

■ 中等职业技术学校计算机专业系列教材



QBASIC

语言程序设计

QBASIC YUYAN CHENGXU SHEJI

胡 泉 编

华中师范大学出版社

中等职业技术学校计算机专业系列教材

胡金柱 杨发明 主编

QBASIC 语言程序设计

胡 泉 编

华中师范大学出版社

内 容 提 要

程序设计是计算机专业学生和应用人员的一门必修课程。学习程序设计,必须选择一种计算机语言。QBASIC 是 DOS 操作系统自带的一门计算机程序设计语言,是在 BASIC 的基础之上发展起来的。它具有容易学习,操作简单,结构化程度高等特点,是广大初学者学习程序设计的首选高级语言。本书从介绍 QBASIC 程序设计语言的特点和使用环境开始,逐步介绍了 QBASIC 语言的基本语句和结构,以及结构化程序设计的基本思想和方法,并在此基础上介绍了如何使用函数和子程序实现结构化程序设计。最后还简单介绍了 QBASIC 语言的图形绘制和数据文件。

本书适用于初学计算机程序设计语言的各类读者。

(鄂)新登字 11 号

图书在版编目(CIP)数据

QBASIC 语言程序设计/胡泉 编.

—武汉:华中师范大学出版社,2001.9

中等职业技术学校计算机专业系列教材

ISBN 7-5622-2465-X/TP·29

I. Q… II. 胡… III. BASIC 语言-程序设计-专业学校-教材 IV. TP 312

中国版本图书馆 CIP 数据核字(2001)第 059912 号

中等职业技术学校计算机专业系列教材

QBASIC 语言程序设计

© 胡 泉 编

华中师范大学出版社出版发行

(武汉市武昌桂子山 邮编:430079 电话:027-87876240)

新华书店湖北发行所经销

京山县印刷厂印刷

责任编辑:勉 之

封面设计:罗明波

责任校对:罗 艺

督 印:姜勇华

开本:787mm×1092mm 1/16

印张:11 字数:282 千字

版次:2001 年 9 月第 1 版

2001 年 9 月第 1 次印刷

印数:1—5 000

定价:12.00 元

本书如有印装质量问题,可向承印厂调换。

前 言

21 世纪是知识经济时代,知识经济的核心是计算机科学与应用,计算机应用的核心之一是程序和程序设计。程序设计又是计算机专业的学生和应用人员的必修课程和基本训练。QBASIC 是 DOS 操作系统自带的一门计算机程序设计语言,是在 BASIC 的基础之上发展起来的。QBASIC 具有容易学习,操作简单,结构化程度高等特点,是广大初学者学习程序设计的首选高级语言。再加上学习了 QBASIC 容易扩展到现在流行的程序设计工具 Visual Basic,所以深受广大初学者的喜爱。

本书根据初学者的特点共分九章,其中第一章首先介绍了程序设计的基本步骤、什么是算法、什么是流程图以及如何画流程图等问题;然后简单介绍了 BASIC 语言的发展过程和 QBASIC 语言的特点;最后较详细地介绍了 QBASIC 程序的组成和 QBASIC 的编程环境。

第二章主要介绍了组成 QBASIC 程序的一些基本成分,以及使用这些成分进行最简单的程序设计方法。这些基本成分包括:QBASIC 的数据类型、常量和变量、内部函数、算术运算符和算术表达式、数据传送语句(如赋值语句、输入输出语句等)、程序结束与暂停语句等。

第三章、第四章、第五章以及第六章主要介绍了程序设计的基础。第三章介绍了关系运算符和关系表达式、逻辑运算符和逻辑表达式,还介绍了用以编写选择结构程序的行 IF 语句和块 IF 语句,以及 SELECT CASE 语句和 ON GOTO 语句。第四章主要介绍了 DO 循环结构、WHILE 循环结构和 FOR 循环结构及其使用方法。第五章主要介绍了结构化程序设计的一些基本概念、原则和方法,即程序的三种基本控制结构及其设计和表示方法。第六章主要介绍数组及其说明方法和使用方法。这几章是本书重点所在。

第七章则是学习程序设计的重点,目的在于教会大家如何使用 QBASIC 实现一个完整的程序。主要包括一个完整源程序的结构,以及构成一个完整源程序的各种函数和各种子程序的结构及其实现方法。

第八章简单介绍了 QBASIC 的图形处理功能,包括屏幕控制方法和图形绘制方法。学习这些内容,有利于深入学习 QBASIC 的强大图形处理功能。

第九章则介绍了文件的概念以及有关文件操作的语句、函数及其使用方法。文件操作是计算机程序在实际运行过程中时常遇到的问题,学习本章内容,有利于保存输入输出数据和程序运行的结果。

本书在编写过程中,得到了华中师范大学计算机科学系的谭根穗书记、覃事太副书记和范重庆老师的积极鼓励 and 大力支持,在此表示衷心的感谢。同时,深深地感谢华中师范大学计算机科学系肖德宝教授、冯刚教授和陈利副教授、何婷婷副教授、魏开平副教授、金汉均副教授、叶俊民副教授、刘明副教授、郑世珏副教授以及喻莹等老师的支持和帮助。参加本书部分编写、整理和程序调试的还有华中农业大学的邓君丽老师以及武汉大学柳轶、华中师范大学计科系陈伟、杜超华、邱林、汪颖、张杨、陈永、蒋志龙、李亚威、杨兵、谢波、谢芳、张青锋、夏浩波及孙为。

