

**21世纪 计算机基础教育系列教材**

**谭浩强 主编**

# **QBASIC**

# **语言教程**

## **(第二版)**

■ **谭浩强 著**



**电子工业出版社**  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>



21 世纪计算机基础教育系列教材

谭浩强 主编

# QBASIC 语言教程 (第二版)

谭浩强 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

QBASIC 语言是 BASIC 语言的新版本。它是一种完全结构和模块化的计算机高级语言,其功能丰富、用户界面友好、使用方便。QBASIC 语言是没有程序设计经验的人学习程序设计的较佳起点。

全书共分 10 章,主要包括:程序设计的基础知识,QBASIC 语言的基础知识,顺序结构程序设计,选择结构程序设计,循环结构程序设计,函数和子程序,数组,字符串处理,屏幕控制和作图,文件等内容。

本书在第一版的基础上进行了修订,使其重点更突出、内容更紧凑,是学习程序设计的一本好的入门教材。本书可作为大专院校教材,也可作为计算机培训班的教材,亦可供自学者使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,翻版必究。

### 图书在版编目(CIP)数据

QBASIC 语言教程/谭浩强著. —2 版. —北京:电子工业出版社, 2002.1

(21 世纪计算机基础教育系列教材)

ISBN 7-5053-7005-7

I .Q… II .谭… III .BASIC 语言—程序设计—教材 IV .TP312

中国版本图书馆 CIP 数据核字(2001)第 063062 号

丛 书 名: 21 世纪计算机基础教育系列教材

主 编: 谭浩强

书 名: QBASIC 语言教程(第二版)

著 作 者: 谭浩强

责任编辑: 应月燕

排版制作: 电子工业出版社计算机排版室

印 刷 者: 中国科学院印刷厂

装 订 者: 三河市金马印装有限公司

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 20.25 字数: 518 千字

版 次: 2002 年 1 月第 2 版 2002 年 1 月第 1 次印刷

书 号: ISBN 7-5053-7005-7  
TP·4016

印 数: 20 100 册 定价: 25.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换;

若书店售缺,请与本社发行部联系调换。电话 68279077

## 《21 世纪计算机基础教育系列教材》序

21 世纪是信息时代,是科学技术高速发展的时代。计算机技术与网络技术的结合,使人类的生产方式、生活方式和思维方式发生了深刻的变化。在新世纪中,计算机知识已成为当代人类文化的一个重要组成部分。我们要将计算机知识和应用向一切有文化的人普及。

高等学校承担着为社会培养高层次人才的任务,大学生毕业后应当成为我国各个领域中的计算机应用人才,成为向全社会推广计算机应用的积极分子。在大学里应当把计算机教育放在十分重要的位置。

我国高校的计算机基础教育起步于 20 世纪 80 年代初。20 年来从无到有迅速地发展,从理工科专业发展到所有专业,从最初只开设一门语言课到按三个层次设置课程,学时也从三四十小时增加到一二百小时。计算机基础教育已经先后上了几个台阶,现在又需要上一个新的台阶。在新世纪初,我们要求进一步提高大学生应用计算机的能力,以适应科学技术和经济发展的需要。

我们在这里所说的计算机基础教育,是指面对全体大学生的计算机教育;而非计算机专业和计算机专业中的计算机教育的特点则有很大的区别。无论学生的基础、培养目标、教学要求、教学内容、教学方法和教材,都和计算机专业有很大的不同。绝不可简单地照搬计算机专业的模式,否则必事倍功半。计算机基础教育实际上是计算机应用的教育,应当以应用为目的,以应用为出发点。

计算机不仅是一种工具,也是一种文化,工具是可选的,文化却是必备的。对学生来说,它还是全面素质教育的一个重要部分,通过学习计算机知识能激发学生对先进科学技术的向往,启发学生对新知识的学习热情,培养学生的创新意识,提高学生的自学能力,锻炼学生动手实践的能力。多年来的实践证明,对计算机感兴趣的学生,绝大多数都是兴趣广泛、思想活跃、善于思考、自学能力较强、喜欢动手实践的。他们绝不是只会死背书本的书呆子。

我们必须认真分析非计算机专业的特点,根据教学上的需要与可能,制定出恰当的教学要求,使学生在有限的时间内能学到最多的有用的知识。全国高等院校计算机基础教育研究会曾提出了在计算机基础教育中应当正确处理的 10 个关系,即:(1) 理论与应用的关系,(2) 深度与广度的关系,(3) 当前与发展的关系,(4) 硬件与软件的关系,(5) 追踪先进水平与教学相对稳定的关系,(6) 课内与课外的关系,(7) 课程设置与统一考试的关系,(8) 计算机课程与其他课程的关系,(9) 要求学生动手能力强与当前设备不足的矛盾,(10) 计算机技术发展迅速与师资现状的矛盾。在教学实践中,许多学校都创造了丰富的经验。

