

BASIC BASICBA

汉字BASIC程序

结构化设计教程

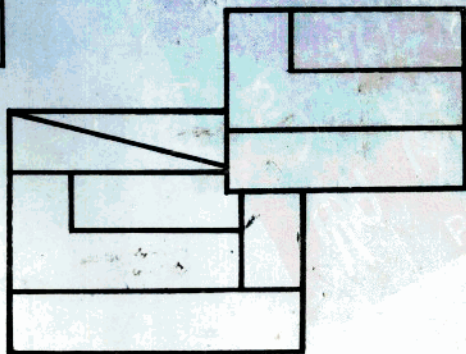
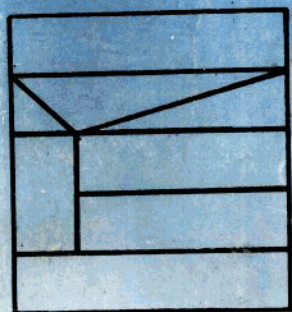
主 编：莫忠学

副主编：雷庆礼 张修如

萧增良 夏 军

李仗国

✍



海南国际新闻出版中心

海南出版社

内 容 提 要

本书是一本 BASIC 语言的新教材。书中采用了新的结构化理论和流程图,内容新颖,图文并茂。

全书共分 12 章,其中包括基础知识、汉字输入、结构化理论、编程技巧、函数、子程序、作图及音乐、上机实验等。突出了算法,例题丰富。

本书可作为大专及中专的教材,亦可作为各类成人学习班及个人自学之用。

前 言

当前,我们所处的时代,是一个科学技术飞速发展的时代,科学要求我们不断改进,不断革新,不断创造。

近年来,计算机科学的发展尤其迅速。作为广泛推广和普及的 BASIC 语言,也应该有所发展。多年来,国内通用的 BASIC 语言,一直沿用传统的理论和方式,这对于推广 BASIC 语言的学习、促进计算机语言的发展,起到了一定的作用。但是,由于微型计算机的大量使用,促进了 BASIC 语言的广泛普及;此外,又由于社会经济的迅速发展,对 BASIC 语言的要求大大提高。因此,对 BASIC 进行研究、改造和提高的工作,势在必行。现代计算机软件系统的社会化大生产,要求既快又好,因而软件的开发,必须高度自动化、标准化。无论对设计程序或是描绘程序流程图,都是如此。

教学工作必须满足社会的需求。因而促使我们编写了这本教材。本书主要讲述高级 BASIC,其中,引入了结构化的新理论和流程图,解决了 BASIC 结构性差的弱点。书中的一些主要程序,已在 IBM PC 机上通过。

考虑到概念、内容的衔接性,本书尽可能采用习惯的术语。为适应不同层次、不同类型读者的使用,文字力求通俗易懂。对于不同的学习对象,可以选讲、选学。为方便国内读者的实际应用,我们增加了使用汉字的内容,这就结束了 BASIC 语言不用汉字的状态。

全书共分十二章,莫忠学主编。副主编由雷庆礼、张修如、肖增良、夏军和李仗国担任,并由莫忠学统编全书。此外,谷映红参加了本书的整理工作。

本书在出版过程中,得到了中南林学院、长沙交通学院等单位的领导和同志们的大力支持和帮助,我们表示深切的谢意。

由于我们的水平有限,加之出版仓促,缺点和错误在所难免,敬请读者批评指正。

编者

1993 年 10 月

目 录

第一章 计算机的基础知识

§1. 概述	(1)
§2. 二进制	(3)
§3. 语言及解题步骤	(8)
§4. 常量和变量	(9)
§5. 表达式	(11)
§6. 内部函数	(14)
§7. BASIC 程序编制的基本规则	(16)
习题	(17)

第二章 汉字的输入

§1. 区位码输入法	(19)
§2. 拼音码输入法	(21)
§3. 五笔划输入法	(22)
§4. 五笔字型输入法	(24)
习题	(34)

