

966370

TP312

3131K3A

B511.2

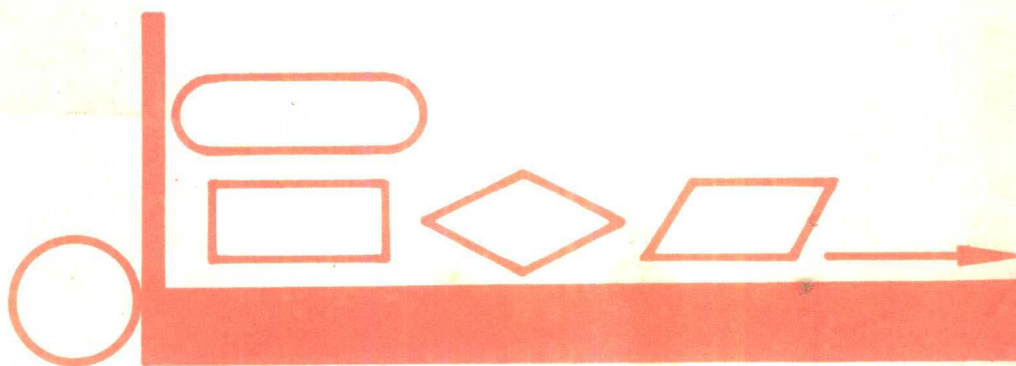
BASIC

语言

结构化 程序设计教程 学习辅导

谭浩强 卜家岐 编著

中国科学技术出版社



3A

内 容 提 要

本书是与谭浩强主编的《BASIC 语言—结构化程序设计教程》配套使用的参考教材。内容包括三大部分：一、教材各章的基本要求，对教材各章内容的进一步补充讲解以及各章习题参考解答；二、BASIC 上机操作指南；三、上机实验指示书。本书可以帮助读者更好地掌握 BASIC 的知识及其应用，掌握编程序方法和上机方法。本书内容丰富，通俗易懂，共包括 177 个习题和程序，是一本学习 BASIC 的较好的参考教材。可供高等学校学生和计算机学习班做教学参考书，也可供自学者使用。

(京)新登字 175 号

BASIC 语言结构化程序设计教程学习辅导

谭浩强 卜家岐 编著

责任编辑 朱桂兰

封面设计 赵一东

*

中国科学技术出版社出版(北京海淀区白石桥路 32 号)

新华书店北京发行所发行 各地新华书店经售

北京市燕山联营印刷厂印刷

*

开本：787×1092 毫米 1/16 印张：15 字数：357 千字

1992 年 10 月第 1 版 1993 年 2 月第 2 次印刷

印数：7501-23500 册 定价：5.50 元

ISBN 7-5046-0730-4/TP·26

第一部分

各章的基本要求,补充说明和习题参考解答

第一章 关于计算机的一般知识

§ 1.1 本章基本要求

1. 了解计算机发展的简史,工作特点和应用领域。
2. 初步了解计算机系统(包括硬件系统和软件系统)的组成。
3. 计算机内存的组织形式和数据的二进制表示法。
4. 十进制、八进制、十六进制和二进制之间的相互转换。
5. 弄清机器语言和高级语言的概念。
6. 初步了解计算机解题的一般步骤。

§ 1.2 关于本章内容的进一步说明

1.2.1 计算机发展简史可简单归纳如下:

从1946年世界上第一台计算机ENIAC开始从元器件的角度来看,计算机的发展经历了电子管、晶体管、集成电路和大规模集成电路四个阶段。这四个时期段的计算机都基于一个基本原理——Von Neumann(冯·诺依曼)原理,即以二进制和程序存贮控制为基础的结构思想,这个思想由美籍数学家冯·诺依曼于1946年最早提出来。它确立了各代计算机的基本原理。本章讲到的计算机内存的组织形式和数据的二进制表示法就是二进制的结构思想;而程序存贮控制,就是人们预先把设计好的程序输入计算机内存(即内存贮器)。运行时,计算机就会根据人们预定的意图进行工作。

目前,科学家正在研制的“第五代计算机”,也称非冯·诺依曼计算机,它采取全新的工作原理和体系结构,更接近于人们的思维,即具有“推理”和“人工智能”的功能,这将是计算机科学技术的一项重大突破,被称为“第二次计算机革命”。

1.2.2 计算机系统的组成(硬件系统和软件系统)

1. 硬件系统由图 1.1.1 所示的部件组成 2. 软件系统由图 1.1.2 所示的部分组成

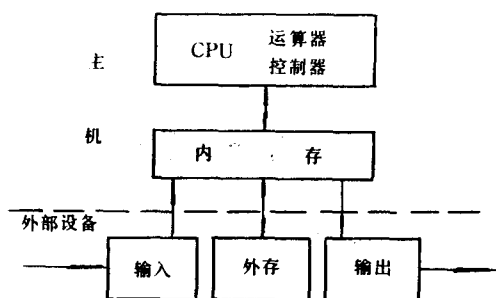


图 1.1.1.