在非计算机专业的教学中,首先需要解决的问题是:准确定位,合理取舍教学内容。我们必须分清楚:哪些内容是需要,哪些内容是不需要的;哪些内容是目前暂时可以不学而留待以后学的,哪些内容是目前不必学而以后也不必学的;哪些内容是主要的,哪些是次要的。绝不可眉毛胡子一把抓,不加分析、不问主次,使学生感到难以入门。

在教学方法和教材的编写上,要善于用通俗易懂的方法和语言说明复杂难懂的概念。传统的教学三部曲是:提出概念—解释概念—举例说明。我在多年教学实践中对于计算机应用课程总结了新的三部曲:提出问题—介绍解决问题的方法—归纳出必要的概念和结论。从具

体到抽象,从实际到理论,从个别到一般。这是符合人们的认识规律的。实践证明,这样做已取得了很好的效果。

为了推动高校的计算机基础教育,我在 1996 年主编了《计算机教育丛书》,由电子工业出版社出版。编写这套丛书的指导思想是 20 个字:“内容新颖、实用性强、概念清晰、通俗易懂、层次配套。”(也可简单地概括为:“新颖、实用、清晰、通俗、配套”。)先后出版的近 20 种供大学非计算机专业使用的教材,受到高校广大师生的欢迎,几年内发行达 75 万册,大家认为它定位准确、程度适当、内容丰富、通俗易懂、便于自学。

在进入 21 世纪之际,我们根据新时期的要求,按照上面所述的指导思想,重新进行规划,对原有的教材进行了筛选,淘汰了部分内容已过时的教材,同时根据计算机技术和高校计算机基础教育的发展,组织编写了一些新教材,并对原有教材进行了修订和补充,以实现推陈出新、不断提高。

我们遴选了具有丰富教学经验的高校老师编写这套教材。在这套系列教材中,我们提供了多种课程的教材供各校选用,其中包括必修课和选修课。不同专业、不同层次的学校都可以从中选到合用的教材,我们还将根据计算机基础教育的需要不断推出新的教材。

本系列教材是由浩强创作室策划、组织和编写的。参加工作的有:谭浩强、薛淑斌、史济民、吴功宜、边奠英、徐士良、赵鸿德、李盘林、孟宪福、张基温、宋国新、徐安东、毛汉书、李风霞、许向荣、周晓玉、张玲、刘星、秦建中、王兴玲、蔡翠平、訾秀玲等。电子工业出版社对本丛书的出版给予了大力的支持,使得本丛书得以顺利出版。

由于我们的水平和经验有限,加以计算机科学技术发展很快,本丛书肯定会有不少缺点和不足,诚恳地希望专家和读者不吝指正,我们将继续努力工作,使本丛书能尽量满足广大读者的要求。

全国高等院校计算机基础教育研究会会长  
《21 世纪计算机基础教育系列教材》主编

谭浩强

2001 年 7 月 1 日

## 前 言

近年来,计算机科学技术与计算机应用以惊人的速度发展。它已渗透到了人类生活的每一个角落。现代社会的各个领域无一例外地广泛使用着电子计算机。计算机知识已成为当代人类文化不可缺少的重要组成部分。

20 世纪 80 年代初,为了适应当时我国普及计算机知识的需要,我和田淑清等编写了一本《BASIC 语言》,并以该书为教材在中央电视台系统地讲授了 BASIC 语言。当年收看人数超过 100 万人,被认为是我国第一次计算机普及高潮兴起的标志。10 多年来,BASIC 语言在我国迅速推广,仅《BASIC 语言》一书已累计发行超过 1 200 万册,创造了科技书籍发行的世界纪录。可以说,BASIC 语言对我国的计算机教育与普及作出了重要的贡献。

随着计算机技术的发展,各种软件层出不穷,人们使用计算机愈来愈方便,学习计算机的入门方式也呈现多样化的趋势。人们根据自己的基础和工作的需要,选学有关的计算机知识。程序设计是计算机应用人员的基本训练与基本功。学习了程序设计,才能懂得计算机是怎样工作的。大部分理工科高校都把程序设计列为学生的必修课。全国计算机等级考试第二级的考试内容就是程序设计。

程序设计语言的选择也呈现多样化的趋势。不同的工作领域、不同基础的人群,选择不同的计算机高级语言。国内外多年实践表明,BASIC 语言是适合于广大计算机初学者的一种语言。10 多年来,我们许多人就是从 BASIC 语言入门进入了计算机应用的大门。随着计算机科学技术的发展,BASIC 语言也在发展。从 20 世纪 60 年代中期间世的早期 BASIC(一般称基本的 BASIC)到在微机上广泛使用的 GW-BASIC 和 BASICA(一般称 MS-BASIC),到完全结构化的 True BASIC,Quick BASIC 和 QBASIC,20 世纪末又推出了在 Windows 环境下使用的 Visual BASIC。作为第三代 BASIC——QBASIC,具有当代语言的许多特征:它是完全结构化的语言;子程序和函数作为单独的程序模块,区分全局变量与局部变量;可以实现递归;有较强的作图功能;具有良好的编辑环境,界面友好,可以使用键盘和鼠标;有功能丰富的联机“帮助”系统;提供“分步执行”和“跟踪”等调试工具。QBASIC 是没有程序设计经验的人学习程序设计的较佳起点,BASIC 语言在今后仍然是适合广大初学者的“大众语言”。