本书所有的程序均经过精心设计和上机认真调试,但由于本人水平有限,难免出现错误,敬请广大读者指正。

编 者

2001 年 9 月于武昌桂子山
华中师范大学计算机科学系

中等职业技术学校计算机专业系列教材 编写及使用说明

为适应我国当前中等职业技术学校计算机专业教学的需要,我们组织编写了一套实用教材——中等职业技术学校计算机专业系列教材。本教材针对中专学生的层次水平(初中毕业生)及初学计算机的特点,以及计算机专业教学的需要,我们对整套书中的内容作了周密的安排,不仅体系合理,而且深入浅出、概念清晰、例题丰富、逻辑性强、文字流畅、通俗易懂。

全套书共分12册,它们分别是:

1. 微型计算机组成原理;2. 微型计算机操作;3. QBASIC 语言程序设计;4. C 语言程序设计;5. 数据结构初步;6. FOXBASE⁺ 关系数据库;7. 关系数据库 FOXPRO;8. 微型计算机系统的安装与维护;9. 计算机网络基础;10. 计算机软件开发技术;11. Windows 98 基础教程;12. 计算机辅助设计;13. 计算机专业英语。

每册教材的详细情况请见各册的前言。

本教材建议安排在3~5个学期讲授完,每学期可安排2~3门课程。教学顺序可按上述排列顺序进行,但应根据各学校学生具体情况和上机实习条件酌情安排。每本书自成体系,可独立使用。这些课程实践性较强,上机实习必不可少。每门课程的讲授与实习学时安排建议如下:

1. 《微型计算机组成原理》:可安排的70~90学时,其中讲授50~60学时,汇编语言的上机可安排20~30学时。

2. 《微型计算机操作》:本课程以上机实习为主,课堂讲授为辅。总学时可安排70~90学时,讲授30学时左右,上机实习40学时以上。

3. 《QBASIC 语言程序设计》:总学时70~90学时,课堂讲授占2/3,上机实习占1/3左右。

4. 《C 语言程序设计》:总学时70~90学时,课堂讲授占2/3,上机实习占1/3左右。

5. 《数据结构初步》:总学时70~90学时,讲授50~70学时,上机实习20学时左右。

6. 《FOXBASE⁺ 关系数据库》:总学时70~90学时,课堂讲授占2/3,上机实习占1/3。

7. 《关系数据库 FOXPRO》:总学时70~90学时左右,课堂讲授占2/3,上机实习占1/3。

8. 《微型计算机系统的安装与维护》:总学时80~100学时,课堂讲授占总学时的3/4,实习可占总学时的1/4(有条件的尽可能安排实习,没有条件的可演示)。

9. 《计算机网络基础》:总学时70~90学时左右,讲授占3/4,实习占1/4,没有实习条件的可安排参观、演示。

10. 《计算机软件开发技术》:总学时70学时左右,讲授50学时左右,寻找小课题按软件开发期进行实习,学时可酌情安排。

11. 《Windows 98 基础教程》:总学时70~90学时,讲授与上机各占一半。

12. 《计算机辅助设计》:总学时70学时左右。

13. 《计算机专业英语》:总学时80学时。

另外:①书中凡打星号(*)的章节可作选讲内容,不作要求;②FOXBASE⁺与FOXPRO的许多命令相同,这两门课程可酌情选其中一种开设。对于学生来说,学会其中一种,另一种则很容易掌握(另一种可作为学生自学的教材);③五笔字型与自然码不要求都学,可选其中一种教学;④有条件的学校应尽量开设Windows 98和网络课程,没有上机实习条件的学校可暂时不开。

中等职业技术学校计算机专业系列教材编写组

2001年9月

目 录

第一章 QBASIC 基础知识	(1)
§ 1.1 算法与流程图简介	(1)
1.1.1 程序设计的基本步骤	(1)
1.1.2 算法与流程图	(2)
§ 1.2 BASIC 语言发展简介	(3)
1.2.1 BASIC 语言的发展简史	(3)
1.2.2 QBASIC 的特点	(3)
§ 1.3 QBASIC 的组成及其使用环境	(4)
1.3.1 QBASIC 程序的组成	(4)
1.3.2 QBASIC 的使用环境	(5)
§ 1.4 QBASIC 程序的编辑与使用过程	(8)
1.4.1 输入 QBASIC 程序	(8)
1.4.2 修改 QBASIC 程序	(8)
1.4.3 保存 QBASIC 程序	(10)
1.4.4 运行 QBASIC 程序	(10)
1.4.5 退出 QBASIC	(10)
习 题 一	(10)
第二章 最简单的程序设计	(12)
§ 2.1 QBASIC 的基本数据	(12)
2.1.1 数据类型	(12)
2.1.2 常量	(13)
2.1.3 变量	(15)
§ 2.2 QBASIC 的运算符与表达式	(17)
2.2.1 QBASIC 语言中的运算符	(17)
2.2.2 内部函数	(17)
2.2.3 QBASIC 语言中的算术表达式	(18)
2.2.4 不同类型数据之间的混合运算	(19)
§ 2.3 输出语句(PRINT)	(19)
2.3.1 PRINT 语句的一般格式	(19)
2.3.2 PRINT 语句的作用	(20)
2.3.3 用于 PRINT 语句格式控制的有关函数	(22)
§ 2.4 提供数据的语句	(24)
2.4.1 赋值(LET)语句	(24)
2.4.2 键盘输入语句(INPUT)	(25)
2.4.3 读数(READ)语句与置数(DATA)语句	(28)
2.4.4 恢复数据语句(RESOTRE)	(29)
§ 2.5 其它语句	(30)