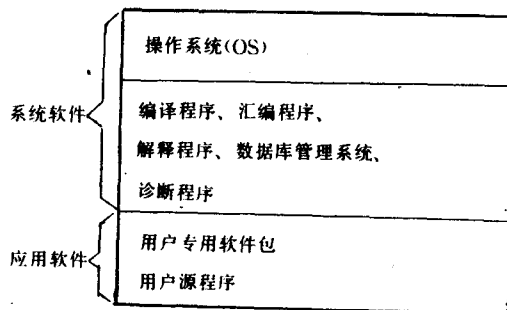


图 1.1.2

计算机的硬件系统指的是具体的设备,即实物;而软件系统指的是程序。计算机正常工作,离开硬件固然不行,而只有硬件,没有必要的软件也不行,计算机做任何工作,都是通过存贮在内存中的软件来指挥的,这是一个重要的概念。

1.2.3 内存的组织形式和数据的二进制表示法

程序和数据可以存贮在内存中,那么内存是由什么组成的呢?它是由千千万万具有相同功能的电子触发器所组成,这种触发器只有两种稳定的工作状态,通常以 0 和 1 来代表,而每个触发器用术语“位”(bit)来称呼,因为它可以代表二进制数的一个基本单元,计算机的内存就是一个庞大的二进制基本单元——电子触发器的集合体。

由“位”组成“字节”(byte),又由“字节”组成“字”(word)。一个“字”一般可以存放一条指令或一个数据,所以也称一个内存单元。一般计算机 8 位组成一个字节,不同数目的字节组成一个字,若以 4 个字节组成一个字,那么这台计算机的字长为 32 位。

计算机上所应用的所有字符(字母、数字或其他专用符号)必须先化成二进制的形式才能存放到计算机的内存单元中,这些字符和二进制代码之间的对应关系很多计算机系统采用 ASCII(American Standard Code for Information Interchange,即“美国标准信息交换码”)。用二进制代码来表示一个符号时,位数往往很多,数字冗长,容易搞错,也不便于记忆。因此,常用八进制或十六进制来书写。而人们日常用的是十进制,计算机输入和输出的数值一般也用十进制表示,MS-BASIC 还可用八进制或十六进制整型常数,因此要懂得二、八、十六进制和十进制之间的转换

1.2.3 十、八、十六进制和二进制之间的转换

大家熟悉的十进制有二个特点:

- (1)用 0,1...,9 十个数字符号来表示;
- (2)逢十进一。

同样原理,十进制的特点是:

(1)用 0,1 两个数字符号来表示;

(2)逢二进一。它是最简单的进位制,也是机器目前所能接受的。用二进制组成的代码,也称机器码。

八进制的特点是:

(1)用 0,1,⋯,7 八个数字符号来表示;

(2)逢八进一。

十六进制的特点是:

(1)用 0,1,⋯,9 以及 A,B,C,D,E,F,十六个字符来表示 0~15,其中 A~F 分别代表 10~15;

(2)逢十六进一。

(一)十进制和二进制的相互转换 十进制转换成二进制可用除 2 求余的长除法,如求 20 的二进制:

$$\begin{array}{r} 2 \overline{) 20} \quad (0) \\ 2 \overline{) 10} \quad (0) \\ \quad 2 \overline{) 5} \quad (1) \\ \quad \quad 2 \overline{) 2} \quad (0) \\ \quad \quad \quad 2 \overline{) 1} \quad (1) \\ \quad \quad \quad \quad 0 \end{array} \quad \begin{array}{l} \uparrow \\ \text{低位} \\ \text{高位} \end{array}$$

得 $(20)_{10} = (10100)_2$

二进制转换成十进制可以借助于数的幂级数形式表达式来完成,从十进制以 10 为基数的幂级数形式很容易得到二进制以 2 为基数的幂级数形式表达式。如

$$\begin{array}{ccc} (2 & 1 & 1)_{10} = 2 \times 10^2 + 1 \times 10^1 + 1 \times 10^0 \\ \downarrow & \downarrow & \downarrow \\ \text{每位的权} & 10^2 & 10^1 & 10^0 & \text{(以十为底的指数形式)} \\ (1 & 0 & 1 & 1)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ \text{每位的权} & 2^3 & 2^2 & 2^1 & 2^0 & \text{(以二为底的指数形式)} \\ & & & & = 8 + 2 + 1 = (11)_{10} \end{array}$$

从二进制的幂级数形式表达式很容易得到它的十进制数值。

(二)八进制、十六进制和二进制的相互转换

已知八进制用 $\underline{0} \sim \underline{7}$ 八个数字组成

相应的二进制是 000~111;

而十六进制用 $\underline{0} \sim \underline{F}$ 十六个数字字母组成,

相应的二进制是 0000~1111

所以,二进制三位代表八进制的一位;二进制四位代表十六进制的一位。转换时,是从二进制数的最右边开始向左数,每三位或四位一组,到最左边位数不足时,用 0 代替。这样,相互间的转换就容易解决了。如:

$$\begin{array}{rcccccccl}
 & 1 & & 0 & & 5 & & 0 & & 0 & \longrightarrow & \text{八进制} & 10500 \\
 \hline
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \longrightarrow & \text{二进制} & 0001000101000000 \\
 \hline
 & 1 & & & & 1 & & & & 4 & & & & & 0 & \longrightarrow & \text{十六进制} & 1140
 \end{array}$$

1.2.5. 机器语言和高级语言

计算机内部传送的各种信息,都是二进制的,也就是说计算机只懂得二进制编码的语言。这种面向机器的语言,所以叫机器语言,也称“低级语言”。它与人们习惯用的语言差别大,而且不同机器有不同的机器语言,它们互不通用,这给计算机的推广造成很大的障碍。