QBASIC 是 Microsoft 公司开发并随着 DOS 5.0 以上版本提供给用户的。因此,用户不必单独去购买 QBASIC 软件。它也能够 Windows 下运行。相信在我国推广 QBASIC 将对提高 QBASIC 课程的教学质量起到积极的作用。

1996 年,当时国内还没有适用于大学的 QBASIC 的系统教材,为了推动 BASIC 语言教学内容的更新,适应开展 QBASIC 教学的需要,作者编写了本书(第一版),以期抛砖引玉,推动 QBASIC 的普及和应用。

由于本书内容全面而丰富,系统而实用,写法深入浅出,深受社会各界欢迎,被认为是学习 QBASIC 的最好教材。本书出版后,迅速打开了我国推广 QBASIC 的局面,中央电视台以本书为教材举办电视讲座,由作者讲授,向全国播送。全国高等院校计算机基础教育研究会在 1996 年举行的“全国高等院校计算机基础教育研究会'96 学术年会”建议以 QBASIC 取代第二代 BASIC。许多大学和成人院校开设了 QBASIC 课程,并选用本书作为正式教材。全国计算机等

级考试和全国计算机应用技术证书考试(NIT)也决定将 QBASIC 作为考试科目。在这种形势下,许多出版社陆续出版了一些 QBASIC 教材,形成了百花齐放的局面。

根据几年来的教学实践,作者最近对《QBASIC 语言教程》一书做了修订,并出版本书的第二版。第二版是在第一版的基础上进行修订的。主要做了以下几方面的修订。

(1) 大多数学校把程序设计作为第二层次的课程来安排,在学习这门课之前已学习过计算机公共基础课程,已掌握了计算机的初步知识,在本门课中不必重复,因此将原来第 1 章“计算机的初步知识”的内容做了精简,只保留与程序设计直接有关的部分,与原来第 2 章“算法与流程图”的内容合并为第 1 章“程序设计的基本知识”,并对“算法与流程图”的内容做了适当压缩。在这一章中简要地介绍算法与流程图的概念,然后在后面各章中结合具体的程序介绍有关的算法,这样对许多初学者来说,可能更便于理解,更有兴趣。

(2) 对原来第 3 章“BASIC 语言的基本知识”的内容进行压缩,与原来第 4 章“QBASIC 程序设计初步”合并为第 2 章“QBASIC 语言的基本知识”,使读者更快地接触 QBASIC。

(3) 考虑到使用本书的读者并不都是理工科的大学生,因此将部分需要高等数学基础的例题(如用梯形法求定积分)删去,这样可以降低多数读者的学习难度。同时考虑到部分理工科读者的学习需要,保留了少量的具有一定数学知识的例题(如用牛顿迭代法求方程的根),供选学。

(4) 根据计算机应用的发展,删去一些已过时的内容,如不符合结构化程序设计要求的 ON-GOTO 等语句,以及对随机文件读写的烦琐的老方法。

修订后的第二版,由原来的 13 章压缩为 10 章和 5 个附录,使本书重点更突出,内容更紧凑,学习更容易。第二版保留了第一版概念清晰、内容详尽、深入浅出、通俗易懂的优点,相信更会受到大家的欢迎。

对非计算机专业的学生,学习程序设计的要求应当恰当,由于学时有限,要求学生在课程期间能独立地编写出可供实际应用的大程序是不实际的。在学校中学习程序设计课程的目的是使学生掌握程序设计的思路和基本方法,为以后在需要时进一步学习有关的知识打下初步的基础。要求学生要学活,会思考问题,会举一反三,而不是只会记住现成的结论。

参加本书编写工作的,除了谭浩强教授外,还有薛淑斌高级工程师编写了本书第 9 章。全国高等院校计算机基础教育研究会和全国许多高校的老师多年来为推广普及 QBASIC 做了大量的工作,作出许多贡献。在此表示衷心的感谢。

为了帮助读者学习 QBASIC,作者另外编写了《QBASIC 语言教程习题和解答(第二版)》一书。书中提供了本书全部习题解答,以及介绍如何使用 QBASIC 菜单与选项的方法。该书由电子工业出版社出版。