第三章 结构化理论基础

§1. 基本原理	(32)
§2. 第一结构	(33)
§3. 第二结构	(34)
§4. 第三结构	(34)
§5. 算法的概念	(35)
§6. 结构化程序设计的方法和风格	(36)
习题	(36)

第四章 第一结构的程序设计

§1. 注释和暂停语句	(38)
§2. 赋值语句	(39)
§3. 键盘输入语句	(41)
§4. 读数和置数语句	(41)
§5. 恢复数据区语句	(43)
§6. 输出语句	(44)

§7. 第一结构算法的应用	(49)
习题	(54)
第五章 第二结构的程序设计	
§1. GOTO 语句	(57)
§2. 选择语句	(58)
§3. 带 ELSE 的选择语句	(62)
§4. 开关选择语句	(66)
§5. 第二结构算法的应用	(67)
习题	(75)
第六章 第三结构的程序设计	
§1. 用 IF、GOTO 语句构成的循环	(77)
§2. 当型循环语句	(80)
§3. 步长型循环语句	(82)
§4. 循环嵌套	(84)
§5. 循环的出口	(87)
§6. 算法的应用	(89)
习题	(100)
第七章 数组	
§1. 下标变量和数组	(101)
§2. 数组说明语句	(102)
§3. 有关数组的其它语句	(103)
§4. 数组的排序	(104)
§5. 查询与搜索	(108)
§6. 数组在算法中的应用	(109)
§7. 算法举例	()
习题	(124)
第八章 函数与子程序	
§1. 自定义函数	(125)
§2. 子程序	(127)
§3. 算法实例	(131)
第九章 作图与音响	
§1. 屏幕控制	(143)
§2. 绘图语句	(146)

§3. 音响	(150)
习题	(153)
第十章 常用编程技术	
§1. 可用于菜单的语句和函数	(154)
§2. 菜单技术	(155)
§3. 使用光标选择菜单	(161)
§4. 陷阱技术	(164)
§5. 打印制表技术	(166)
§6. 程序链接技术	(169)
§7. 动画技术	(175)
习题	(178)
第十一章 文件	
§1. 文件的使用	(179)
§2. 文件函数	(183)
§3. 顺序文件的读写	(185)
§4. 随机文件的读写	(188)
§5. 文件的编制	(190)
§6. 文件的应用	(195)
第十二章 上机实验	
§1. 实验一 微型计算机的认识	(200)
§2. 实验二 汉字的录入	(203)
§3. 实验三 第一结构程序设计	(205)
§4. 实验四 第二结构程序设计	(208)
§5. 实验五 第三结构程序设计	(210)
§6. 实验六 数组、函数和子程序	(211)
§7. 实验七 屏幕控制与作图	(214)
§8. 实验八 菜单技术	(216)
附录一 ASCII 码表	(219)
附录二 BASIC 错误信息	(221)

第一章 计算机的基础知识

§1. 概 述

一 计算机的发展简史

现代新兴科学技术的发展,都是与电子计算机科学分不开的。两者的渗透、结合,便出现了一系列的新兴边缘学科。从而进一步导致了更多的科学和技术的重大突破。

然而,电子计算机本身也是从初级到高级发展起来的。一般按照组成电子计算机的物理器件来看,可以把它顺次分为五代。

1. 电子管计算机

世界上第一台电子计算机是1946年美国宾文法尼亚大学创制的ENIAC。(Electronic Numerical Integrator and calculator,译为“电子数值积分器与计数器”)。这一代机器的逻辑元件主要是电子管,共18800个,还用了1500个继电器,体积为3000立方尺,运算速度为5000次/秒。使用机器语言,主要用于科学计算。