50年代以来,人们为普及计算机的应用,已发明创造了几百种“高级语言”。而 BASIC 语言就是其中的最易学而有效的高级语言之一,所谓高级语言就“高”在语言本身更接近于人们习惯用的日常语言和数学公式。例如,打印数据用 PRINT 语句,打开文件用 OPEN 语句,计算 X 的正弦函数就写 SIN(X)就可以了。

可是,如前所说,计算机只懂得机器语言,不能直接接受和执行高级语言写的程序(通常称“源程序”),这就要有“翻译”,担任这翻译任务的通常有“解释程序”和“编译程序”这两种类型的软件来完成,它们的区别在于“翻译”方法不同。

“解释程序”只善于逐句翻译,译一句执行一句,所以耗机时长但占内存少。而“编译程序”可以把整个源程序全部翻译成机器能执行的机器语言目的程序(机器语言),要用时可直接运行这个目的程序,所以运行时,耗机时短,也就是执行速度快。但占内存多。

§ 1.3 习题及参考解答

1.1 请用你自己的语言说明什么是电子计算机。电子计算机与袖珍计算器的主要区别在哪里?

【解】电子计算机是一个高速进行操作、具有内部存储能力、由程序控制操作过程的自动的电子装置。它与袖珍计算器的主要区别是:计算器没有内存,无法保存更多的信息,因而也无法实现程序的自动控制,只能用人工逐步控制计算过程。此外计算机不仅能进行数值计算,还能进行数据处理和控制等,它实际上是一个“信息处理机”,其功能比计算器丰富得多。

1.2 请你从自己的见闻中举出计算机应用的五个例子。

【解】:略、请读者自己回答

1.3 请说明一个计算机系统应包括哪些部分(分别说明硬件系统和软件系统所包括的部分)

【解】略。

1.4 计算机的内存贮器的作用是什么?它本身有无运算功能?

【解】略。

1.5 计算机内存贮器的组织形式是怎样的?什么叫“位”、“字节”、“字”?存贮单元的“地址”是指什么?“地址”与存贮单元中的信息有什么关系?

【解】略。

1.6 计算机内部为什么要采用二进制形式存贮数据和进行运算?为什么不用十进制?

【解】因为目前所使用的电子元器件一般有两种稳定工作状态(例如灯泡的亮与灭,二极管、三极管的导通和截止、脉冲的有无等),用来代表二进制数0和1比较方便。而且二进制数的计算公式简单,即只有4个加法公式和4个乘法公式。

$$0+0=0, 0+1=1, 1+0=1, 1+1=10$$

$$0 \times 0=0, 0 \times 1=0, 1 \times 0=0, 1 \times 1=1$$

总之,二进制数容易实现,容易运算。

1.7 把下列各十进制数用二进制形式表示:

(1)92 (2)128 (3)136 (4)246 (5)1024

【解】

$$\begin{aligned}(1) \quad (92)_{10} &= 64 + 16 + 8 + 4 \\ &= 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \\ &= (1011100)_2\end{aligned}$$

$$\begin{aligned}(2) \quad (128)_{10} &= 1 \times 2^7 \\ &= (10000000)_2\end{aligned}$$

$$\begin{aligned}(3) \quad (136)_{10} &= 128 + 8 \\ &= 1 \times 2^7 + 1 \times 2^3 \\ &= (10001000)_2\end{aligned}$$

$$\begin{aligned}(4) \quad (246)_{10} &= 128 + 64 + 32 + 16 + 4 + 2 \\ &= 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + \\ &\quad 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= (11110110)_2\end{aligned}$$

$$\begin{aligned}(5) \quad (1024)_{10} &= 1 \times 2^{10} + 0 \times 2^9 + 0 \times 2^8 + 0 \times 2^7 + 0 \times 2^6 \\ &\quad + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ &= (10000000000)_2\end{aligned}$$

1.8 把下列各二进制数化成十进制数:

(1)110 (2)1010 (3)101111 (4)11000010 (5)1011010

【解】

$$(1) \quad (110)_2 = (6)_{10}$$

$$\begin{aligned}(2) \quad (1010)_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 8 + 2 = (10)_{10}\end{aligned}$$

$$\begin{aligned}(3) \quad (101111)_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= (47)_{10}\end{aligned}$$

$$\begin{aligned}(4) \quad (110000)_2 &= 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ &= (48)_{10}\end{aligned}$$

$$\begin{aligned}(5) \quad (1011010)_2 &= 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ &= (90)_{10}\end{aligned}$$

1.9 把下面二进制数用八进制和十六进制形式表示:

(1)10110111 (2)110001 (3)0101010101 (4)1111111111 (5)1000001

【解】

- (1) $(10110111)_2 = (267)_8$
 $(10110111)_2 = (B7)_{16}$
- (2) $(110001)_2 = (61)_8$
 $(110001)_2 = (31)_{16}$
- (3) $(0101010101)_2 = (525)_8$
 $(0101010101)_2 = (155)_{16}$
- (4) $(1111111111)_2 = (1777)_8$
 $(1111111111)_2 = (3FF)_{16}$
- (5) $(1000001)_2 = (101)_8$
 $(1000001)_2 = (41)_{16}$

1.10 试述机器语言与高级语言的特点,为什么要使用高级语言? 计算机为什么能执行高级语言程序?

