

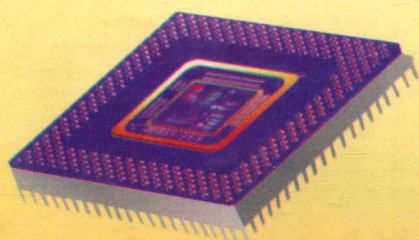
计算机等级考试系列教程（之二）

考试·教学·自学相结合的实用教材

BASIC语言程序设计·上机操作

基础教程

彭小宁 许 远 编著



电子工业出版社
PRESS OF UNIVERSITY OF ELECTRONIC
SCIENCE AND TECHNOLOGY OF CHINA

教育部高等学校计算机专业教学指导委员会 立项项目
教育部、工业和信息化部 高等学校计算机专业教学指导委员会 立项项目

BASIC语言程序设计·上机操作

基础教程

（第2版）

清华大学出版社

112374

计算机等级考试系列教程（之二）

考试、教学、自学相结合的实用教材

一级 适用

BASIC 语言程序设计·上机操作 基础教程

许宣伟

彭小宁

延红

许 远

策划

编著

审校

电子科技大学出版社

内 容 提 要

本书是计算机等级考试系列教程的第二册,全书以通俗、浅显的文字介绍了 Quick BASIC 的基础知识和程序设计技术。全书突出基本技能的培养,适合于参加各类计算机等级考试的读者自学使用,亦可作为计算机基础教育的人门教材。本书分为四大部分:第一部分(1,2章)讲解了程序设计语言的基础知识和程序设计中常用到的流程图以及 BASIC 语言的发展历史、Quick BASIC 的基本概念、Quick BASIC 的数据、表达式与赋值语句;第二部分(3,4,5,6章)讲解了 Quick BASIC 语言中最基础的编程方法,包括输入输出控制、分支结构程序设计、循环结构程序设计、子程序和函数过程;第三部分(7,8,9,10章)讲解了 Quick BASIC 的数组、字符串、文件、图形等内容;第四部分(11,12章)讲解了 Quick BASIC 的集成环境、Quick BASIC 程序的调试。本书附录还收录了 Quick BASIC 的全部关键字和出错信息。

声 明

本书无四川省版权防盗标识,不得销售;版权所有,违者必究,举报有奖,举报电话:(028) 6636481 6241146 3201496

计算机等级考试系列教程(之二)

BASIC 语言程序设计·上机操作基础教程

彭小宁 许 远 编著

延 红 审校

出 版:电子科技大学出版社 (成都建设北路二段四号,邮编:610054)

责任编辑:许宣伟

发 行:新华书店经销

印 刷:成都福利东方彩印厂印刷

开 本:787×1092 1/16 印张 24.75 字数 600 千字

版 次:1998 年 4 月第一版

印 次:1998 年 4 月第一次印刷

书 号:ISBN 7-81043-838-7/TP·353

印 数:1—4000 册

定 价:26.80 元

出版说明

随着科学技术的迅猛发展,计算机已成为各个学科领域不可缺少的应用工具,计算机知识和应用能力已成为当代大学生知识和能力结构的一个重要组成部分,也是我国教育培养跨世纪人才最突出的需要加强的环节之一。目前在高校中普遍开展的计算机知识和应用能力等级考试正有效地推动这一目标的实现。1994年12月国家教委考试中心颁布的在全国进行计算机应用能力认证考试文件,将进一步推动全社会学习计算机、使用计算机的热潮。与此有关的教材和参考资料的需求与日俱增。

我们和电子科技大学出版社合作推出的《计算机等级考试通用辅导教材》系列丛书的前七册业已顺利出版,这七册分别是:《DOS操作、文字处理及问题解答》、《BASIC语言基础知识及问题解答》、《汉字dBASEⅢ基础知识及问题解答》、《C语言基础知识及问题解答》、《PASCAL语言基础知识及问题解答》、《汇编语言基础知识及问题解答》、《FoxBASE+基础知识及问题解答》。毋庸置疑,这套丛书收到了预期的效果,对计算机知识的普及尽了自己的绵薄之力,读者来信亦充分显示,这套丛书采用的编排体例是合乎读者自学需要的。