由于作者水平有限,本书难免存在缺点或不足之处,敬请批评指正。

谭浩强谨识

2001.7

教授計算技術的大師  
普及現代科技之巨擘

敬頌譚浩強教授創杰成就

宋健

一九九五年一月

# 目 录

|                              |      |
|------------------------------|------|
| 第1章 程序设计的基本知识 .....          | (1)  |
| 1.1 为什么要学习程序设计 .....         | (1)  |
| 1.2 计算机语言和程序设计 .....         | (1)  |
| 1.2.1 机器语言 .....             | (1)  |
| 1.2.2 高级语言 .....             | (2)  |
| 1.2.3 面向过程程序设计的任务 .....      | (3)  |
| 1.3 程序运行环境 .....             | (5)  |
| 1.3.1 计算机的硬件系统和软件系统 .....    | (5)  |
| 1.3.2 操作系统 .....             | (6)  |
| 1.3.3 翻译程序 .....             | (6)  |
| 1.4 算法与流程图 .....             | (8)  |
| 1.4.1 算法的概念 .....            | (8)  |
| 1.4.2 算法的表示形式 .....          | (9)  |
| 1.5 算法举例 .....               | (18) |
| 1.6 结构化算法和结构化程序设计方法 .....    | (22) |
| 习题 .....                     | (24) |
| 第2章 QBASIC 语言的基本知识 .....     | (26) |
| 2.1 BASIC 语言的发展 .....        | (26) |
| 2.2 QBASIC 的使用环境 .....       | (28) |
| 2.2.1 QBASIC 的启动 .....       | (29) |
| 2.2.2 QBASIC 的工作窗口 .....     | (30) |
| 2.2.3 QBASIC 环境的其他成分 .....   | (31) |
| 2.3 编辑和运行 QBASIC 程序 .....    | (33) |
| 2.3.1 用键盘输入 QBASIC 源程序 ..... | (33) |
| 2.3.2 运行 QBASIC 程序 .....     | (33) |
| 2.3.3 修改和编辑源程序 .....         | (35) |
| 2.3.4 退出 QBASIC .....        | (39) |
| 2.4 QBASIC 源程序的结构 .....      | (39) |
| 2.5 常量 .....                 | (42) |
| 2.5.1 数值常量 .....             | (42) |
| 2.5.2 字符串常量 .....            | (46) |
| 2.5.3 符号常量 .....             | (46) |
| 2.6 变量 .....                 | (48) |
| 2.6.1 变量名和变量的值 .....         | (48) |
| 2.6.2 变量的类型 .....            | (49) |
| 2.7 标准函数 .....               | (51) |
| 2.8 运算符和表达式 .....            | (52) |
| 2.8.1 算术运算符 .....            | (52) |

|                                       |       |
|---------------------------------------|-------|
| 2.8.2 算术表达式 .....                     | (53)  |
| 2.8.3 不同类型数据的混合运算 .....               | (55)  |
| 2.9 编写和运行一个简单的程序 .....                | (56)  |
| 2.9.1 编写一个简单的程序 .....                 | (56)  |
| 2.9.2 向计算机输入一个新程序 .....               | (57)  |
| 2.9.3 将程序存盘 .....                     | (58)  |
| 2.9.4 打开已存盘的文件 .....                  | (58)  |
| 习题 .....                              | (59)  |
| <b>第3章 顺序结构程序设计</b> .....             | (62)  |
| 3.1 输出语句(PRINT 语句) .....              | (62)  |
| 3.1.1 PRINT 语句的一般格式 .....             | (62)  |
| 3.1.2 PRINT 语句的作用 .....               | (63)  |
| 3.1.3 PRINT 语句的输出格式 .....             | (65)  |
| 3.1.4 LPRINT 语句(在打印机上输出) .....        | (69)  |
| 3.2 赋值语句(LET 语句) .....                | (70)  |
| 3.2.1 赋值语句的一般格式 .....                 | (70)  |
| 3.2.2 赋值语句的作用 .....                   | (70)  |
| 3.2.3 对变量的赋值操作 .....                  | (71)  |
| 3.3 键盘输入语句(INPUT 语句) .....            | (74)  |
| 3.4 读数语句(READ 语句)和置数语句(DATA 语句) ..... | (77)  |
| 3.5 恢复数据指针语句(RESTORE 语句) .....        | (80)  |
| 3.6 结束语句(END 语句)和暂停语句(STOR 语句) .....  | (82)  |
| 3.6.1 结束语句(END 语句) .....              | (82)  |
| 3.6.2 暂停语句(STOP 语句) .....             | (82)  |
| 3.7 分步执行和设置断点 .....                   | (83)  |
| 3.7.1 QBASIC 的功能键提示行 .....            | (83)  |
| 3.7.2 分步执行 .....                      | (84)  |
| 3.7.3 设置断点 .....                      | (84)  |
| 习题 .....                              | (85)  |
| <b>第4章 选择结构程序设计</b> .....             | (89)  |
| 4.1 概述 .....                          | (89)  |
| 4.2 关系表达式和逻辑表达式 .....                 | (89)  |
| 4.2.1 关系运算符和关系表达式 .....               | (89)  |
| 4.2.2 逻辑运算符和逻辑表达式 .....               | (91)  |
| 4.3 行 IF 语句 .....                     | (93)  |
| 4.4 块 IF .....                        | (97)  |
| 4.4.1 块 IF 结构的一般格式 .....              | (97)  |
| 4.4.2 块 IF 应用举例 .....                 | (98)  |
| 4.4.3 块 IF 的嵌套 .....                  | (99)  |
| 4.4.4 在块 IF 中使用 ELSEIF .....          | (101) |
| 4.5 多分支选择结构 .....                     | (103) |
| 4.5.1 最基本的 SELECT CASE 结构 .....       | (103) |
| 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 .....      | (106) |