2.5.1	注释语句(REM)	(30)
2.5.2	结束语句(END)	(31)
2.5.3	暂停语句(STOP 和 SLEEP)	(31)
2.5.4	无条件转向语句(GOTO)	(32)
2.5.5	清屏语句(CLS)	(32)
2.5.6	定位光标语句(LOCATE)	(32)
2.5.7	SWAP 语句	(32)
§ 2.6	字符串	(33)
2.6.1	字符串变量	(33)
2.6.2	用 LINE INPUT 语句输入字符串变量	(34)
2.6.3	字符串的连接运算	(34)
2.6.4	字符串函数	(35)
习 题 二		(37)
第三章	选择结构程序设计	(42)
§ 3.1	关系表达式与逻辑表达式	(42)
3.1.1	关系运算符和关系表达式	(42)
3.1.2	逻辑运算符与逻辑表达式	(44)
§ 3.2	IF 语句	(45)
3.2.1	行 IF 语句	(45)
3.2.2	块 IF 语句	(47)
3.2.3	块 IF 语句的嵌套	(49)
3.2.4	ELSEIF 结构	(50)
§ 3.3	多路分支选择结构	(52)
3.3.1	SELECT CASE 语句	(53)
3.3.2	ON...GOTO...语句	(55)
3.3.3	多路分支选择结构实现方法比较	(57)
习 题 三		(57)
第四章	循环程序设计	(60)
§ 4.1	DO 循环结构	(60)
4.1.1	无条件的 DO 循环	(60)
4.1.2	用 EXIT DO 语句退出循环体的 DO 循环	(61)
4.1.3	带条件的 DO 循环	(61)
§ 4.2	WHILE~WEND 循环	(65)
§ 4.3	FOR~NEXT 循环	(66)
4.3.1	FOR~NEXT 循环的基本格式	(66)
4.3.2	FOR~NEXT 循环的执行过程	(68)
§ 4.4	多重循环	(70)
4.4.1	FOR~NEXT 循环的嵌套	(70)
4.4.2	WHILE~WEND 结构的嵌套循环	(73)
4.4.3	DO~LOOP 结构的嵌套循环	(73)

4.4.4 各种循环结构的嵌套使用	(74)
习 题 四	(76)
第五章 结构程序设计基础	(82)
§ 5.1 基本概念	(82)
§ 5.2 程序的控制结构	(82)
5.2.1 引言	(82)
5.2.2 顺序结构	(84)
5.2.3 选择结构	(84)
5.2.4 循环结构	(88)
§ 5.3 结构化流程图与缩格书写风格	(93)
5.3.1 结构化流程图(N-S图)	(93)
5.3.2 缩格书写	(94)
§ 5.4 程序设计技巧与简单测试	(94)
5.4.1 评价程序质量的基本标准	(94)
5.4.2 程序设计的一些基本技巧	(95)
5.4.3 简单的测试与排错方法	(96)
§ 5.5 小结	(96)
习 题 五	(96)
第六章 数组	(98)
§ 6.1 组与数组元素的概念	(98)
§ 6.2 数组的说明与引用	(99)
6.2.1 数组说明语句 DIM	(99)
6.2.2 一维数组的使用	(102)
6.2.3 二维数组与多维数组的使用	(104)
§ 6.3 动态数组的定义与使用	(108)
§ 6.4 数组的应用	(113)
6.4.1 最简单的矩阵运算	(113)
6.4.2 简单的排序算法	(118)
6.4.3 简单的查找算法	(123)
习 题 六	(125)
第七章 子程序与函数	(128)
§ 7.1 源程序的结构	(128)
7.1.1 程序的结构	(128)
7.1.2 模块化程序设计的优点	(129)
§ 7.2 函数	(129)
7.2.1 自定义函数(DEF 函数)	(129)
7.2.2 函数过程	(132)
§ 7.3 子程序	(137)
7.3.1 块内子程序	(138)
7.3.2 子程序过程	(141)

习 题 七	(148)
第八章 屏幕控制与图形绘制	(152)
§ 8.1 屏幕显示模式	(152)
§ 8.2 有关文本模式的语句	(152)
8.2.1 定义屏幕宽度语句	(152)
8.2.2 保存当前光标所在位置函数	(153)
§ 8.3 有关图形模式的语句	(153)
8.3.1 屏幕设置语句	(153)
8.3.2 屏幕颜色设置语句	(154)
§ 8.4 简单图形绘制语句	(155)
8.4.1 画点语句	(155)
8.4.2 画线语句和连续画线语句	(155)
8.4.3 绘制圆、椭圆和圆弧的语句	(157)
8.4.4 图形着色语句	(157)
§ 8.5 简单的动画功能和声音功能	(158)
8.5.1 简单的动画功能	(159)
8.5.2 简单的声音功能	(159)
习 题 八	(160)
第九章 数据文件	(161)
§ 9.1 文件的基本概念	(161)
§ 9.2 顺序文件的基本操作	(162)
9.2.1 顺序文件的打开	(162)
9.2.2 顺序文件的关闭	(163)
9.2.3 对顺序文件的输出(写文件)	(163)
9.2.4 从顺序文件输入(读文件)	(164)
9.2.5 对顺序文件的修改	(166)
§ 9.3 随机文件	(166)
9.3.1 随机文件的打开与关闭	(166)
9.3.2 对随机文件的读写	(167)
习 题 九	(168)