2. 晶体管计算机

1957年,美国飞歌公司研制了世界上第一台晶体管电子计算机TRANSTCS-1000,运算速度达到几万~几十万次/秒,它的逻辑元件是采用晶体管分离元件,体积减小,软件开始采用高级语言,如FORTRAN等,开始用于过程控制。

3. 集成电路计算机

到了60年代,这一代机器采用中、小规模集成电路,运算速度提高到几十万~几百万次/秒,软件已能使用操作系统,功能更加提高。

4. 大规模集成电路计算机

它的逻辑器件主要采用大规模和超大规模集成电路,运算速度达几十万~几十亿次/秒,体积大大缩小,软件、硬件能互相结合。

此外,集成电路派生了单板机、单片机及巨型机。

上述四代电子计算机的方案都是分成运算器、控制器、存储器和输入/输出设备共五个基本组成部分。其中,运算器与控制器合称为中央处理器,即Central Processing Unit,简称CPU。再配以软件,使得计算机成为全自动信息处理机。这一原理是冯·诺依曼(Von Neumann)在1946年最先提出来的,因此,前四代又称为冯·诺依曼计算机。而下述第五代则称为非冯·诺依曼计算机。

5. 第五代计算机

它是科学家们正在研制的全新计算机。具有“推理”和“人功智能”的功能,它包含超大规模集成电路、生物、光学、超导器件和仿生、智能思维技术。最近制成了第一台超导计算机及第一台数字光学处理器就是一个证明。

计算机科学的发展,已使个人机向网络发展。计算机网络是计算机科学技术和数据通讯

技术两者结合的产物,它实现资源共享、用于办公、管理、气象预报、情报检索及学术交流等许多方面。

我国的计算机科学起步较早,1958年便研制出第一台电子管计算机,1964年后,又研制出晶体管计算机,如X-2、DJS-21等;1971年研制了集成电路计算机TQ-16;1973年研制了大型集成电路计算机DJS-11;1983年研制了“银河”巨型机;1992年研制了银河-II型10亿次机和世界先进水平的仿真机。由此可见,我国的科学技术,在世界上已经步入了先进行列。

二 电子计算机的组成:

电子计算机系统的组成如图1-1所示。

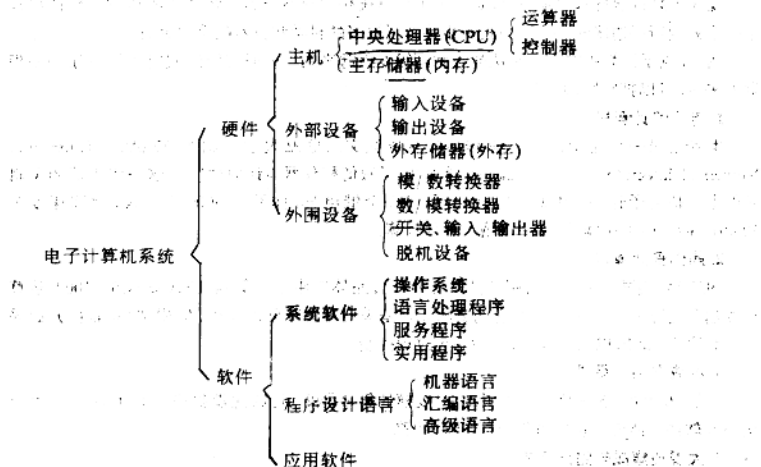


图1-1

1. 硬件

组成计算机的各种物理部件,人们称它们为硬设备,或叫做计算机的硬件。

微型计算机的硬件系统的组成如图1-2所示

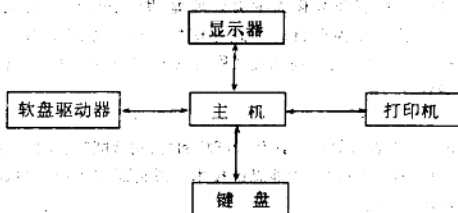


图1-2

主机包括运控器、内存储器。大多数微型计算机的磁盘驱动器(硬盘和软盘)是装在主机箱内。外部设备是显示器、打印机和键盘。

2. 软件

软件是计算机各类程序的统称,又叫做程序系统。它的作用在于发挥机器的效能,其功能是管理和更好地使用机器。

§2. 二进制

人们习惯于用十进制,逢十进一。这完全是由于人们的习惯,而并非天经地义的。事实上,人们还用了其它一些进制,如六十进制(一分钟等于六十秒,一度等于六十分),十六进制(一市斤等于十六老两),十二进制(一打等于十二个,一英尺等于十二英寸,一年等于十二月)等。人们生活中也有用二进制的,如鞋、袜、手套、筷子等,都是逢二进一。可见,用什么进制完全取决于人们的需要。

一 为什么要用二进制

电子数字计算机内部都是用二进制数,这是由于二进制数在电气元件中容易实现,容易运算。二进制中只有两个数,即0和1,在电学中具有两种稳定状态以代表0和1的东西是很多的,如:电压的高和低,电灯的亮和灭,电容器的充电和放电,脉冲的有和无,晶体管的导通和截止……等。而要找出一种具有十个稳定状态的电气元件是很困难的。

二进制数的运算公式很简单:

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=10$$

$$0 \times 0=0$$

$$0 \times 1=0$$

$$1 \times 0=0$$

$$1 \times 1=1$$

即加法四条,乘法四条(各有 $2^2=4$ 条)。而十进制的运算公式从 $0+0=0$ 到 $9+9=18$ 共有加法规则100条,从 $0 \times 0=0$ 到 $9 \times 9=81$ 乘法规则也是100条($10^2=100$ 条)。显然,计算机进行二进制数的运算比十进制数简单得多。

二 十进制和二进制间的转换(整数)

由于人们习惯于十进制,因此常常要进行十进制数的转换工作。只要记住,二进制的最基本的规定是逢二进一。一个十进制整数要化为二进制整数只需将它一次又一次地被2除,得到的余数(从最后一次的余数读起)就是用二进制表示的数,叫做“除2取余法”。

如

$$\begin{array}{r} 2 \overline{) 11} \quad (1) \\ 2 \overline{) 5} \quad (1) \\ 2 \overline{) 2} \quad (0) \\ 2 \overline{) 2} \quad (1) \\ 0 \end{array}$$

得到: $(11)_{10} = (1011)_2$

在上面一行中,括弧外的注脚10或2分别表示括弧中的数是十进制数或二进制数。

换句话说,把十进制数化为以2为底的指数形式,其系数的顺序排列(由高次到低次)就是以二进制表示的数。

由0到9的十进制数转换成二进制数见表1-1。

表 1-1

十进制数	化为以2为底的指数形式	二进制数
0	0×2^0	0
1	1×2^0	1
2	$1 \times 2^1 + 0 \times 2^0$	10
3	$1 \times 2^1 + 0 \times 2^1 + 0 \times 2^0$	11
4	$1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	100
5	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	101
6	$1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$	110
7	$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$	111
8	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	1000
9	$1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	1001

如果一个十进制数F可表示为

$$(F)_{10} = a_0 \cdot 2^0 + a_1 \cdot 2^1 + \dots + a_{n-1} \cdot 2^{n-1} + a_n \cdot 2^n$$

则 $a_0, a_1, a_2, a_3, \dots, a_n$ 就是F在二进制中的表示形式。反之,二进制可化为十进制数,如表1-2。

表 1-2

二进制数	十进制数
1	$2^0 = 1$
10	$2^1 = 2$
100	$2^2 = 4$
1000	$2^3 = 8$
10000	$2^4 = 16$
100000	$2^5 = 32$
1000000	$2^6 = 64$
10000000	$2^7 = 128$
⋮	⋮
$\underbrace{100 \dots 000}_{\text{几个0}}$	2^n