【解】略。

1.11 用计算机解题一般应经历哪几个阶段? 为什么要书写程序文档?

【解】略。

第二章 最简单 BASIC 程序分析和流程图

§ 2.1 本章基本要求

1. BASIC 语言的基本特点和 MS-BASIC(BASICA)字符集。
2. BASIC 程序的构成和基本规则。
3. 数值型常数的表示方法。
4. 掌握变量类型和定义。
5. 熟练掌握算术表达式的书写方法和运算法则。
6. 掌握常用标准函数的应用。
7. 懂得结构化程序设计要点和组成它的三种基本结构。
8. 懂得用常用流程图和 N-S 图来表示算法。

§ 2.2 关于本章内容的进一步说明

2.2.1 MS-BASIC(BASICA)的基本字符集

- (1) 26 个英文字母;大写 A~Z 或小写 a~z。
- (2) 10 个数字:0,1,...,9。
- (3) 25 个专用字符:
+, -, *, /, \ (整除号), ^ (乘幂号)
(,), =, =, =, ', ., ;, :, &, ?, <, >, ",
%, #, \$, !, _ (下划线)。

2.2.2 数值型常数的表示方法

BASICA 中有两类常量:数值型常量和字符型常量。字符型常量我们留在第八章讲解,而数值型常量(即常数)又分整型常数和实型常数,其中实型常数又有单精度常数和双精度常数之分。

(一) **整型常数** 在微机中一般在内存中以二个字节(每字节 8 位)共 16 位的二进制位来存储一个整数,其中前 1 位是符号位,后 15 位为数字位。因此整数的范围为 $-2^{15} \sim 2^{15}-1$, 即 $-32768 \sim 32767$ 。整型常数可以有十进制整型数,八进制整型数和十六进制整型数几种形式。

1. 十进制整型数

十进制整型数在 $-32768 \sim +32767$ 之间,如:368,0,5,-86 等。

2. 八进制整型数

第一位数字前加符号 &, 如: &184, &687356。

八进制整型数最多可含 6 位数字。

3. 十六进制整型数

第一位数前冠以 &H, 如: &H128, &HFA5。

十六进制整型数最多可含 4 位数字。

(二) **实型常数** 它有两种表示方法, 即小数表示法和指数表示法(科学表示法)。

1. 小数表示法, 如: 237.8, -0.58, 7.0。

2. 用指数形式表示, 如: 0.76E-1 代表 0.76×10^{-1} - 5.1E10 代表 -5.1×10^{10} 。

注意:

(1) 单独的指数部分是非法的, 如: E10, E-2 等。

(2) 字母 E 后面的正号可以省去, 如: 1.5E+2 和 1.5E2 相同。

(3) 字母 E 前面可以是整数或小数, 如: 2E-3, .34E4 等。

3. 单精度实数(占 4 个字节)。

包括: (1) 小于等于 7 位有效数字的小数(如: 1.23456, -876.321) 以及超过 -32768 ~ 32767 范围的整数(不超过 7 位数字), 如: 76842, -112233。

(2) 用字母 E 表示的实型数, 如: 108E-3

(3) 数字末尾带有“!”的数, 如: 25.1!, 0.7548!, 72!

(关于单精度数和双精度数的表示形式请参阅教材第十二章 § 12.2 节)

4. 双精度实数(占 8 个字节)

比单精度提供更多的有效数字位, 包括:

(1) 含 8 位或大于 8 位的有效数字位。如: 3421897696, 7864.8456705

(2) 带指数时, 用字母 D 代替 E 的数。如:

672.3D-8 代表 672.3×10^{-8} 8.45D4 代表 8.45×10^4

(3) 数字末尾带有“#”号的数, 如 37829.45#, 67.0#, 21# 等计算机运算时, 有效位数和精度是双精度高, 但运算速度大大降低, 所以在一般情况下, 用单精度数。

2.2.3 关于简单变量

程序中出现一个变量, 计算机就会分配给一个存储单元, 即变量是表示某一组内存单元的符号。(例如一个实型变量名代表 4 个字节的内存单元)

(一) **简单变量的定义** MS-BASIC 中简单变量的定义是: 以字母开头, 由不超过 40 个字符(字母, 数字或小数点)构成的字符串组成。

(二) **简单变量的类型** 可分为整型变量, 单精度变量和双精度变量。

(1) 整型变量(占 2 个字节), 变量后加“%”表示。如 ABC%, M%, X1% 等

(2) 单精度变量(占 4 字节), 变量后加“!”表示。如 MIN!, A2.3!, X5! 等

(3) 双精度变量(占 8 个字节), 变量后加“#”表示。如 KG#, M03#, BETA# 等

需要指出的是, 一个变量, 如果其尾部不加任何类型说明符(% , ! , #) 时, 则作单精度变量处理, 也就是说变量类型隐含为单精度(在实数范围内), 这也是读者平时应用最多的, 你自己是否意识到了呢?

(三)变量的类型说明语句 除上所述在变量后面加类型说明符来定义变量类型外,我们还可以用类型说明语句来说明变量类型。

一般格式是:

DEF<类型><字母>[—<字母>],...