|                                                |              |
|------------------------------------------------|--------------|
| 4.5.4 在 CASE 子句中使用多个条件 .....                   | (108)        |
| 4.5.5 SELECT CASE 结构的一般格式 .....                | (108)        |
| 4.5.6 用不同语句实现多分支选择结构的比较 .....                  | (111)        |
| 习题 .....                                       | (112)        |
| <b>第 5 章 循环结构程序设计 .....</b>                    | <b>(114)</b> |
| 5.1 概述 .....                                   | (114)        |
| 5.2 WHILE 循环结构 .....                           | (116)        |
| 5.3 FOR 循环结构 .....                             | (120)        |
| 5.3.1 FOR-NEXT 循环的结构 .....                     | (121)        |
| 5.3.2 FOR-NEXT 循环的执行过程 .....                   | (121)        |
| 5.3.3 EXIT FOR 语句 .....                        | (126)        |
| 5.3.4 FOR-NEXT 循环举例 .....                      | (126)        |
| 5.4 DO 循环结构 .....                              | (130)        |
| 5.4.1 DO 循环的一般格式 .....                         | (130)        |
| 5.4.2 最简单的 DO 循环 .....                         | (130)        |
| 5.4.3 用 EXIT DO 语句终止循环 .....                   | (131)        |
| 5.4.4 带 WHILE 子句的 DO 循环 .....                  | (132)        |
| 5.4.5 带 UNTIL 子句的 DO 循环 .....                  | (134)        |
| 5.5 循环的嵌套 .....                                | (137)        |
| 5.6 程序举例 .....                                 | (141)        |
| 习题 .....                                       | (148)        |
| <b>第 6 章 函数与子程序 .....</b>                      | <b>(151)</b> |
| 6.1 概述 .....                                   | (151)        |
| 6.2 用户自定义函数——DEF 函数 .....                      | (152)        |
| 6.2.1 单行 DEF 函数 .....                          | (152)        |
| 6.2.2 多行 DEF 函数 .....                          | (154)        |
| 6.2.3 使用自定义函数时应注意的问题 .....                     | (156)        |
| 6.3 块内子程序——子例程 .....                           | (157)        |
| 6.3.1 GOSUB-RETURN 语句 .....                    | (157)        |
| 6.3.2 ON GOSUB-RETURN 语句 .....                 | (159)        |
| 6.3.3 ON KEY( <i>n</i> ) GOSUB-RETURN 语句 ..... | (160)        |
| 6.4 独立模块的子程序 .....                             | (162)        |
| 6.4.1 建立一个子程序 .....                            | (162)        |
| 6.4.2 在 QBASIC 环境中有关子程序的操作 .....               | (164)        |
| 6.4.3 带参数的子程序 .....                            | (166)        |
| 6.4.4 虚实结合 .....                               | (167)        |
| 6.4.5 程序举例 .....                               | (169)        |
| 6.5 模块化的函数 .....                               | (172)        |
| 6.5.1 概述 .....                                 | (172)        |
| 6.5.2 建立一个函数 .....                             | (173)        |
| 6.5.3 函数的调用 .....                              | (174)        |
| 6.5.4 STATIC 选项 .....                          | (175)        |
| 6.5.5 程序举例 .....                               | (176)        |
| 6.6 全局变量与局部变量 .....                            | (177)        |