然而,计算机科学在不断地发展,计算机等级考试也以更加猛烈之势席卷神州大地,全国范围内的各系统、行业、组织的等级考试已有十余种之多,8086/8088汇编语言等课程也列入了考试范围,而数据库技术的发展也导致了FoxBASE与FoxPro列入等级考试的范畴。熟练地掌握计算机硬件和计算机软件的理论也成为等级考试的较高要求科目。我们深知,跟上时代前进的步伐,保持这套丛书的全面性、时效性,是刻不容缓的。

没有必要掩饰本套丛书的不足与谬误之处,读者来信以及专家的评审都诚恳地指出了这一点,因此,在续编本套丛书的过程中,我们也对已出版的几种书中存在的不足之处进行了必要的更正。

为此,《计算机等级考试通用辅导教材》一书的主编约请国内部分高等学校从事计算机等级考试教学第一线工作的教师和一些对计算机普及教育有经验的同仁,共同对原书进行了系统的修改,并根据新的国家大纲充实了原书内容。考虑到本套丛书经过广泛试用,已经成熟,故第二版时将丛书名改为《计算机等级考试系列教程》并作为正式的大专教材向大专院校推广。本丛书有让人耳目一新的感觉,它浅显易懂、循序渐进、深入浅出。每章大都提供习题,特别适合初学者又是自学为主的读者之学习要求。全书在培养读者上机操作能力方面的指导意义较为突出。参与本书修订工作的教师来自电子科技大学、复旦大学等大专院校。中国电子学会的计算机专家对本书的修订再版暨续编的工作给予了极大的帮助,虽然编委会作了大量细致的工作,但肯定还有不少谬误之处,欢迎广大读者多提意见,以利再版更正。

编者

1997年9月于
电子科技大学

目 录

第 1 章 程序设计基础与 Quick BASIC 概述	(1)
1.1 程序与语言	(1)
1.1.1 程序	(1)
1.1.2 计算机语言	(1)
1.1.3 程序设计的基本步骤与任务	(2)
1.2 算法与流程图	(3)
1.2.1 算法	(3)
1.2.2 流程图	(3)
1.3 语言的识别与程序的执行	(4)
1.3.1 计算机最终能执行的是机器语言程序	(4)
1.3.2 BASIC 语言是一种高级语言	(5)
1.3.3 翻译程序	(5)
1.3.4 翻译程序的分类	(6)
1.4 BASIC 语言的发展历程	(7)
1.4.1 BASIC 语言产生的背景	(7)
1.4.2 本书为什么要以 Quick BASIC 为蓝本	(9)
1.5 Quick BASIC 程序结构	(9)
1.5.1 简单的 Quick BASIC 程序	(9)
1.5.2 Quick BASIC 的子程序结构	(11)
1.6 Quick BASIC 中的基本概念	(12)
1.6.1 程序行	(12)
1.6.2 字符集	(13)
1.6.3 Quick BASIC 的关键字	(13)
1.6.4 数据类型	(14)
1.6.5 基本数据类型	(14)
1.6.6 用户自定义的数据类型	(15)
第 2 章 数据与表达式	(16)
2.1 常 量	(16)
2.1.1 字符串常量	(16)
2.1.2 数值常量	(16)
2.1.3 Quick BASIC 的数域范围	(18)
2.1.4 符号常量	(20)
2.2 变 量	(21)
2.2.1 变量名和变量的值	(22)