第一章 QBASIC 基础知识

本章首先介绍程序设计的基本步骤、什么是算法、什么是流程图以及如何画流程图等问题;然后简单介绍 BASIC 语言的发展过程和 QBASIC 语言的特点;最后较详细地介绍 QBASIC 程序的组成和 QBASIC 的编程环境,以及如何在环境中编辑(主要包括输入一个程序和修改一个程序)、保存和运行一个程序,其中还介绍了如何进入和退出该环境。

§ 1.1 算法与流程图简介

1.1.1 程序设计的基本步骤

人们通过程序与计算机交流,使计算机为人们服务。所以如何通过程序设计语言反映人们的需求,使计算机按照人们的要求工作,是程序设计的核心。这就要求程序员首先了解程序设计的基本步骤,一般来说程序设计有如下的五个过程:

(1)需求分析过程 在接受一个任务之前,必须先调查研究清楚问题的性质、任务和要求,即明确用户要求“做什么”,分析用户提出的问题需不需要用计算机求解,能不能用计算机求解,应该用什么样的计算方法求解;同时要分析问题的规模、复杂程度、实际效益与可能出现的其它问题。在调查分析的基础上再将该问题进行简化与抽象,并用适当的图形和语言将它描述、表示出来。即抽象出该问题的三要素:用户要求输出的信息,用户提供输入的信息,输入与输出之间应完成的工作。

(2)设计过程 分析是设计的基础,设计过程包括建立数学模型、选择或设计合适的计算方法、设计具体的算法。只要设计的算法无误,编写程序过程就会比较容易实现,所以设计过程是用计算机解题的关键过程。

(3)程序编写过程 根据算法和流程图使用计算机高级语言编写具体程序的过程。

(4)程序上机调试过程 一个程序往往很难一次编写成功,在程序编写完成后,还需要多次上机调试,以发现其中的错误,不断修改,直到程序能达到预期的目的为止。调试方法可以取多组数据运行程序,得到输出结果,检测所得结果是否满足用户的所有需求。

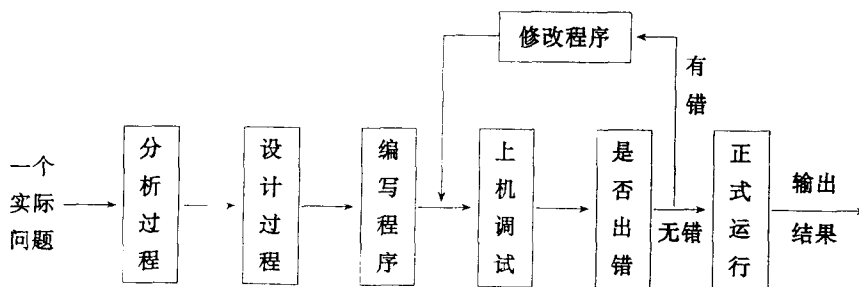


图 1-1 程序设计的五个过程

(5)运行维护过程 程序在运行过程中还可能发现或出现错误,所以还需要不断地找出错误点、修改错误。另外,用户要求也可能变化,这也需要修改程序。因此,程序在运行过程中还需要不

断地进行修改性维护。

以上五个过程如图 1-1 所示。

1.1.2 算法与流程图

对于一个问题的求解,计算机是按照一系列基本操作所组成的过程来完成的。对于不同的软件和硬件环境,所设计的基本操作是不同的。但是,无论基本操作如何设计,计算机这个问题的步骤始终不变。所以,可以给算法下这样一个定义,即一个算法就是一个有穷规则的集合,其中的规则确定了一个解决某一特定类型的问题的运算序列。

一般来说,一个算法应该具有以下特性:

- (1)有穷性 算法所描述的过程应该在有限时间和有限步骤内完成,而不能是无限的;
- (2)确定性 算法中的每一步的含义都必须清楚无误,不能含有歧义(即不能存在两种以上的解释);
- (3)可行性 算法中的每一个有待执行的步骤都能够精确执行,进行有穷次运算便可以得到确定结果的;
- (4)初始性 算法有零个或多个输入;
- (5)可预期性 算法的执行结果是有效的、可预测的,并有一个或多个输出。

其中,有穷性和可行性是算法最重要的两个性质。

在程序设计过程中,我们常常利用流程图来表示算法。算法流程图又称为程序框图,是算法的一种图形表示法。在编写一个具体问题的求解程序时,先设计其算法,然后根据算法的逻辑关系画出表示程序执行顺序的流程图,再对照流程图编写源程序。这是一种处理复杂问题的有效方法,这样写出的程序不仅容易阅读、检查和修改,而且也容易推广使用。

最常用的流程图的符号及其含义表示如图 1-2 所示。

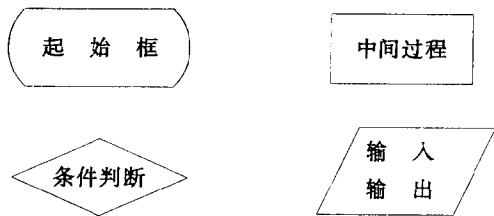


图 1-2 最常用的流程图的符号及其含义

【例 1-1】设计一个求 $S=1+2+3+\cdots+100$ 的算法。

求解该问题的具体算法是:

- (1) 设初值 $S=0$;
- (2) 设初值 $a=1$;
- (3) 将 $S+a$ 的值赋给 S ;
- (4) 将 $a+1$ 的值赋给 a ;
- (5) 当 $a\leq 100$ 时,返回执行步骤(3)、(4),否则执行步骤(6);