如果一个二进制整数要化为十进制数,只要将它的最后一位乘以 2^0 , 最后第二位乘以 2^1 , 依此类推,将各项相加就得到用十进制数表示的数。如:

$$\begin{aligned} (101101)_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 2^5 + 2^3 + 2^2 + 2^0 \end{aligned}$$

$$=32+8+4+1=(45)_{10}$$

三 二进制数和八进制数转换(整数)

由于二进制数写起来很长,很难记,为方便起见,常将二进制数由低向高每三位组成一组,如 10110101111 可分为 10, 110, 101, 111 四组:每一组(包括三位二进制数)代表一个从 0 到 7 之间的数,因为三位的二进制数是不会等于或大于 8 的, $(111)_2 = (7)_{10}$, 也就是说,以三位二进制作为一组(位)的数是逢八进一的, $(8)_{10} = 2^3 = (1000)_2$ 就需要四位二进制数表示,即要向前一组数进一位。这种逢八进一的数称八进制数。现分别把上面的数据每三位一组用八进制表示:

$$\begin{array}{cccc} 10 & 110 & 101 & 111 \\ \hline 2 & 6 & 5 & 7 \end{array}$$

也就是说 10110101111 的八进制数为 2657(注意: 10110101111 的以十进制表示的数为 1455 而不是 2657,读者可自己转换一下)。

八进制数和二进制数很容易相互转换。一个二进制数要转化为八进制数,只需将每三位二进制的数用一个八进制数表示即可。反之,如果知道一个八进制数,要化为二进制的数,只需将每位八进制数分别用三位二进制数表示即可。如八进制数 10500 用二进制数表示:

$$\begin{array}{ccccc} 1 & 0 & 5 & 0 & 0 \\ \hline 001 & 000 & 101 & 000 & 000 \end{array}$$

即 001000101000000。

四 八进制与十进制的转换(整数)

如果要将一个八进制数化为十进制数,只要把它最后一位乘以 8^0 , 最后第二位乘以 8^1 , ... 依次类推,最后将各项相加即可。如:

$$(105)_8 = 1 \times 8^2 + 0 \times 8^1 + 5 \times 8^0 = 64 + 5 = (69)_{10}$$

八进制中的 105 等于十进制中的 69。

反之,一个十进制整数要化为八进制数,只需将它不断除以 8,其余数的排列(由最后一个余数开始)就是以八进制表示的数,如:

$$\begin{array}{r} 8 \overline{) 10} \quad (2) \\ 8 \overline{) 1} \quad (1) \\ \hline 0 \end{array}$$

$$(10)_{10} = (12)_8$$

$$\begin{array}{r} 8 \overline{) 69} \quad (5) \\ 8 \overline{) 8} \quad (0) \\ 8 \overline{) 1} \quad (1) \\ \hline 0 \end{array}$$

$$(69)_{10} = (105)_8$$

$$\begin{array}{r} 8 \overline{) 128} \quad (0) \\ 8 \overline{) 16} \quad (0) \\ 8 \overline{) 2} \quad (2) \\ \hline 0 \end{array}$$

$$(128)_{10} = (200)_8$$

在应用时,必须弄清楚所接触的数是二进制数? 八进制数? 或十进制数? 例如二进制中的 10 和 100 应分别读作“壹零”和“壹零零”,而不要误读作“拾”和“一百”。

五 二进制与十六进制的转换

1. 要把二进制数转换为十六进制数,可把数从小数点开始向左、右每四位分节,不足四位

则补零,然后写出对应的十六进制数。

四种进制的对应关系见表 1-3。

表 1-3

十进制	二进制	八进制	十六进制
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

例 $(10010001 \cdot 1011)_2 = (1001 \cdot 0001 \cdot 1011)_2$

$$= (91 \cdot B)_{16}$$

2. 把十六进制数转换为二进制数,方法与上例相反。

例 $(91 \cdot B)_{16} = (1001 \cdot 0001 \cdot 1011)$

$$= (10010001 \cdot 1011)_2$$

这时,若整数部分前面和小数部分后面有多余的 0,则应去掉。

3. 十六进制与十进制转换

(1) 二进制过渡法。即先把十六进制数转换成二进制,然后再转换成十进制,反之亦然。

例 1 $(11)_{16} = (1011)_2 = (B)_{10}$

例 2 $(B)_{16} = (1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
 $= (10)_{10}$