2.2.2	变量的类型及其说明	(22)
2.3	运算符与表达式	(25)
2.3.1	Quick BASIC 语言中的运算符	(25)
2.3.2	巧用标准函数	(29)
2.3.3	Quick BASIC 语言中的表达式	(29)
2.3.4	不同类型数据的混合运算	(31)
2.4	赋值语句	(31)
2.4.1	赋值语句的语法	(31)
2.4.2	赋值语句的作用	(32)
2.4.3	对变量赋值操作的注意事项	(33)
2.5	其他形式的赋值语句	(34)
2.5.1	SWAP 语句	(34)
2.5.2	访问内存的语句和函数	(35)
2.6	注释、暂停与结束程序	(35)
2.6.1	注释语句	(35)
2.6.2	暂停语句	(36)
2.6.3	程序结束语句	(36)
2.7	习 题	(36)
第3章	输入输出控制	(38)
3.1	输入语句	(38)
3.1.1	问题的提出	(38)
3.1.2	键盘输入语句 (INPUT)	(39)
3.1.3	INPUT 语句使用注意事项	(42)
3.1.4	行输入语句	(43)
3.2	读数与置数	(43)
3.2.1	读数/置数 (READ/DATA) 语句	(43)
3.2.2	Quick BASIC 中的特殊规定	(45)
3.3	恢复数据区	(45)
3.3.1	问题的提出	(45)
3.3.2	恢复读数据语句	(46)
3.4	最基本的输出语句——PRINT	(47)
3.4.1	输出的意义	(47)
3.4.2	PRINT 语句的格式	(47)
3.4.3	PRINT 语句的初步使用	(48)
3.4.4	输出格式的控制	(50)
3.4.5	使用 PRINT 语句时容易出现的错误分析	(54)
3.4.6	实数的输出	(55)
3.4.7	应用举例	(55)
3.5	与 PRINT 语句有关的函数	(56)
3.5.1	TAB 函数	(56)
3.5.2	SPACE\$ (n) 函数	(61)
3.6	自选格式输出语句	(61)

3.6.1	PRINT USING 语句	(61)
3.6.2	PRINT USING 语句使用示范	(62)
3.7	其他输出语句	(66)
3.7.1	WRITE 语句	(66)
3.7.2	LPRINT 和 LPRINT USING 语句	(66)
3.8	特殊输入操作	(67)
3.8.1	INKEY\$ 函数	(67)
3.8.2	INPUT\$ 函数	(67)
3.8.3	KEY 语句	(68)
3.9	文本光标控制	(68)
3.9.1	光标定位 (LOCATE)	(68)
3.9.2	定义光标大小	(69)
3.9.3	检查光标位置	(70)
3.10	顺序结构程序举例	(70)
3.11	习 题	(72)
第 4 章	流程控制 (I) —— 分支结构程序设计	(74)
4.1	无条件转移语句	(74)
4.1.1	问题的引入	(74)
4.1.2	GOTO 语句应用实例	(76)
4.1.3	使用无条件转移语句 GOTO 的注意事项	(77)
4.2	开关转向语句	(77)
4.2.1	多分支转向语句 ON~GOTO	(77)
4.2.2	ON~GOTO 语句应用实例	(79)
4.3	关系运算与逻辑运算	(80)
4.3.1	关系运算	(80)
4.3.2	关系运算中的注意事项	(82)
4.3.3	基本逻辑运算	(83)
4.3.4	为什么不能连续赋值	(85)
4.4	条件语句	(86)
4.4.1	条件转移语句	(86)
4.4.2	条件执行语句	(87)
4.4.3	应用举例	(88)
4.4.4	IF 语句多重嵌套的问题	(90)
4.4.5	块 IF 结构的一般格式	(94)
4.4.6	块 IF 的应用举例	(95)
4.4.7	块 IF 的嵌套	(97)
4.4.8	在块 IF 中使用 ELSEIF	(98)
4.5	情况语句	(103)
4.5.1	最基本的 SELECT CASE 结构	(104)
4.5.2	在 CASE 子句中使用“TO”指定值的范围	(106)
4.5.3	在 CASE 子句中使用“IS”指定条件	(107)
4.5.4	在 CASE 子句中使用多个条件	(108)