<类型>可以是:INT(整型)

SNG(单精度)

DBL(双精度)

STR(字符型)

【例】DEFINT A,X—Z

表示字母 A 和 X,Y,E 开头的变量为整型变量。

DEFSNG I—K,N

表示字母 I,J,K,N 开头的变量为单精度变量。

注意,类型说明语句应放在程序的开头部分。

2.2.4 算术表达式

BASIC 中的算术表达式可以归纳为:

1. 一个常数、变量或函数用算术表达式的最简单的形式。如 $3A \cdot \sin(A)$ 等。
2. 把常数、变量或函数用算术运算符连接起来的式子(有时包含括号)是算术表达式。如 $(A+B) \cdot \sin(C)$ 。

读者一般对 2 容易理解,对 1 容易忽视,有时容易跟赋值语句混淆,当提问

PI=3.14159

是不是表达式时,很多人会认为是表达式,其实它是赋值语句,不是表达式,这里表达式是一个常数 3.14159 将其值赋给变量 PI

(一)MS—BASIC 的算术运算符包括: +, —, *, /, \ (整除), ^ (乘幂), MOD (求余)

(二)运算优先次序 $\square \rightarrow \square \rightarrow \square \rightarrow \square, / \rightarrow \square \rightarrow \square \rightarrow \square, -$

(三)其他还要说明的

(1)整除运算时,先将除数和被除数四舍五入取整,且要求取整的数必须在 -32768 ~ 32767 之间,然后相除,商的小数部分无条件地被舍去(不四舍五入)。

【例】

$10 \setminus 3 = 3, -10 \setminus 3 = -3$

$26.82 \setminus 12.3 = 27 \setminus 12 = 12$

$1 \setminus 2 = 0, 3 \setminus 8 = 0, -5 \setminus 7 = 0$

(2)求余运算指的是两个整数相除,结果取余数。若参加运算的两数是非整数,则四舍五入取整后再运算。而余数的符号跟被除数相同,与除数的符号无关。

【例】

$8 \text{ MOD } 3 = 2, -8 \text{ MOD } 3 = -2$

$8 \text{ MOD } -5=3, 8 \text{ MOD } 5=3$

3. 两个整数相除, 结果为单精度数。

如 $1/2=0.5, 3/8=0.375$

4. 两个不同类型的数据进行四则运算时, 其结果和精度高的数据类型相同。

如 $A\%+B$ 结果为单精度数

$C! * D\#$ 结果为双精度数

$3\backslash Y\#$ 结果为双精度数

5. 乘幂运算和函数运算结果为单精度, 跟参与运算的数据类型无关。

如 $\text{LOG}(X), \text{LOG}(X\#), A \wedge 2$ 均为单精度数。

2.2.5 算法的概念

所谓“算法”, 粗略地讲, 是指解题的具体步骤, 即把为解决某一问题所需进行的具体步骤一一详细地写出来。广义地说, 处理任何问题都有相应的算法, 例如太极拳的图解就是“打太极拳的算法”, 因为它指出了每一个步骤如何执行。又如一个学生一天的生活学习安排计划, 也是一个算法。当然, 我们讨论的是计算机算法, 即计算机能执行的算法。并不是任何算法都可以由计算机执行的, 例如, 打太极拳的算法, 计算机就不能执行, 而让计算机做一道四则运算题, 或解一道微分方程题, 计算机是可以实现的。

有的同志以为只要把问题原封不动地交给计算机, 计算机就会给出结果。这是一种误解; 至少目前是这样。例如你要求一个联立方程式的解, 你不能只把方程试输入计算机然后坐等解答, 计算机并不能代替人做所有事情。目前我们所使用的计算机只能是按照人们事先给定的步骤去进行工作。因此, 在拿到题目后, 必须设想好让计算机一步一步做什么。例如解联立方程式, 就需要按照人工解题的步骤, 一一列出让计算机执行的步骤, 这就是算法。因此, 程序设计的基础是设计算法。

一个完整的程序, 一般包括两个方面的要素: (1) 数据。这是计算机进行操作的对象。例如要求解 $z=x+y$, x, y, z 都是数据。数据有一定属性(例如有整型、实型...), 数据之间有一定联系(例如若干个数组成一个数组)。这就是数据结构。计算机语言中的数据类型就是体现数据结构的。(2) 对数据所施加的操作。即算法。

数据结构和算法是程序的两大要素。数据是原料, 算法是对原料的加工。正如一个菜谱, 指出为做出某一道菜, 所需要准备的原料以及如何对这些原料进行加工。因此, 进行程序设计时要清楚地安排好数据结构和设计出好的算法。著名计算机科学家沃思(Niklaus Wirth)有一个有名的公式:

程序 = 算法 + 数据结构

在初学者所遇到的简单程序中, 数据结构不会很复杂, 因此算法的设计就显得突出。读者从一开始就应该把主要注意力放在算法设计上。在拿到一个题目后不要匆匆忙忙动于编程序, 而应当仔细思考解题应采取的方法和步骤, 画出流程图。只要算法正确, 流程图无误, 那么用任何一种计算机语言编程序去解此题都是轻而易举的。不要形成这样的情况: 在学了某一种计算

机语言后,只会用该语言编程序,而在需要使用另一种语言时感到束手无策。学习计算机语言程序设计时,要注意,语言只是工具,目的是编出程序解决问题。各种计算机语言虽有不同之处,但它们的基本成份和规律是相似的,只要设计出算法,画出流程图,一些有关的计算机语言的知识,就能把由一种语言写的程序转换为另一种语言写的程序。