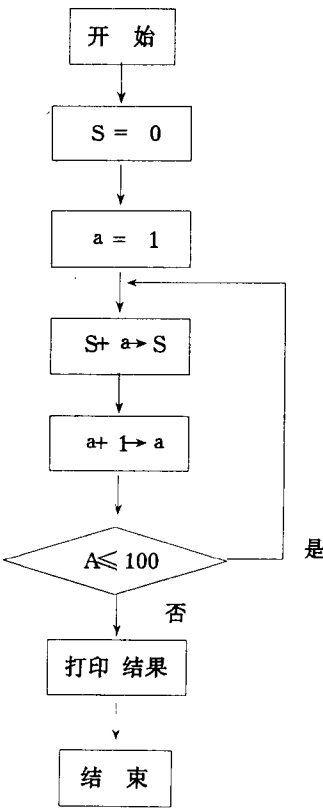


图 1-3 例 1-1 算法的流程图表示

(6)打印总和 S 的值；

(7)结束。

用流程图表示该算法如图 1-3 所示。

§ 1.2 BASIC 语言发展简介

1.2.1 BASIC 语言的发展简史

BASIC 是英文 Beginner's All Purpose Symbolic Instruction Code(即初学者通用符号代码)的缩写,它的初衷是为初学者开发的一种小型程序设计语言,最初是由美国 Dartmouth 学院的 John G. Kemeny 和 Thomas E. Kurtz 两位教授于 1964 年开发完成。当时的 BASIC 语言只有 14 条语句。BASIC 从 1964 年产生至今,经历了四个重要的发展阶段:

(1)第一阶段 从最初的 14 条语句发展至 17 条语句,这就是最“基本的 BASIC”。这时的 BASIC 主要在小型机上运行,以编译方式执行。

(2)第二阶段 20 世纪 70 年代,BASIC 已经成为了一种相当稳定的通用语言,正向着多功能、标准化的方向前进,微型计算机时代来到了。由于当时微型计算机的配置很低,众多的高级语言都难以在微机上实现,而 BASIC 语言由于对硬件要求较低,容易实现,逐步成为了微机必备的语言。在这一个时期内,BASIC 语言得到了长足的发展。但是,由于各种微机的机型不同,在硬件的激烈竞争中,各微机厂商为了表现自己的机器的优势,给 BASIC 加入了许多依赖硬件的特性,导致了同是 BASIC 语言却相互不可兼容的情况,影响了 BASIC 的可移植性和通用性,给 BASIC 的标准化带来很大的困难。此时的 BASIC 改用解释方式执行,方便了用户调试程序。

(3)第三阶段 从 70 年代初开始,为了使程序结构清晰、易读性强,提高程序设计的效率,结构化程序设计的方法开始萌芽。人们也逐渐认识到,BASIC 语言中的控制结构不适于结构化程序设计的要求,往往需要利用 GOTO 语句以及行号来实现选择和循环。再加上人们对于标准化 BASIC 的强烈要求,在 80 年代中期,美国国家标准化协会(ANSI)在结构化理论的基础上提出了一个 BASIC 标准化的草案。与此同时,出现了一些结构化的 BASIC 语言,主要有 TRUE BASIC, QUICK BASIC, TURBO BASIC 等等。

QUICK BASIC 是微软公司(Microsoft)于 1987 年推出的,它提供了一个友好的菜单界面,用户可以通过菜单对程序进行编辑、编译、调试和运行;它与 GW—BASIC 和 BASICA 高度兼容,还实现了程序的模块化。QUICK BASIC 采用编译方式生成可执行文件,提高了程序的执行效率。

(4)如今,Windows 操作系统以其方便的可操作性、友好的图形界面已经成为当今微型计算机,系统的工作平面,而基于 Windows 操作系统的 BASIC 也应运而生,由 Microsoft 开发的 Visual BASIC 完全兼容了 QUICK BASIC,同时以图形界面取代了过去的字符界面,受到了普遍地重视,是一种很有发展前途的高级语言。

1.2.2 QBASIC 的特点

QBASIC 是 QUICK BASIC 的一个子集,是一个模块化和结构化的 BASIC,它作为 MS-DOS 5.0 及其以上版本的一个组成部分免费提供给用户。和以往的 BASIC 相比,QBASIC 具有以下的一些特点:

(1)扩充了变量和常量的类型 QBASIC 在以往的基本 B 基础之上,支持了长达 40 个字符的

变量名,并且允许大小写自由组合。变量类型除了原有的整型、实型、双精度型和字符串类型外,新增加了长整型、定长字符串变量,以及可定义数值的常量和字符串常量。

(2)增加了新的控制结构 为了适应结构化的程序设计需要,QBASIC 摒弃了以往的借助 GOTO 语句和行号实现的控制结构的缺陷。QBASIC 增加的块 IF 结构、WHILE 型循环和 DO 型循环,可以规范的满足结构化程序设计的需求,使程序清晰、规范化。

(3)定义了全局变量和局部变量 在以往的 BASICA 和 GW-BASIC 中,由于没有定义全局变量和局部变量,故同名的变量使用相同的内存空间,难以实现模块的独立性。而 QBASIC 很好的解决了这个问题。在 QBASIC 中,子程序的变量只在子程序中有效,是局部变量,所以 QBASIC 中局部变量的定义为函数和子程序的模块化提供了条件。