4.5.5	使用 SELECT CASE 结构的注意事项	(108)
4.6	分支结构嵌套	(113)
4.6.1	单行 IF 语句的嵌套	(114)
4.6.2	块 IF~THEN~ELSE 语句的嵌套	(114)
4.6.3	SELECT CASE 语句的嵌套	(115)
4.7	程序举例	(115)
4.8	习 题	(119)
第 5 章	流程控制 (II) —— 循环结构	(121)
5.1	概 述	(121)
5.2	FOR~NEXT 循环	(122)
5.2.1	FOR~NEXT 语句的语法	(122)
5.2.2	FOR~NEXT 语句疑难解答	(126)
5.2.3	EXIT FOR 语句	(130)
5.2.4	FOR 语句嵌套	(130)
5.3	WHILE~WEND 循环	(134)
5.3.1	WHILE~WEND 语句的语法	(134)
5.3.2	WHILE 循环结构的嵌套	(135)
5.4	DO~LOOP 循环	(135)
5.4.1	DO~LOOP 语句的语法	(135)
5.4.2	最简单的 DO~LOOP 语句	(137)
5.4.3	用 EXIT DO 语句终止循环	(137)
5.4.4	带 WHILE 子句的 DO 循环	(138)
5.4.5	带 UNTIL 子句的 DO 循环	(139)
5.5	多重循环	(144)
5.6	用循环语句解决复杂数学问题	(145)
5.7	习 题	(147)
第 6 章	流程控制 (III) —— 子程序与函数	(149)
6.1	概 述	(149)
6.1.1	子程序的概念	(149)
6.1.2	为什么要引入子程序	(149)
6.1.3	子程序概念的发展	(150)
6.2	用 GOSUB 语句调用子程序	(151)
6.2.1	GOSUB~RETURN 语句的语法	(151)
6.2.2	调用 GOSUB 子程序的三种方法	(155)
6.2.3	ON GOSUB~RETURN 语句	(155)
6.2.4	ON~GOSUB 开关转子语句应用举例	(156)
6.2.5	ON KEY (n) GOSUB~RETURN 语句	(157)
6.3	独立模块子程序	(159)
6.3.1	SUB 过程的定义	(159)
6.3.2	SUB 过程的调用	(160)
6.4	随机函数	(162)

6.4.1	随机函数的意义	(162)
6.4.2	随机化语句	(163)
6.4.3	随机函数的应用	(165)
6.5	自定义函数	(166)
6.5.1	单行形式的自定义函数的概念与格式	(166)
6.5.2	关于单行形式的自定义函数的定义和使用说明	(166)
6.5.3	块形式的自定义函数	(170)
6.5.4	独立模块的自定义函数	(171)
6.5.5	STATIC 选项	(175)
6.5.6	程序举例	(176)
6.6	自变量的传递	(177)
6.6.1	自变量的传递形式	(177)
6.6.2	常量和表达式的传递	(177)
6.6.3	变量的传递	(178)
6.7	过程的说明	(181)
6.7.1	DECLARE 语句	(181)
6.7.2	蕴含文件使用说明	(183)
6.8	局部变量与全局变量	(184)
6.8.1	局部变量	(184)
6.8.2	全局变量	(185)
6.9	共享变量	(186)
6.9.1	与指定过程共享变量	(186)
6.9.2	同一模块中的所有过程共享变量	(189)
6.9.3	与其他模块共享变量	(189)
6.10	STATIC 语句与 STATIC 变量	(190)
6.10.1	STATIC 语句	(190)
6.10.2	STATIC 变量	(191)
6.11	嵌套与递归	(191)
6.11.1	GOSUB 子程序嵌套	(192)
6.11.2	独立模块子程序的嵌套调用	(194)
6.11.3	递归	(194)
6.11.4	CLEAR 语句	(196)
6.12	SUB 过程与 GOSUB 子程序的比较	(196)
6.13	程序的运行控制	(197)
6.13.1	控制始终在主模块的程序运行	(197)
6.13.2	控制在各模块间的转换的程序运行	(197)
6.14	程序举例	(199)
6.15	习 题	(203)
第7章	数组——有序数据的处理	(205)
7.1	概 述	(205)
7.1.1	有序数据处理的一个例子	(205)
7.1.2	下标变量与一维数组	(205)