一个算法可以用不同的公式表示

(1)自然语言。即人们日常使用的语言(如汉语、英语等)。但用自然语言表示算法显得冗长而且有歧义性。

(2)用传统流程图

(3)用 N-S 流程图

(4)用伪代码。所谓“伪代码”是指一种介于自然语言与计算机语言之间的无严格规则的语言。例如,“输入 x 并累加之”的算法(见教材第 22 页图 2.6)用伪代码表示如下:

```
input x
while  $x \geq 0$ 
{
    sum + x  $\Rightarrow$  x
    print x
}
end
```

可以看出,它的书写形式灵活自由,不受任何语法规则限制。用它可以很容易地转换成任何一种计算机语言程序,也容易修改。

1. 传统流程图是各种计算机书籍中用得较多的一种流程图,它由不同形状的框、带箭头的流程线和必要的文字组成。这种流程图存在一些缺点,如没有表示循环结构的符号,使用带箭头的流程线,允许流程和随意转移,这是不宜提倡的。国际标准化组织 ISO(International Standard Organization)在 ANSI 流程图(即传统形式流程图)的基础上作了改进。主要是:(1)增加了表示循环的符号。(2)流程线由上到下和由左到右时可以不带箭头(但自下向上和自右向左时仍需要用箭头)。我国已接受 ISO 标准提出的流程图符号,并以此为基础制定出我国的标准。它规定的流程图符号见图 1.2.1 示

用这种流程图符号表示教材第 26 页图 2.18 如本书的图 1.2.2 所示那样。

由于结构化程序的推广,也大大减少了使用箭头的必要性,因此用这种改进的流程图更为方便。但这种流程图目前在国内远未完全推广开,多数书籍仍使用原来的 ANSI 流程图。因此读者应能看懂并会使用这两种流程图。

2. 关于 N-S 结构化流程图

N-S 结构化流程图是适应结构化程序而出现的一种新型流程图。它没有流向线,因而不会出现无规律的流程转移。N-S 流程图的基本图形元素是三种基本结构。因此,一个用 N-S 流程图表示的算法必然是一个结构化的算法。用 N-S 流程图描述算法形象直观,一目了然,能帮助阅读和理解程序。由于 N-S 流程图的形状象一个矩形的盒子(在“大盒子”中套“小盒子”)因此 N-S 流程图又称为“盒图”。

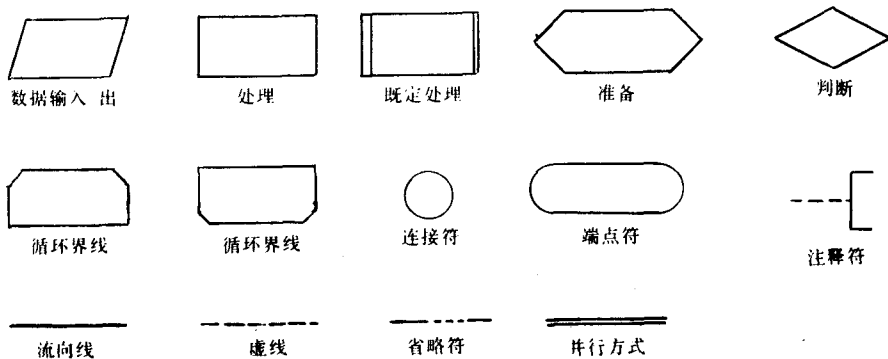


图 1.2.1

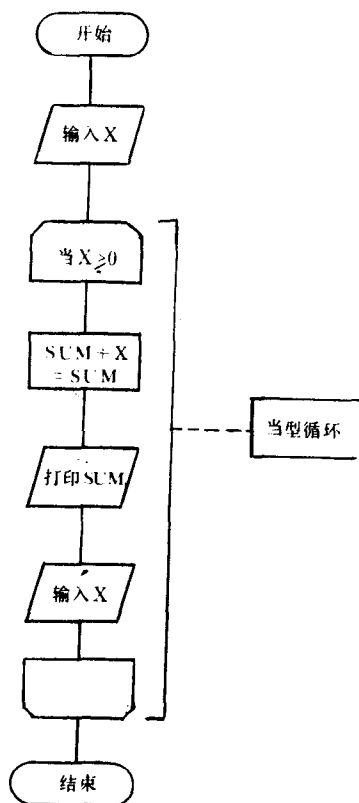


图 1.2.2

根据 n 的值决定			
n = 1	n = 2	n = 3	n = 4
执行 A 操作	执行 B 操作	执行 C 操作	执行 D 操作

图 1.2.3

除了三种基本结构有相应的 N-S 流程图基本图形元素以外,对于多分支结构,可以用如图 1.2.3 那样的图形元素表示。

任何算法都可以表示为结构化的算法。已经证明,用 N-S 结构图可以描述任何算法。

§ 2.3 习题及参考解答

1. 请分析 BASIC 语言的优缺点以及它的适用领域。

【解】BASIC 语言好懂易学,BASIC 的常用语句只有二三十种,BASIC 的命令和语句中使用的单词保留以及运算符号与英语中使用的单词以及数学中的运算符差不多。因此,易于理解和记忆。BASIC 语言功能较强,不仅适用于数值计算,而且还适合于数据处理,又能用于实时控制 BASIC 的打印格式灵活多样,用来处理小型的事务是很方便的,BASIC 还有绘图,音乐等功能,这是其它一些高级语言所不及的。