QBASIC 的特点还有很多,例如,QBASIC 提供了一个非常友好的用户操作界面;同时还提供了完备的联机帮助,使用者可以在编程过程中方便的得到各种必须的“帮助”信息;另外,用 QBASIC 编写的程序可以在 QUICK BASIC 或 Visual BASIC 环境中运行,有较好的兼容性。

§ 1.3 QBASIC 的组成及其使用环境

1.3.1 QBASIC 程序的组成

要了解 QBASIC 程序的组成,首先来看一段源程序例子。

【例 1-2】将例 1-1 中的算法用 QBASIC 语言描述。

```
LET S = 0:LET a = 1
WHILE a <= 100
    S = S + a
    a = a + 1
WEND
PRINT "S="; S
END
```

该程序的执行结果如下:

S=5050

从上面的例子可以看出,QBASIC 的程序应该由以下的规则组成:

(1) 一个 QBASIC 程序是由若干行语句组成的。每一行可以包含一条语句,也可以包含若干条语句,在例 1-2 中,第一行中包含了两条语句,其它各行都只有一条语句。如果一行内有两或两条以上的语句,每个语句与语句之间都要用冒号隔开。

(2) QBASIC 的程序可以使用“行号”,也可以不使用行号,为了尽可能的和结构化的程序设计相结合,QBASIC 程序建议不使用行号。

(3) 不论是使用了行号还是没有使用行号的程序,QBASIC 在执行过程中均按照语句的排列顺序依次执行,不按照行号顺序执行。

(4) QBASIC 中还有“行标号”。行标号和“行号”的区别就在于“行标号”以冒号结尾,和行号一样可以作为某一行的标志,常用于 GOTO 语句中。下面给出使用了行号和行标号的 QBASIC 源

程序。

10 INPUT A,B	INPUT A,B
20 IF A<0 AND B<0 GOTO 50	IF A<0 AND B<0 GOTO error
30 PRINT A+B	PRINT A+B
40 END	END
50 PRINT "ERROR"	error:PRINT "ERROR"
60 END	END

(5) 每一条语句要以 QBASIC 的语句关键字开头(仅有赋值语句“LET”关键字可以省略),后面接语句体,例如下面的语句:

LET a=3:INPUT a

其中,“LET”和“INPUT”就是关键字,“a=3”和“a”就是语句体。这些语句告诉 QBASIC 执行什么操作以及操作的对象是什么。

(6) 一个 QBASIC 的源程序只有一个主程序,也可以是包含一个主程序和若干个子程序(或子函数)的程序。有关子函数和子程序的概念将在第七章介绍。

(7) 一个源程序单位由可执行语句和非执行语句组成。可执行语句使计算机完成用户指定的操作;非执行语句(如 REM,注释语句)仅仅是为了方便人们对程序的理解、使用和维护,计算机不执行任何操作。

(8) 每个程序的最后都以 END 语句结束,一个程序执行时遇到 END 语句就结束其操作。

1.3.2 QBASIC 的使用环境

由于 QBASIC 是 MS-DOS 5.0 及其以上版本的一部分,所以, QBASIC 由 QBASIC.EXE 和 QBASIC.HLP 两个文件组成。其中 QBASIC.EXE 是 QBASIC 的主体,它以菜单的形式为用户提供人一机对话功能,并解释执行用户程序;而 QBASIC.HLP 则是 QBASIC 的联机帮助,负责向用户提供 QBASIC 编辑环境和 QBASIC 语法规则的有关信息。

QBASIC 可以在 DOS 环境或者 WINDOWS 环境下启动,如果用户安装了 MS-DOS 5.0 或更高版本的 DOS,在 DOS 的安装目录下就已经拥有了 QBASIC。在 DOS 环境下,可以按照以下步骤启动 QBASIC:

(1) C:\>CD DOS (在当前盘符为 C 的情况下,进入 DOS 安装目录)

(2) C:DOS\>QBASIC (启动 QBASIC)

如果没有安装 MS-DOS 5.0,用户可以自建一个子目录,将 QBASIC.EXE 和 QBASIC.HLP 这两个文件复制到该子目录。然后进入该子目录,执行上面的第二步运行 QBASIC。

如果用户使用的是 WINDOWS 95 或 98 操作系统,可以在“开始”菜单中选择“查找”,在查找文件中输入关键字 QBASIC,找到 QBASIC 所在目录,双击 QBASIC 运行即可。

每次启动 QBASIC 后,都会看到如图 1-4 的画面,如果按下“回车”键,将进入 QBASIC 的联机帮助;如果按下 ESC 键,则关闭 Welcome 窗口,进入图 1-5 的界面,这就是 QBASIC 的程序编辑画面。现在我们就可以在 QBASIC 中输入自己的程序了。