7.1.3	应用举例	(207)
7.2	数组的定义	(208)
7.2.1	定义数组语句 DIM	(208)
7.2.2	第一种常用格式	(208)
7.2.3	改变数组的下界	(209)
7.2.4	DIM 语句的其他格式	(209)
7.2.5	数组的引用	(210)
7.2.6	使用 DIM 语句时的注意事项	(212)
7.2.7	DIM 说明数组语句的完整形式	(212)
7.3	二维数组	(214)
7.3.1	数据表格处理引例	(214)
7.3.2	双下标变量与二维数组	(215)
7.3.3	应用举例	(216)
7.4	数组的基本操作	(218)
7.4.1	数组元素的输入	(219)
7.4.2	数组元素的输出	(219)
7.4.3	数组元素的复制	(220)
7.5	排序与矩阵运算	(220)
7.5.1	最简单的排序方法	(220)
7.5.2	直接插入排序	(221)
7.5.3	起泡排序法	(223)
7.5.4	数组在数学上的应用——矩阵运算	(225)
7.6	静态数组与动态数组	(228)
7.6.1	数组删除语句	(228)
7.6.2	数组重定义语句	(229)
7.7	过程调用中数组自变量的传递	(230)
7.8	程序举例	(232)
7.9	习 题	(236)
第 8 章	文件系统	(238)
8.1	文件的基本概念	(238)
8.1.1	文件的分类	(238)
8.1.2	文件与记录	(239)
8.1.3	文件名	(240)
8.1.4	文件的读写和文件缓冲区	(241)
8.1.5	文件指针	(241)
8.2	顺序文件	(241)
8.2.1	什么是顺序文件	(241)
8.2.2	顺序文件的打开与关闭	(242)
8.2.3	顺序文件的写操作	(243)
8.2.4	顺序文件的读操作	(245)
8.2.5	顺序文件的维护	(247)
8.3	随机文件	(252)

8.3.1	随机文件的概念	(252)
8.3.2	随机文件的建立	(252)
8.3.3	从随机文件读入数据	(255)
8.4	记录类型变量	(259)
8.4.1	随机文件的记录定义和记录长度计算	(259)
8.4.2	记录变量	(260)
8.4.3	用于记录变量的读写语句	(262)
8.4.4	记录变量应用举例	(263)
8.5	二进制文件	(265)
8.5.1	二进制文件的特点	(265)
8.5.2	二进制文件的读写语句	(265)
8.6	文件操作的总结	(266)
8.6.1	文件的打开和关闭	(266)
8.6.2	文件的打开 (OPEN 语句)	(266)
8.6.3	文件的关闭 (CLOSE 语句)	(269)
8.6.4	文件操作语句和函数	(270)
8.6.5	随机文件用到的语句和函数	(275)
8.7	Quick BASIC 处理文件与目录的语句	(275)
8.8	程序举例	(277)
8.9	习 题	(279)
第9章 字符串处理		(281)
9.1	概 述	(281)
9.2	字符串常量	(282)
9.2.1	无名字符串常量	(282)
9.2.2	符号字符串常量	(282)
9.3	字符串变量	(282)
9.3.1	变长字符串变量	(283)
9.3.2	定长字符串变量	(284)
9.4	字符串变量的赋值	(284)
9.4.1	用 LET 语句给字符串变量赋值	(285)
9.4.2	用 INPUT 语句给字符串变量赋值	(285)
9.4.3	用 LINE INPUT 语句给字符串变量赋值	(286)
9.4.4	用 READ/DATA 语句给字符串变量赋值	(286)
9.5	字符串的运算	(287)
9.5.1	字符串连接运算	(287)
9.5.2	字符串的比较运算	(288)
9.5.3	字符关系表达式	(288)
9.5.4	举例	(289)
9.6	字符串数组	(290)
9.7	子字符串	(291)
9.7.1	LEFT\$ 函数	(291)
9.7.2	RIGHT\$ 函数	(292)