(2) 用反复除 16 取余和乘 16 取整法,方式同除 2 取余和下述的乘 2 取整相同,但因过程烦琐,使用不多。

六 小数的数制转换

1. 十——二转换

例 把十进制数 0.2 转换为二进制数。

解 先把 0.2 反复乘 2, 其过程如下:

	取整数部分	排列方向
0.2		
× 2		
0.4	0	
× 2		
0.8	0	
× 2		
1.6	1	
取 0.6		
× 2		
1.2	1	
取 0.2		
× 2		
0.4	0	

于是得 $(0.2)_{10} = (0.0011)_2$

由此可得算法: 带小数的十进制数转换为二进制数, 只要将此小数乘 2, 取其整数; 当积有整数后, 只把纯小数继续乘 2, 取其整数。如此反复进行, 直到小数部分为 0 或达到要求的精度为止, 再把每次乘积的整数由上而下排列, 前面加上小数点, 即构成该数的二进制表示的数, 此法为乘 2 取整法。

上例只要求小数点后四位, 第五位为 0, 可舍去, 其原则是 1 进 0 舍 (即四舍五入)。

当一个数既有整数, 又有小数时, 可分别对整数、小数进行转换。

例 $(13.2)_{10} = (?)_2$

解 把 $(13.2)_{10}$ 分为 $(13)_{10}$ 和 $(0.2)_{10}$

而 $(13)_{10} = (1101)_2$

$(0.2)_{10} = (0.0011)_2$

则 $(13.2)_{10} = (1101.0011)_2$

2. 二——十转换

方法与整数部分相似, 只要把 2 的方次 (即数值位的“权”) 改为负号, 并在 2^0 后 (或在 2^{-1} 前) 面加上小数点即可。

例 $(11.01)_2 = (?)_{10}$

解 $(11.01)_2 = 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$

$$= 2 + 1 + 0 + 1/4$$

$$= (3.01)_{10}$$

3. 八——十转换

可以把数的整数和小数分别处理, 整数处理同前述; 小数处理采用乘 8 取整法, 与乘 2 取整法相同。

§3. 语言及解题步骤

一 程序设计语言

计算机语言系统包括机器语言、汇编语言和高级语言。

机器语言是面向机器的；它是由机器可识别的二进制代码(0和1)组成的机器指令的集合。这种语言可读性、通用性很差，一般很少使用。

汇编语言是用一些特定的助记符号以取代机器语言中的二进制编码，比机器语言有了进一步改进，但它仍然是与机器有关，通用性差，因此，使用得也不广泛。

高级语言是接近人类自然语言和数学语言的，面向问题的程序设计语言，因而通用性好。但是用高级语言编写的源程序必须通过翻译程序转换成机器语言的目的程序，机器才能执行。翻译有解释和编译两种方式。

编译方式是：事先编好一个称为编译程序的机器指令程序，并放在计算机中。把用高级语言写的源程序输入计算机；编译程序便把源程序整个地翻译成用机器指令表示的目的程序。然后执行该目的程序，得到计算结果。如图1-3所示。