|                                     |       |
|-------------------------------------|-------|
| 6.6.1 局部变量 .....                    | (177) |
| 6.6.2 全局变量 .....                    | (177) |
| 6.7 过程的嵌套调用 .....                   | (179) |
| 6.8 过程的递归调用 .....                   | (179) |
| 6.9 在屏幕上同时观察两个模块的方法 .....           | (185) |
| 习题 .....                            | (186) |
| <b>第7章 数组</b> .....                 | (188) |
| 7.1 数组和数组元素的概念 .....                | (188) |
| 7.2 数组的建立和引用 .....                  | (189) |
| 7.2.1 数组的建立——DIM 语句 .....           | (189) |
| 7.2.2 数组的引用 .....                   | (191) |
| 7.3 静态数组和动态数组 .....                 | (193) |
| 7.4 一维数组 .....                      | (195) |
| 7.5 二维数组 .....                      | (205) |
| 7.6 数组作为过程的参数 .....                 | (210) |
| 习题 .....                            | (218) |
| <b>第8章 字符串处理</b> .....              | (221) |
| 8.1 概述 .....                        | (221) |
| 8.2 字符串常量 .....                     | (221) |
| 8.3 字符串变量 .....                     | (222) |
| 8.3.1 变长字符串变量 .....                 | (222) |
| 8.3.2 定长字符串变量 .....                 | (223) |
| 8.4 对字符串变量赋值 .....                  | (224) |
| 8.4.1 用 LET 语句给字符串变量赋值 .....        | (224) |
| 8.4.2 用 INPUT 语句给字符串变量赋值 .....      | (225) |
| 8.4.3 用 LINE INPUT 语句给字符串变量赋值 ..... | (225) |
| 8.4.4 用 READ/DATA 语句给字符串变量赋值 .....  | (226) |
| 8.5 字符串的连接 .....                    | (227) |
| 8.6 字符串的比较 .....                    | (228) |
| 8.6.1 字符比较的规则 .....                 | (228) |
| 8.6.2 字符串比较的规则 .....                | (228) |
| 8.6.3 字符关系表达式 .....                 | (229) |
| 8.6.4 举例 .....                      | (229) |
| 8.7 字符串数组 .....                     | (231) |
| 8.8 子字符串 .....                      | (235) |
| 8.9 有关字符串的函数 .....                  | (240) |
| 8.9.1 测字符串长度的函数 .....               | (240) |
| 8.9.2 字符串与数值间的转换函数 .....            | (241) |
| 8.9.3 字符与 ASCII 码间的转换函数 .....       | (242) |
| 8.9.4 大小写字母之间的转换 .....              | (244) |
| 8.9.5 建立由相同字符组成的字符串函数 .....         | (245) |
| 8.9.6 日期和时间函数 .....                 | (246) |
| 8.9.7 INKEY\$ 函数 .....              | (247) |
| 8.10 字符串处理程序举例 .....                | (248) |

|                             |              |
|-----------------------------|--------------|
| 习题 .....                    | (250)        |
| <b>第9章 屏幕控制和作图 .....</b>    | <b>(252)</b> |
| 9.1 文本模式与图形模式 .....         | (252)        |
| 9.1.1 文本模式 .....            | (252)        |
| 9.1.2 图形模式 .....            | (252)        |
| 9.2 屏幕控制语句 .....            | (253)        |
| 9.2.1 LOCATE 语句 .....       | (253)        |
| 9.2.2 WIDTH 语句 .....        | (253)        |
| 9.2.3 CLS 语句 .....          | (253)        |
| 9.2.4 CSRLIN 和 POS 函数 ..... | (253)        |
| 9.2.5 SCREEN 语句 .....       | (254)        |
| 9.2.6 COLOR 语句 .....        | (255)        |
| 9.2.7 程序举例 .....            | (256)        |
| 9.3 画点和画线 .....             | (258)        |
| 9.3.1 画点 .....              | (258)        |
| 9.3.2 画线 .....              | (261)        |
| 9.3.3 程序举例 .....            | (263)        |
| 9.4 画圆、椭圆和圆弧 .....          | (265)        |
| 9.4.1 画圆 .....              | (265)        |
| 9.4.2 画椭圆 .....             | (267)        |
| 9.4.3 画圆弧 .....             | (268)        |
| 9.5 图形着色 .....              | (269)        |
| 9.6 综合举例 .....              | (270)        |
| 习题 .....                    | (272)        |
| <b>第10章 文件 .....</b>        | <b>(273)</b> |
| 10.1 文件的概念 .....            | (273)        |
| 10.1.1 文件的分类 .....          | (273)        |
| 10.1.2 文件与记录 .....          | (274)        |
| 10.1.3 文件名 .....            | (274)        |
| 10.1.4 文件的读写和文件缓冲区 .....    | (275)        |
| 10.1.5 文件指针 .....           | (276)        |
| 10.2 顺序文件 .....             | (276)        |
| 10.2.1 顺序文件的概念 .....        | (276)        |
| 10.2.2 顺序文件的打开和关闭 .....     | (277)        |
| 10.2.3 对顺序文件的输出(写文件) .....  | (278)        |
| 10.2.4 从顺序文件输入(读文件) .....   | (282)        |
| 10.2.5 对顺序文件的修改 .....       | (284)        |
| 10.2.6 顺序文件有关的函数和语句 .....   | (287)        |
| 10.3 随机文件 .....             | (289)        |
| 10.3.1 随机文件的概念 .....        | (289)        |
| 10.3.2 随机文件的打开和关闭 .....     | (290)        |
| 10.3.3 记录型变量 .....          | (291)        |
| 10.3.4 对随机文件的读写语句 .....     | (293)        |
| 10.3.5 记录变量应用举例 .....       | (293)        |

|                                |       |
|--------------------------------|-------|
| 10.3.6 随机文件用到的语句和函数 .....      | (296) |
| 10.4 QBASIC 处理文件与目录的语句 .....   | (297) |
| 习题 .....                       | (299) |
| <b>附录</b> .....                | (301) |
| 附录 A 字符与 ASCII 代码对照表 .....     | (301) |
| 附录 B QBASIC 保留字 .....          | (304) |
| 附录 C QBASIC 语句一览表 .....        | (305) |
| 附录 D QBASIC 函数一览表 .....        | (309) |
| 附录 E PRINT USING 语句的格式字符 ..... | (311) |
| <b>参考文献</b> .....              | (312) |

