表达中所用的行话是否大家都明白？有二义性吗？用户提供了哪些信息？这些信息有什么用？用户要求得到什么结果？是否遗漏了什么信息？遗漏的信息有何用途？如何判定求得的结果是所需要的？描述中做了哪些假定？等等。必须认真审查问题的有关描述，深入调查，以加深对问题的了解。

例如货郎担问题：设货郎在一天内要到 n 个村庄去推销货物，已知从一个村庄到其它各村庄的费用，求总费用最小的路线。给出的信息有 n 个村庄的关系图和相应的费用矩阵 $C=(c_{ij})_{n\times n}$。费用矩阵的元素 c_{ij} 表示从村庄 i 到村庄 j 的费用。显然，费用矩阵是一个 n 阶的方阵。

为了求得来往于 n 个村庄之间的最小总费用，有一些附加问题尚须进一步了解清楚。例如一天之内能否两次经过某个村庄？除基地村庄外，其它 $n-1$ 个村庄中是否有必须优先访问的村庄？等等。如果说一天之内不能两次访问某个村庄，各村庄之间没有优先权。这样，问题的描述可以归结为求一张村庄表。表的起点和终点为基地村庄，其它村庄在表中必须且只能出现一次。表中村庄的次序即为货郎访问的次序，在表中相邻村庄之间标上费用，它们之和即为总费用。显然，向用户提供的是一张总费用最小的村庄表。

2. 模型的建立

用计算机解决实际问题必须有合适的数学模型，不然，在大型、复杂的问题面前，计算机是无能为力的。对一个实际问题建立数学模型，可以考虑这样两个基本问题：最适合于此问题的数学结构是什么？是否有已经解决了的类似问题可借鉴？

如果上述第二个问题的答复是肯定的，那么通过类似问题的分析、比较和联想，可加速问题的解决。但上述第一个问题毕竟更重要些。如何选择恰当的数学工具来表达已知的和要求的量，受多种因素影响：设计人员的数学知识水平；数据结构是否表达方便；计算是否简单；所要进行的操作种类的多少与功能的强弱等。显然，对同一问题可以用不同的数学工具建立不同的模型，因此要对不同的模型进行分析、比较，从中选出最优、最有效的模型。然后，根据选定的数学模型，对问题进行重新描述。此时，应考虑下列一些问题：

模型能否清楚地表达与问题有关的所有重要信息？模型中是否存在与所期望的结果相关的数学量？能否正确反映输入、输出的关系？用计算机处理该模型是否有困难？如能取得满意的回答，那么该数学模型可作为候选模型。

例如，根据图论的知识，可以把 n 个村庄画成 n 个顶点（结点），分别标上编号 $1,2,\cdots,n$。任意两村庄 i,j 之间用线连起来，标上费用。这样建立起来的图（或网络）就是解决货郎担问题的