由于许多 BASIC 系统采取解释执行方式,执行速度比较慢,这对完成大型的复杂的任务是不利的,BASIC 提供的数据结构不如 PASCAL 和其它一些语言丰富,有些功能受限制,因

此 BASIC 一般适用于初学者用于相对小型的任务。

2. BASIC 程序中的行号有什么作用？

【解】BASIC 解释系统会把源程序中所有的语句行按行号大小顺序排列好，执行时依此顺序。长城机和 IBM-PC 机上使用的 BASICA 规定行号的范围为，从 0 到 65529，程序中行号可以不按大小顺序写。

3. BASIC 变量名的规则是什么？

【解】变量名的第一个字符必须为字母，其后的字符可以是字母，数字和小数点。不允许用 BASIC 的保留字，如：SIN, LET, GOTO, END 都不能作变量名。

4. 把下列各数由日常记数法转换成指数表示，形式或由指数表示形式转换成日常记数法。

- | | |
|-------------------|----------------------|
| (1) 9423456 | 【解】9.42E+6 |
| (2) 7123456000000 | 【解】7.123456E+12 |
| (3) 0.9999999 | 【解】9.999999E-01 |
| (4) 0.0000000004 | 【解】4.0E-10 |
| (5) 8.67876E+8 | 【解】867876000 |
| (6) 0.123456E-6 | 【解】0.000000123456 |
| (7) 3.1415928E+9 | 【解】3141592800 |
| (8) 2.567E-12 | 【解】0.000000000002567 |

5. 把下列代数式用 BASIC 表达式表示：

(1) $\frac{88 - 52 \times 63}{18 + 47 \div 3}$

【解】(88-52*63)/(18+47/3)

(2) $\frac{(\sin 45^\circ)^4}{e^2 \ln 5}$

【解】(SIN(3.14159*45/180))^4/(EXP(2)*LOG(5))

(3) $G \cdot \frac{m_1 m_2}{r^2}$

【解】G*(M1*M2/R^2)

6. 分别用传统流程图和 N-S 图表示下列问题求解的算法。

(1) 有两个变量 a、b 将它们的值互换。

【解】传统流程图(见图 1.2.4)。N-S 图(见图 1.2.5)。

(2) 送入计算机 10 个学生的成绩，需求将其中不及格的分数打印出来。

【解】传统流程图(见图 1.2.6)。N-S 图(见图 1.2.7)

7. 求 $1+2+3+\dots+10$ ，即 $10 \sum_{n=1}^{10} n$

【解】传统流程图：(见图 1.2.8)，N-S 图(见图 1.2.9)