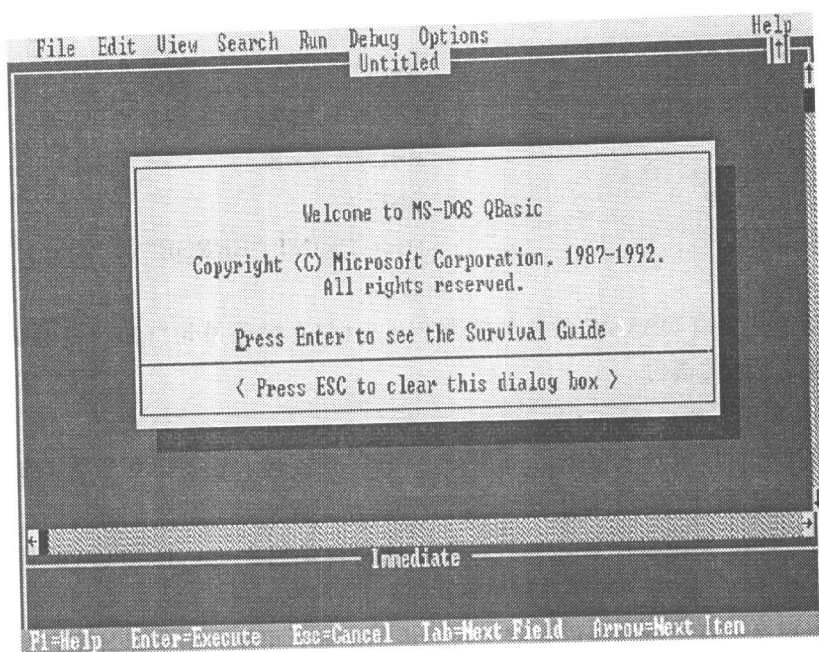


图 1-4 QBASIC 的 Welcome 画面

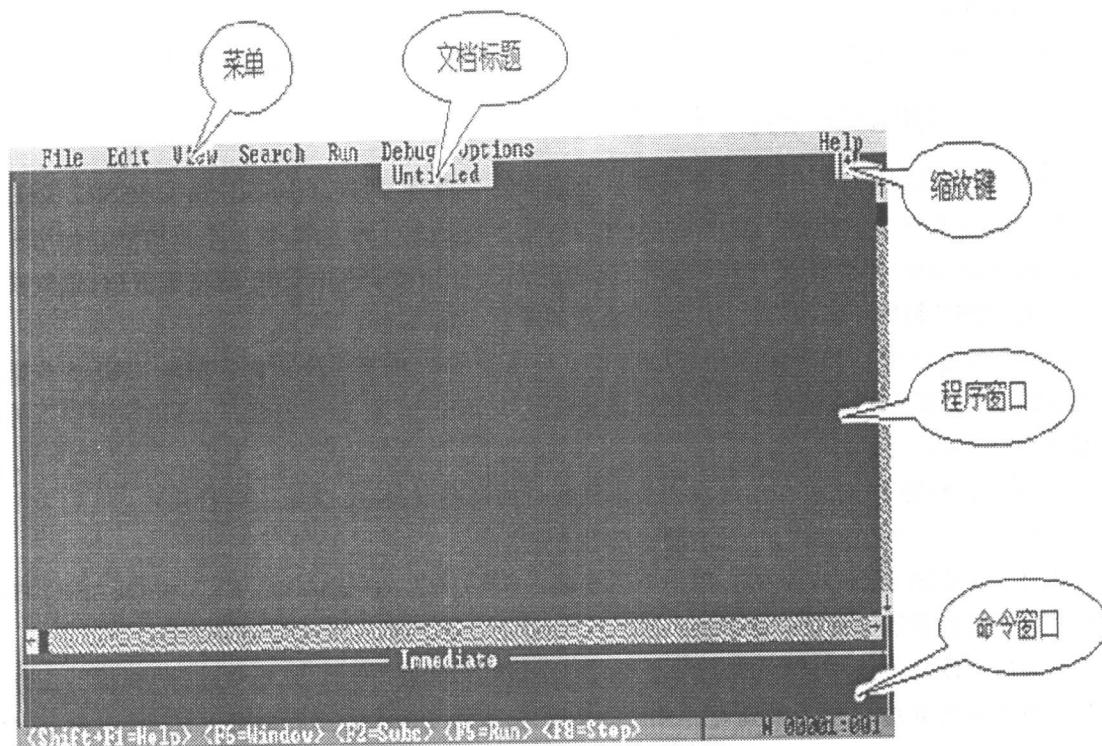


图 1-5 QBASIC 的程序编辑界面

从图 1-5 中我们可以看到, QBASIC 的主界面分成三个部分, 最上面的是菜单, 中间的窗口是程序编辑窗口, 最下面的窗口是命令窗口。

(1) 菜单简介 QBASIC 的大部分操作是由菜单项和中间的程序编辑窗口完成的。在进入程

序编辑状态后,按下 Alt 键就可以进入菜单项进行选择。QBASIC 提供的菜单命令项有:File(文件菜单),Edit(编辑菜单),View(视图菜单),Search(查找菜单),Run(运行菜单),Debug(调试菜单),Option(选项菜单)和 Help(在线帮助菜单)。

(2) 程序编辑窗口 在菜单下就是程序编辑窗口,它用来显示我们所输入的源程序,并提供修改和解释功能。在窗口的正上方是源程序的文件名,如图 1-5 中,文件没有保存,显示 untitled。在窗口的右上角,有一个缩放功能键,用来放大或缩小程序编辑窗口。图 1-6 是输入了例 1-2 的源代码,并保存为 a1.bas 文件后的 QBASIC 窗口。