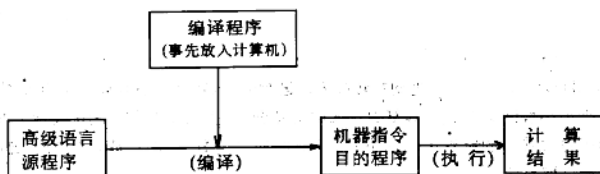


图 1-3

解释方式是：事先编好一个称为解释程序的机器指令程序，并放在计算机中。当高级语言源程序输入计算机后，它并不是象编译方式那样把源程序整个地翻译成目的程序，然后再执行该目的程序，而是逐句地翻译，译出一句立即执行，即边解释边执行。见图1-4。这种方式比编译方式多费机器时间，但可少占计算机的内存。

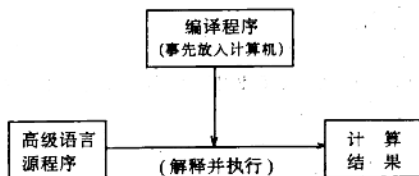


图 1-4

FORTRAN、PASCAL、ALGOL、COBOL、C语言、BASIC等高级语言采用编译执行方式，而大多数BASIC高级语言采用解释方式。

二 解题的一般步骤

1. 分析问题,确定数学模型。当接到一个项目之后,要对问题进行认真的分析,找出其中的相互关系,归纳为准确的数学公式,即为数学模型。

2. 设计算法步骤,画出程序流程图。将所得到的数学公式,进行实现程序化的具体步骤处理,反复推敲,力求合理。根据结构化理论画出 N-S 图。

3. 编出程序。根据 N-S 图,用高级语言,逐步编写出程序。本书所讨论的,是用 BASIC 语言编制的程序。

4. 上机调试运行。编出的程序,一般难免有问题,通过上机实际运行调试程序,发现并解决问题。

§4. 常量和变量

BASIC 语言是由 FORTRAN 语言简化来的,是 Beginner's All-Purpose Sybolic Instruction Code 的缩写,即“初学者通用符号指令代码。”

BASIC 语言的基本要素有常量、变量、函数和表达式等。这里,我们首先讨论常量和变量。函数和表达式将在其后讨论。

一 常量

BASIC 中的常量,是指在程序执行时,数值保持固定不变的量。

常量一般可分为三种:数值型、布尔型和字符型。

1. 数值型常量

在 BASIC 中,数值型常量就是常数。可分为整型和实型两类。

(1) 整型常数

不带小数点的常数称为整型常数,简称整数。通常使用十进制数,二个字节,数字范围为 $-32768 \sim 32768$,超出时为溢出,系统发出错误信息。

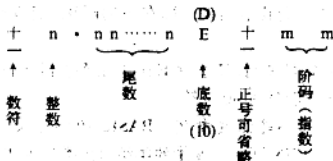
在 BASIC 语言中,整数内不允许用逗号分位。例如:10000 不能写成 10,000。

(2) 实型常数

带有小数部分的数均为实型常数,即实数。BASIC 中表示实数的方法有两种,一种是常用的方法,例如:

-1.414 3.14159 8.88

另一种是科学记数法,即指数法,其格式为:



这一方法是用 E 代表幂指数的底数 10,数为正,可省略数前的“+”号;小数点前必须而且只能有一非零数字;阶码只能是整数。

在运算中,整数不会有误差,而实数则会产生微小的误差。例如,把十进制数 0.1 转换成

二进制数时,就会有误差产生。

实型常数又可分为定点数和浮点数。

小数点固定的常数称为定点数。例如, 13.2, -4.8, -0.0035 等均为定点数。

用指数形式(科学记数法)表示的是浮点数。其基数,单精度型用字母 E,双精度型用字母 D。例如:

$$3E-2 \qquad 432D+5$$

浮点数所表示数的范围为 2.938×10^{38} 到 1.7014×10^{-38} 之间的任何正数或负数。对于绝对值小于此范围的数,系统作 0 处理,称为机器零;绝对值大于此范围的数,则产生溢出,系统显示出错误信息。

单精度和双精度实数的区别,在于单精度的存储和输出的 7 位有效数值中,只有前 6 位是精确的;而双精度型的存储和输出为 17 位有效数字,但只能有 16 位输出,精度虽高,但占内存多,而且运算速度较慢。