9.7.3	MID\$ 函数	(292)
9.7.4	MID\$ 语句	(294)
9.7.5	INSTR 函数	(295)
9.7.6	删除字符串首尾空格的函数	(296)
9.8	有关字符串的函数	(296)
9.8.1	测字符串长度的函数 (LEN)	(296)
9.8.2	字符串与数值间的转换函数	(297)
9.8.3	字符与 ASCII 码间的转换函数	(298)
9.8.4	大小写字母之间的转换	(300)
9.8.5	建立由相同字符组成的字符串函数	(301)
9.8.6	日期和时间函数	(302)
9.8.7	INKEY\$ 函数	(303)
9.9	字符串处理程序举例	(303)
9.10	习 题	(305)
第 10 章	图形程序设计	(306)
10.1	屏幕显示方式参数设置	(306)
10.2	参数设置语句与函数	(307)
10.2.1	SCREEN 函数和 SCREEN 语句	(307)
10.2.2	视见区与窗口语句	(309)
10.2.3	清屏与改变行宽语句	(314)
10.2.4	屏幕颜色设置语句 (COLOR)	(315)
10.2.5	屏幕颜色点测试函数和屏幕坐标测试函数 (POINT 函数)	(316)
10.3	基本绘图语句	(316)
10.3.1	画点语句 (PSET 和 PRESET)	(316)
10.3.2	画线语句 (LINE)	(317)
10.3.3	连续画线语句 (DRAW)	(318)
10.3.4	画圆和弧语句 (CIRCLE)	(319)
10.3.5	颜色填充语句 (PAINT)	(320)
10.4	动画技术	(322)
10.4.1	字符动画	(322)
10.4.2	图形动画	(323)
10.4.3	快速动画	(325)
10.5	函数图形的显示	(329)
10.5.1	Quick BASIC 中绘制函数图形	(329)
10.5.2	Quick BASIC 绘图的特点	(334)
第 11 章	Quick BASIC 系统简介	(335)
11.1	Quick BASIC 的解释方式与编译方式	(335)
11.1.1	DOS 下的解释 Quick BASIC (又称 QBASIC)	(335)
11.1.2	编译的 Quick BASIC	(336)
11.1.3	Quick BASIC 的特点	(341)
11.1.4	Quick BASIC 的启动命令行	(342)

11.2 Quick BASIC 的窗口与菜单	(343)
11.2.1 窗口	(343)
11.2.2 光标	(344)
11.2.3 行和列的位置显示	(344)
11.2.4 鼠标指针	(344)
11.2.5 菜单条和菜单名称	(345)
11.2.6 参考条	(345)
11.2.7 窗口的缩放控制	(345)
11.3 编辑和运行 Quick BASIC 程序	(345)
11.3.1 从键盘输入 Quick BASIC 源程序	(345)
11.3.2 运行 Quick BASIC 程序	(346)
11.3.3 修改和编辑源程序	(348)
11.4 编写和运行程序一览	(353)
11.4.1 编程注意事项	(353)
11.4.2 向计算机输入一个新程序	(354)
11.4.3 将程序存盘	(355)
11.4.4 打开已存盘的文件	(356)
11.4.5 退出 Quick BASIC	(356)
第 12 章 建立子程序与程序调试	(358)
12.1 建立一个子程序	(358)
12.1.1 在 Quick BASIC 环境中有关子程序的操作	(360)
12.1.2 在屏幕上同时观察两个模块	(361)
12.2 立即执行方式	(362)
12.3 分步执行与设置断点	(363)
12.3.1 Quick BASIC 的功能键提示行	(363)
12.3.2 分步执行	(364)
12.3.3 设置断点	(364)
附录	(365)
附录 A ASCII 字符代码	(365)
附录 B Quick BASIC 保留字	(366)
附录 C Quick BASIC 语句一览表	(367)
附录 D Quick BASIC 键盘命令一览表	(370)
附录 E Quick BASIC 内部函数一览表	(371)
附录 F 启动、编译和运行时的出错信息	(373)
附录 G 连接时错误信息	(377)
附录 H LIB 错误信息	(379)