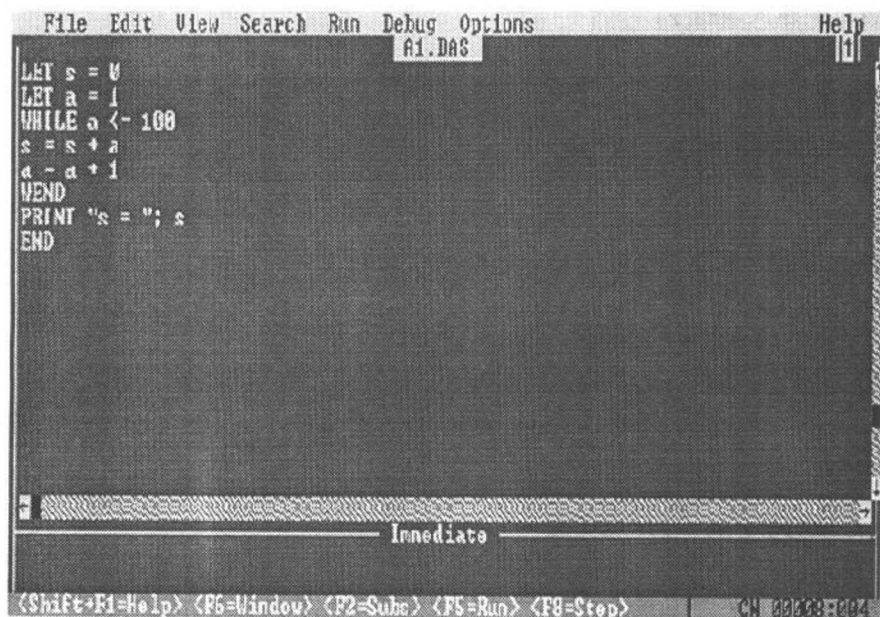


图 1-6 程序编辑窗口和命令窗口

(3) 命令窗口 在 QBASIC 中按 F6 键,可以在程序编辑窗口和命令窗口间切换。命令窗口的作用很简单,是用来输入单行命令直接执行使用。它每次执行一条命令,在按下“回车”键后,输出该行语句的结果。

(4) 其它部分 除了上面主要的三个部分以外,QBASIC 还有以下的一些组成部分:

光标 光标在活动窗口中是不停地闪烁着的,它提示下一个输入字符将要出现的位置。光标的形状有两种,一种是闪烁的短横线,一种是闪烁的方块,它们分别代表光标处于插入状态或是修改状态。

鼠标 如果安装了鼠标及其驱动程序,在 QBASIC 中将会出现一个鼠标指针。鼠标的操作大致和 Windows 中的鼠标操作相同。

提示栏 提示使用者当前状态下可以使用的快捷键,如图 1-6 中,提示的快捷键是 <Shift + F1 = Help> <F6 = Window> <F2 = Subs> <F8 = Step>,使用这些快捷键,可以方便的运行程序或是切换窗口等等。

行和列位置提示 在 QBASIC 的界面的右下角,有两个用冒号分隔的数字,它们分别代表当前的行和列的位置,图 1-6 中,从 00017:001 可以看出,光标现在处于第 17 行,第 1 列。

以上就是 QBASIC 的基本使用环境,在今后的 QBASIC 程序设计过程中,会经常用到以上的内容。

§ 1.4 QBASIC 程序的编辑与使用过程

1.4.1 输入 QBASIC 程序

先将程序编辑窗口变成为当前活动窗口,并将光标移动到窗口的第一行第一列,然后一个字符一个字符地输入程序。例如,下面是一个输入三个数,判断是否构成三角形的程序,请在 QBASIC 的程序编辑窗口中输入该程序:

```
M=0
DO
  PRINT "Input three number:"
  INPUT a,b,c
LOOP UNTIL a>0 AND b>0 AND c>0
IF a+b>c AND a+c>b AND b+c>a THEN M = 1
IF M=1 THEN PRINT "This is a triangle!"
IF M=0 THEN PRINT "This is not a triangle!"
END
```

在输入过程中,可以看到,在每一行输入完毕,进入下一行时,QBASIC 将自动把所有的关键字转变成大写字母。

1.4.2 修改 QBASIC 程序

在输入过程中不可能没有错误,为了解决在编辑源程序过程中出现的错误,QBASIC 提供了强大的修改功能。

(1) 简单修改 当源程序中出现了少量的错误,比如将“INPUT”输入成了“INUPT”,可以利用键盘上“→、↓、←、↑”四个键移动光标至需要修改的位置,然后进行修改;或者将鼠标移动至要修改的位置,左键单击该位置,然后进行修改。在修改时,要注意光标所处的状态,利用小键盘的 Insert 键,可以改变光标的状态。

插入字符操作:在光标为插入状态时,可以在当前位置插入字符。

插入行操作:在两行之间插入一行,可以将光标置于第一行的末尾或者第二行的开头,然后按回车键,便插入了一行空行,这时就可以在此空行中输入所需要的内容。

改写字符操作:在光标为改写状态时,可以改写光标当前位置的字符。

删除字符操作:用“Del”键可以删除光标所在位置的字符,用“Backspace”键(即退格键)可以删除光标所在位置的前一个字符。

删除行操作:可以按 Ctrl + Y 删除光标所在行。

(2) 块操作 如果需要对源程序中一大段连续的语句进行操作,可以利用 QBASIC 提供的块操作功能,此项功能需要利用 Edit 菜单。

选择块:按下 Shift 键,同时使用“←、↑、→、↓”移动光标,可以看到用白色标记的一段程序,表示该段程序已被选中;或者按下鼠标左键不放,拖动鼠标,选中所需要的部分。

块操作:在选择了块之后,按 Alt + E,打开 Edit 菜单,可以看到这样几个选项:Cut(剪切),Copy(复制),Paste(粘贴)和 Clear(清除)。在 Cut 或 Copy 两个选项选择后,将光标移动到要修改的位