一个定点数可用多种形式的浮点数来表示,例如,定点数 0.88 可化成以下浮点数:

$$8.8E-1 \qquad 880E-3 \qquad 0.88E+1$$

在这些浮点数中,对于能满足不等式:

$$1 \leq | \text{尾数} | < 10$$

称为标准化的浮点数,唯一的是 $8.8E-1$ 。

计算机对于实数的输入,可按定点数,也可用浮点数;输出时,对特大和极小的数,则一律按标准化的浮点数形式;一般数则以定点数形式输出。

2. 布尔型常量

此即为逻辑常数,是一个逻辑值,即“真”和“假”。一般用“1”表示“真”(TRUE),用“0”表示“假”(FALSE)。主要用来表示关系运算或逻辑判断的结果。

3. 字符串型常量

它是用定界符(双引号)括起来的一串字符。例如:

“ABC”, “1945-1995”, “新时代”

字符串的大小用长度来计算,一个字符串的长度是指两引号之间的字符的个数。两字符串可以进行运算比较,以字符的 ASCII 码为标准,逐个进行。字符串长度为零称为空串,但不是空格,空格是一个字符。一个汉字算作两个 ASCII 码字符。

二 变量

变量是指在 BASIC 程序执行中,能改变其值的量。即在不同时刻,变量代表不同的数值。

IBM PC 机的 MS BASIC 的变量命名规定,变量名必须是由英文字母开头的并由字母、数字和小数点组成的字符序列,最多不能超过 40 字符。

值得注意的是:希腊字母 $\alpha, \beta, \gamma, \pi$ 等不能作变量名;BASIC 语句定义符也不许单独作变量名使用,但可嵌入变量名中,不许用 FN 开头。变量的类型可用类型说明符或类型说明语句来定义。详见表 1-4。

表 1-4

变 量 类 型			类型说明符	类型说明语句	占字节数
字 符 型			S	DEFSTR	
数 值 型	简单型	整 型	%	DEFINT	2
		单精度型	! (或无)	DEFSNG	4
		双精度型	=	DEFDBL	8
	下标型				

组成变量名的格式有两种(下标型除外):

(1) 用类型说明符的格式为

< 变量名字符序列 > < 后跟类型说明符 >

例

M %

M 为整型变量

A!, A

A 为单精度型变量

X =

X5 为双精度型变量

BLS

BL 为字符型变量

(2) 用类型说明语句的格式为

整 型: < 行 号 > DEFINT < 字母表 >

单精度型: < 行 号 > DEFSNG < 字母表 >

双精度型: < 行 号 > DEFDBL < 字母表 >

字符串型: < 行 号 > DEFSTR < 字母表 >

字母表的格式: 用逗号分隔的单个字母, 或用“-”连接字母的区间。

例: 10 DEFINT A-D ; A, B, C, D 为整型变量

20 DEFSNG F ; F 为单精度型变量

30 DEFDBL X ; X 为双精度型变量

40 DEFSTR Y, Z ; Y, Z 为字符串型变量

§5. 表达式

在 BASIC 中, 表达式是指用规定的运算符、圆括号, 将常数、变量及函数连接成的式子。它可以分为算术、关系、逻辑和字符串四种表达式。

一 算术表达式

在 BASIC 中, 用算术运算符、圆括号把常数、变量及函数连接的式子, 称为算术表达式, 它的运算结果是一个数值。书写时应注意以下几点:

(1) 不要漏写乘号和括号, 不要把除法写成分式。

(2) 只能成对使用圆括号, 不允许用花括号或方括号。

(3) 表达式中每个字符占一格, 并写在同一横线上。

(4) 乘幂运算时, 例如 X^Y , 若 Y 为整数, 当 $X < 0$, 且为低幂次时, 可用连乘形式, 不要用