第 1 章 程序设计基础与 Quick BASIC 概述

本章学习要点

- 程序的概念
- 什么是计算机语言
- 流程图及其应用
- 程序设计的步骤
- 算法的概念
- Quick BASIC 的主要特点
- Quick BASIC 中的若干基本概念
- Quick BASIC 的启动与退出

1.1 程序与语言

1.1.1 程序

计算机是一个能高速地运算,具有存储与记忆能力的由程序控制的电子装置。由于它是一种电子装置,又称为电子计算机,其工作原理基于电子脉冲电路,由电路产生脉冲,依靠脉冲信号进行数据传递与运算等操作。

但是,这些操作必须在人们的控制下完成,它无法完全自动地进行工作。只有当操作者向计算机输入一定的信息(这种信息必须是计算机能够接受的),它才能按照操作者的要求进行工作,并且得出所需的结果。那么,人们是怎样通过输入的信息来让计算机工作的呢?这就涉及到计算机的工作机制。目前的计算机的工作机制基本上是这样的:

(1)将需要计算机完成的任务编成一条一条的指令,输入计算机,存放在计算机的内存存储器中(称为“编程序”);

(2)计算机在工作时逐条的从内存中取出指令,然后再执行(称为“运行程序”)。

可见计算机是通过一条一条的指令来完成一定的工作的。因此,人们要控制计算机也是通过一条一条的指令来进行的。用计算机术语来说,指令的序列被称为程序,计算机就在人们编制的程序控制下工作。程序是人们意志的体现,它表明了要计算机“做什么”和“怎么做”。同时,程序又是计算机处理问题的灵魂,只有当操作者向计算机输入一定的程序,计算机才能按照程序中规定的步骤工作。失去了程序的控制,计算机便无法发挥其作用,变成一堆废铜烂铁。

1.1.2 计算机语言

程序是计算机指令的集合。我们做一件事时,总要按照一定的步骤一步一步地进行。计算机也是一样的,执行一个程序时总是一条指令一条指令地执行。例如,要求三个数 A,B,C

的平均数 M , 就可以按照以下的“指令”来进行:

- (1) 输入 A ;
- (2) 输入 B ;
- (3) 输入 C ;
- (4) 求和 $S=A+B+C$;
- (5) 求平均数 $M=S/3$;
- (6) 打印 M 的值。

那么, 怎样把以上(1)~(6)所确定的步骤以一种计算机能够接受的形式输入计算机呢? 这就必须克服以下问题:

首先, 计算机并不认识汉字或者其他的自然语言, 如英语、法语等。若强行把以上(1)~(6)以某种自然语言形式输入计算机, 计算机也无法“执行”你的“程序”。

其次, 自然语言的意义往往和具体的语言环境有关。一个句子, 一个词语在不同的场合的含义往往不同, 具有“二义性”, 而计算机不能容忍这种“二义性”。因为它根本不具备智能, 只能机械地、古板地按照你的规定步骤去做, 如果你的程序中存在“二义性”, 计算机将不知所措。

正是因为以上的原因, 我们期待着一种能够准确无误地表达我们编制的程序含义的、一种能够被计算机和人们所接受的、一种相当严谨的, 不具有“二义性”的表达方法的出现。这种表达方法正是我们所说的“计算机语言”。

“计算机语言”是人与计算机之间进行通信的工具, 是一种计算机能够接受的信息, 它由一些简单的单词符号、数字和严谨的语法组成, 能够准确无误地表达程序的含义。它是专门用于人们与计算机之间信息交流的一种特殊语言, 在人与计算机之间建立起了一座信息的桥梁。目前计算机语言的种类很多。总的来说, 可以分成机器语言、汇编语言和高级语言三大类。