# 第1章 程序设计的基本知识

## 1.1 为什么要学习程序设计

人们都知道计算机能完成各种复杂的任务。在 Windows 环境下,只要用鼠标单击某一菜单项,就能使计算机执行某一项操作(例如复制、粘贴、删除)。有些人感到计算机真神秘,甚至不可思议。其实,计算机执行每一个操作都是按照人们事先指定的内容和步骤进行的。这些事先指定的内容和步骤就称为程序。不同的程序产生不同的操作。软件的设计者事先设计好一个个程序,分别对应不同的操作,当人们单击鼠标时,就会触发相应的程序,从而引起相应的操作。

因此,如果不满足于简单地使用现成的软件,希望由自己来规定计算机工作的内容和步骤,或者想对已有的软件增加或修改某些功能,就应当学习程序设计。

对于大、中学生来说,程序设计是计算机基础知识的一个重要组成部分。通过学习程序设计,可以使我们懂得计算机是怎样进行工作的,不会在计算机的强大功能面前感到“不可思议”。

## 1.2 计算机语言和程序设计

语言分为自然语言和人工语言两大类。自然语言是人类在自身发展的过程中形成的语言(如中文、英文等),是人与人之间传递信息的媒介。人工语言指的是为了某种目的而自行设计的语言。计算机语言就是人工语言的一种。

计算机语言是用于人与计算机之间通信的语言,人们通过计算机语言编写的程序来控制计算机的操作,例如,指示计算机计算和输出一个班的平均成绩。

人们要使用计算机,使计算机按人们的意志进行工作,就必须使计算机能理解和执行人们给它的指令。这就需要找到一种人和计算机都能识别的语言。

### 1.2.1 机器语言

计算机是不能识别与执行人类的自然语言的。计算机内部存储数据和指令是采用二进制(“0”和“1”)方式的。计算机只能接收和识别“0”和“1”这样的二进制信息。每一类型的计算机都分别规定了由若干个二进位的信息(即若干个“0”和“1”组成的信息)组成一条指令。

这种计算机能直接识别和执行的二进制形式的指令称为“机器指令”。例如 1011011000000000 和 1011010100000000 就是某一型号计算机的两条机器指令。每一种计算机都确定有若干种指令(例如,加法指令、减法指令、传送指令、取数指令、存数指令、输出指令……),以实现不同的操作。

一种计算机的指令的集合称为该计算机的机器语言(Machine Language),或者说该计算机的指令系统。正如同用算盘算题一样,每一条珠算口诀就是一条“指令”,算盘全部口诀之和

就是“珠算语言”。也就是说,“语言”是全部指令的总和。人们为了解决某一问题,可以从该“语言”中选择所需的指令,组成一个指令序列。这个指令序列称为“机器语言程序”。

不同的计算机系统的电路逻辑是不同的,因此,对不同的计算机,即使是执行同一种操作(例如在两类不同的计算机上都执行一次加法操作),它们的机器指令是不同的。或者说,不同的计算机有不同的指令系统。譬如有的计算机指令的长度为 16 位,有的计算机则为 32 位,假如用 A 类型机器上的机器语言编写了一个程序,拿到 B 类型机器上就不能用,需要重新编写程序。这显然是很不方便的。因此说,机器语言是依赖于具体计算机的(而不是各类计算机都通用的),它是“面向机器”的语言。

用机器语言编写出的程序,计算机能直接识别和执行,执行效率比较高。但人们要直接用“0”和“1”这样的二进制代码编程序,实在是难学、难记、难写、难检查、难调试、难以推广。只有在计算机产生初期,计算机专门人员才用机器语言编写程序。

### 1.2.2 高级语言

为了解决机器语言的上述缺陷,人们创造了一种各类计算机都通用的、接近于人类“自然语言”和“数学语言”的程序设计语言。譬如写出以下一条指令:

`print sin(a+b)+cos(a-b)+3.6`

“print”是一个英文单词,意思是“打印”。“ $\sin(a+b)+\cos(a-b)+3.6$ ”是一个数学式子,它的数学含义是:分别计算 $(a+b)$ 的正弦值和 $(a-b)$ 的余弦值,把它们相加之后再加 3.6。以上是一条接近自然语言(英文)和数学语言的指令。如果计算机能执行这样的指令,将为使用者提供极大的方便。

这种人工创造的语言称为“高级语言”(High-level Language),机器语言称为“低级语言”(Low-level Language)。所谓“低级”,指它直接贴近机器;“高级”指离机器远一些,不是直接面向机器的。高级语言是各种计算机都通用的。