8. 给出三角形三边 a, b, c 求三角形面积(要求先验算 a, b, c 是否符合构成三角形的条件。

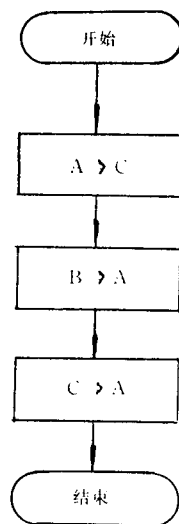


图 1.2.4

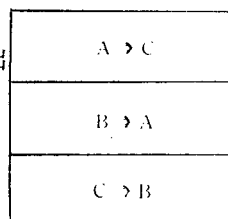


图 1.2.5

【解】传统程序图(见图 1.2.10)。N-S 图:见图 1.2.11

9. 求方程 $ax^2+bx+c=0$ 的根(考虑 $d=b^2-4ac$ 等于零,小于零和大于零三种情况。如果 $d<0$,则分别打印出复根的实部和虚部)。

【解】传统程序图:(见图 1.2.12)。N-S 图:(见图 1.2.13)。

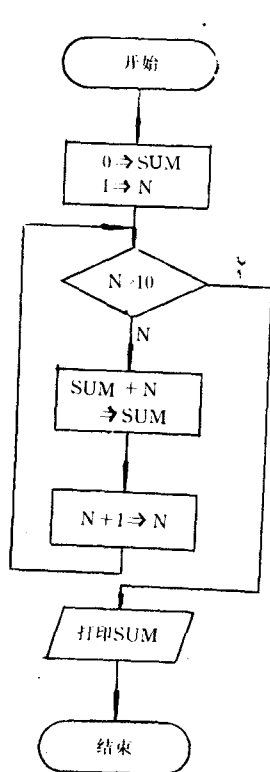


图 1.2.6

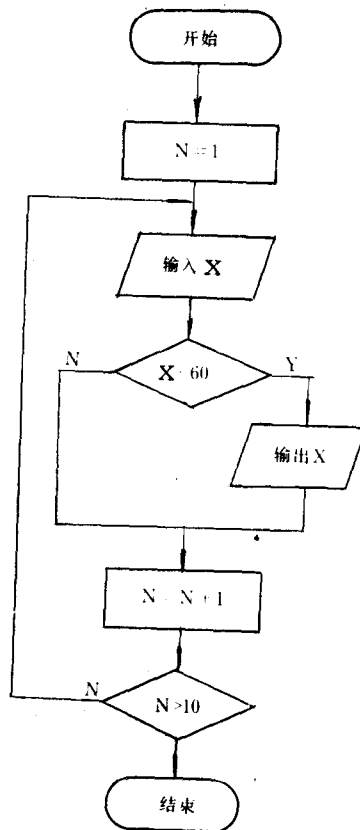


图 1.2.8

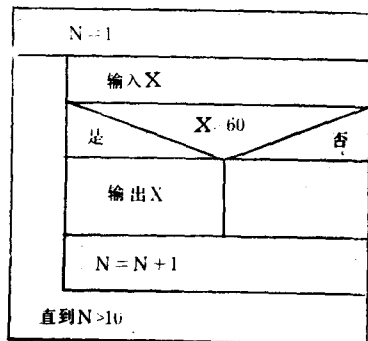


图 1.2.7

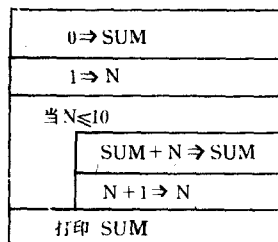


图 1.2.9

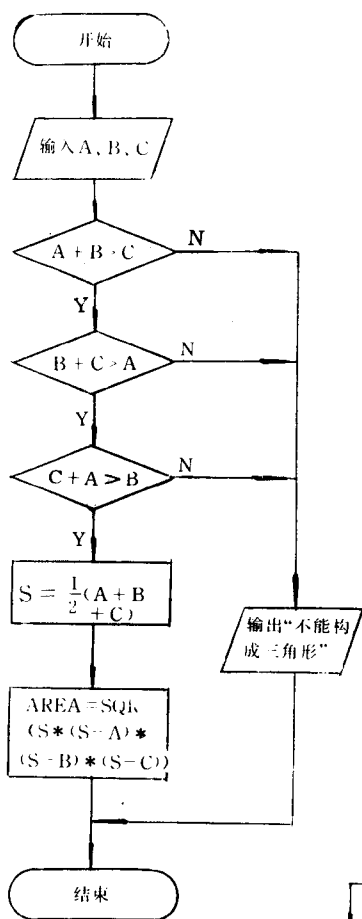


图 1.2.11

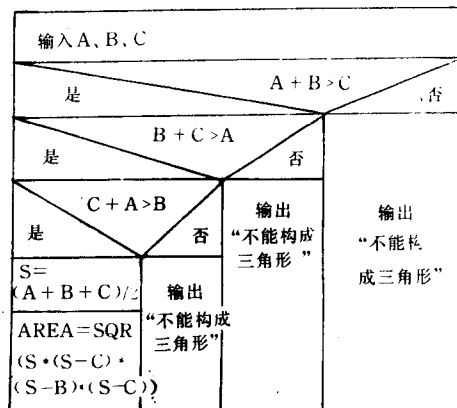


图 1.2.10

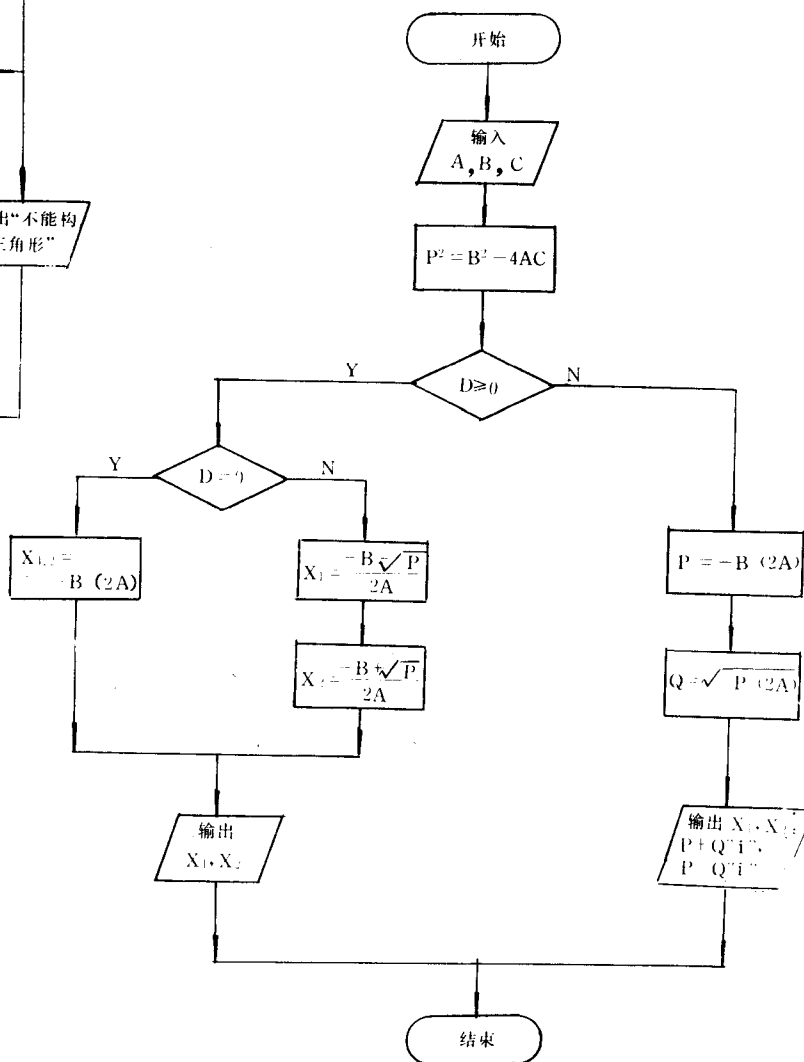


图 1.2.12