1.1.3 程序设计的基本步骤与任务

人们通过程序让计算机工作起来, 以便处理各种信息, 解决各种问题, 让计算机为人类服务, 所以“程序”是人们意志的体现; 反过来说, 要让计算机按自己的“意志”进行运算, 就必须编制程序。编制程序的任务在于对用户提供的输入信息进行加工、处理并输出预期的结果。编制程序是本书的核心内容, 以下将简要地说明编制程序的过程。

(1) 分析问题 这是编程序的第一步, 因为任何程序都是为了解决一定的实际问题, 编程序时不能无的放矢, 而要认真考察实际问题, 找出解决问题的大致思路。

(2) 提出算法 把第一步中的解决问题的思路进一步明确化、详细化, 建立解题需要的数学或物理模型。也就是把解题步骤一步一步详细地写出来, 为下一步用计算机语言来表达这些方法奠定基础。

(3) 编写程序 根据第二步的方案用一定的计算机语言把程序写出来。

(4) 上机调试 对编好的程序进行实际检验, 发现其中的错误之处, 不断加以改正, 直到程序能达到预期目的。

(5) 运行程序 将程序投入运行, 并输出结果。

1.2 算法与流程图

在系统地学习 BASIC 语言程序设计知识之前,先介绍一种进行程序设计的工具。

1.2.1 算法

算法就是对问题求解方法的精确描述。在进行程序设计时,最关键的问题是算法的提出。因为它直接关系到你写出来的程序的正确性、可靠性。如果没有认真地研究实际的问题,就草率地提出一些不成熟的算法,那么编写出来的程序就可能出现错误或疏忽。算法作为对解题步骤的精确描述,应具备如下性质:

1. 有穷性

一个算法必须在有限步骤之后结束,而不能无限制地进行下去。因此,在算法中必须给出一个结束的条件。

我们不能指望计算机算出圆周率的精确值,因为 π 是一个无穷不循环小数,无法求出它的精确值。同理,我们也不要让计算机计算诸如 $\lim_{n \rightarrow \infty} \frac{n^2+1}{n^2-1}$ 的精确值,因为计算机根本无法使“ $n \rightarrow \infty$ ”这个条件成立。

2. 明确性

一个算法中的任何步骤都必须意义明确,不能模棱两可、含混不清,即不允许有“二义性”,不能在计算机中使用诸如“老张对老李说:他的儿子考上了清华大学”这类有歧义的表达方法,到底是老张的儿子上了大学还是老李的儿子上了大学? 无法确知。

3. 可执行性

所采用的算法必须能够在计算机上执行,因此,在算法中所有的运算必须是计算机能够执行的基本运算。要计算机执行的步骤,计算机应该能够实现,不能提出像“让计算机去煮饭,煮完饭之后再炒菜”之类的算法,至少它在目前无法实现。

4. 有一定的输入与输出

要计算机解决问题时,总是需要输入一些原始的数据;计算机向用户报告结果时,总是要输出一些信息。因此,一个算法中必须有一定的输入与输出。

以上是算法的基本性质,在此基础上就可以学习描述算法的基本工具——流程图。

1.2.2 流程图

流程是一种能够比较形象地描述“算法”的工具,它对于编制程序很有帮助。流程图又称为框图,是由几种不同的图形组合而成的,如图 1-1 所示。

①起止框,如图 1-1(a),它代表一个算法的开始与结束之处。

②处理框,如图 1-1(b),它可以表示算法中的一个或若干个步骤,这些步骤不涉及输入与输出。

③输入输出框,如图 1-1(c),它表示一个算法中需要进行输入或输出处理的步骤。为了与一般的处理步骤区别开来,输入输出框采用了平行四边形的形式。

④条件判断框,如图 1-1(d),当一个算法中需要依据某一条件来决定后续操作时,采用