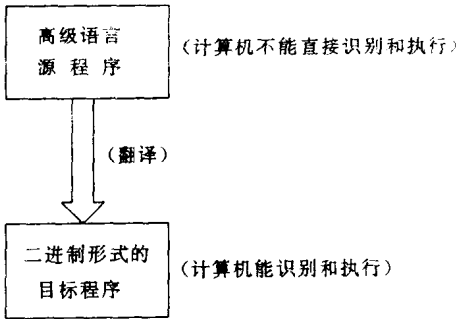


图 1.1 高级语言翻译成计算机语言示意图

世界上最早出现的高级语言是 FORTRAN 语言(1954 年诞生,1956 年正式使用)。它是一种用于数值计算的高级语言。它规定了可以使用的几十个英文单词以及它们对应的操作,并且规定了必须遵循的语法。计算机显然不能识别这种人们自己创造的语言,需要有一个“翻译”,把用高级语言编写的程序翻译成用二进制形式表示的机器语言程序(即由若干条机器指令组成的指令序列)见图 1.1 示意。这个“翻译”工作不是由人来完成的,而是用一个计算机软件来实现的。人们在“创造”一种语言的同时,

必须设计出这样一个翻译系统。在运行高级语言程序之前,先将这个翻译系统调入计算机内,然后让它工作,它会将高级语言程序逐条翻译成机器语言程序,然后计算机执行机器语言程序。

高级语言程序称为“源程序”,翻译后得到的机器语言程序称为“目标程序”(或“目的程序”)。计算机执行的是目标程序,运行后可得到结果。

目前,国内外使用的高级语言种类很多,不下几百种。最常用的也有十几种。运用的范围各不相同。前一时期比较通用的有如下几种。

FORTRAN 语言 适用于数值计算。

COBOL 语言 适用于商业和管理领域。

Pascal 语言 最早出现的结构化语言,适用于教学使用。

C 语言 适用于编写系统软件。

BASIC 语言 一种易学易用又有实际使用价值的语言,尤其适用于初学者。

ADA 语言 一种工程化的大型语言,在国际系统使用。

LISP, PROLOG 语言 用于人工智能领域。

用以上计算机语言编写程序时,必须写出计算机的执行过程,即规定出计算机执行时的每一个具体步骤,即不仅要指出“做什么”,又要指出“怎么做”,计算机根据程序中指定的操作步骤去执行,一步一步地实现目标。因此以上这些语言称为“面向过程的语言”。

近年来,出现了一种称为“面向对象的语言”,它认为客观世界是由许许多多对象组成的。每一个对象包括它所用到的数据及对这些数据的有关操作。在一个程序中包括若干个由程序设计者定义的对象,在程序中向对象发出一个个命令,指定各个对象应完成的操作任务,如果把一个程序看做一个大任务,那么通过每一个对象来完成各个子任务。程序设计的任务是设计对象及通知对象完成所需的任务。C++, Visual BASIC, Java 等语言是适用于面向对象程序设计的语言。有关面向对象的概念在此不做详细介绍。

学习面向对象程序设计需要有面向过程程序设计的基础,因为在一个对象中的操作仍然是面向过程的。现在用来编程的许多优秀的高级语言仍然是过程化的语言。学习用过程化的语言编写程序,是计算机应用人员的一项基本训练。有了这个基础,再掌握面向对象程序设计是毫不困难的。本书只介绍面向过程的程序设计。

自从有了高级语言后,一般的科技人员和大、中学生都能很快地学会使用计算机,而可以完全不顾什么机器指令,也可以不必深入懂得计算机的内部结构和工作原理,就能方便地使用计算机进行各种科学计算或事务管理等。因此有人说,高级语言的出现是计算机发展史中“惊人的成就”。

使用高级语言编写程序还有一个很大的优点,就是它可以适用于不同的计算机,或者说,对不同的计算机具有通用性。用某一种高级语言编写的源程序几乎可以不加修改就能使用在另一类型的计算机上。这就给使用者带来很大的方便。应当指出:即使是同一种高级语言,对不同型号的计算机来说,它所使用的编译系统是不同的,如同把同一篇中文翻译为英文和翻译为法文需要不同的翻译一样。但是这个问题用户不用考虑,在计算机出厂时,已经将该机器所使用的各种语言的编译程序(或解释程序)存储在磁盘上作为计算机系统的软件提供给用户了。

### 1.2.3 面向过程程序设计的任务

什么叫程序(Program)? 先打个比方,人们开会,常事先制订“大会程序”,如第一项“全体起立,唱国歌”;第二项“通过主席团名单”;第三项“领导作报告”……可见,“程序”包括两个方面的内容:要做什么事,按什么顺序进行。计算机程序就是为使计算机完成一个预定的任务而设计的一系列语句或指令。要让计算机实现一组操作,必须先编写程序,然后使计算机执行程序。

下面是一个 BASIC 程序(包含 4 个语句):

```
let    a=3
```