

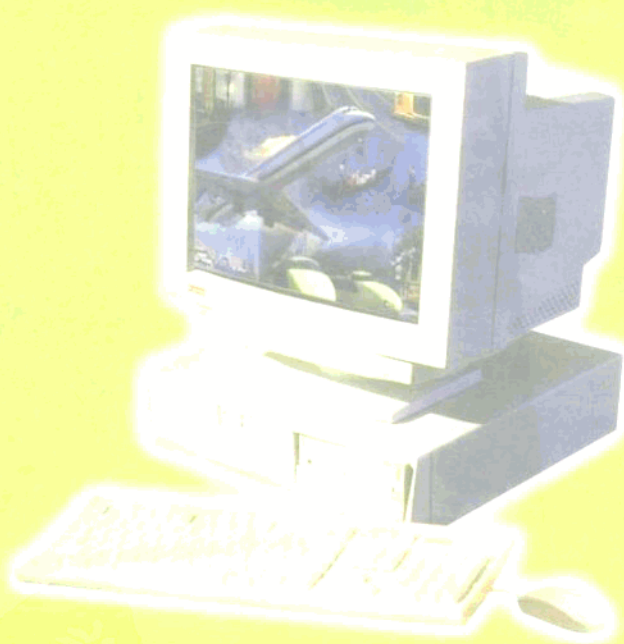
Q BASIC 语言技术基础

哈尔滨工程大学出版社电脑丛书

Q BASIC

语言技术基础

主编 蔡德利 刘 军



哈尔滨工程大学出版社

《QBASIC 语言技术基础》编委会名单

主 编	蔡得利	刘 军	张富志
副主编	刘桂阳	高 军	史国军
主 审	李伟凯		
编 委	蔡得利	刘 军	张富志
	刘桂阳	高 军	史国军
	赵恩民	许善祥	何 燕

《QBASIC 语言技术基础》编审单位名单

黑龙江八一农垦大学
燕山大学
佳木斯大学

前 言

在我国,高等学校非计算机专业的计算机基础教育起源于 80 年代初,此前,我国高校计算机基础教育几乎是一片空白。经过十多年的努力,现在大学里已普及了计算机教育,从无到有地建立了计算机基础教育体系,成绩是巨大的。进入 90 年代,计算机普及进入了新的高潮,高校计算机基础教育也面临着进一步改革和提高的问题。

“全国高校计算机基础教育研究会课程建设委员会”提出,高校非计算机专业的计算机教育应面向应用,以培养各个领域计算机应用人才为目标。因此,计算机基础教育的改革必须紧紧扣住这个目标。课程体系、课程内容、教材建设、教学手段、教学方法等各个环节的改革都必须以加强应用、提高大学生的应用能力和全面素质为出发点。

高校计算机基础教育应分设三个层次:第一层次为计算机文化基础。第二层次为计算机技术基础,教学的主要任务是使学生掌握计算机软硬件技术的基础知识、基本思想和基本方法,培养学生利用计算机处理问题的思维方式和解决本领域一些问题的初步能力。第三层次为计算机应用基础。

非计算机专业的计算机基础教育是为培养本专业的应用型和复合型人才服务的。学生在学习计算机文化基础知识之后,再学习一门计算机语言程序设计是进入第二个层次的必由之路。只有学习编程,才能使计算机按照程序的要求去完成指定的任务,成为计算机的主人;只有学习编程,才能了解计算机是怎样工作的。程序设计是计算机应用人员的基本功,计算机应用人员应当能够根据本领域的需要进行必要的开发工作。

程序设计语言种类繁多,国内外多年实践表明,BASIC 语言是适合广大计算机初学者的语言。多年来,BASIC 语言随着计算机的发展而迅速成长,由 60 年代的基本 BASIC 发展到结构化 BASIC 和可视化 BASIC。“全国高校计算机基础教育研究会”1996 年会上提出推广 QBASIC 语言,QBASIC 由 Microsoft 公司开发,并随 DOS 5.0 以上版本提供给用户,与其他高级语言相比,具有自己独到的特色。

首先,QBASIC 具有良好的编程环境。采用下拉菜单方式工作;随时可以通过联机帮助系统得到帮助信息;在程序窗口编辑源程序,编写时自动进行语法检查,自动对语句书写格式标准化;简单语句可在命令窗口直接执行;程序运行结果显示在用户屏幕上。所有这些使编程工作流程清晰,效率大增。

其次,QBASIC 提供了对结构化程序设计的最好支持。表现在程序不需要行号,编辑源程序时循环和选择结构采用缩进格式;提供了多行 IF 语句,用来解决复杂的两路分支问题;为方便编写多路分支问题的程序,提供了 SELECT-CASE 语句;在对循环结构的支持方面,QBASIC 新增了 DO 循环语句,有 5 种不同的格式,其中有 4 种最能体现 N-S 流程图的 4 种不同循环类型;在程序结构上,支持独立模块的过程,使多人合作编程成为可能。

最后,QBASIC 学习难度适中,有利于普及和提高。通过学习 QBASIC 语言编程,可以掌握结构化设计的基本方法和技巧,对于今后继续学习非常有利,可以很容易过渡到 C、Visual BASIC、FoxPro 等语言。

本书按照循序渐进的原则,根据编者多年教学经验,对 QBASIC 语言的语法作了详细的

讲解，增编了大量的应用程序，以最大限度体现 QBASIC 的语言特点，使读者全面掌握 QBASIC 程序设计知识，达到国家等级考试的要求。

本书由蔡德利统编，李伟凯作了最后的审校。由于时间仓促，错误、不妥之处在所难免，请读者批评指正，以利于再版修订。

编 者

1998.12.20

目 录

第1章 QBASIC 起源与发展	1
1.1 计算机和程序设计语言	1
1.2 BASIC 语言的发展	8
1.3 QBASIC 的使用方法	12
1.4 结构化程序设计方法概述	26
习题	34
第2章 QBASIC 程序设计基础	35
2.1 QBASIC 程序结构	35
2.2 常量和变量	38
2.3 运算符和表达式	48
习题	65
第3章 顺序结构程序设计	67
3.1 数据输出	67
3.2 数据输入	81
3.3 结束、暂停和程序调试	91
习题	93
第4章 选择结构程序设计	97
4.1 两路分支选择结构	97
4.2 多路分支选择结构	104
4.3 应用举例	110
习题	112
第5章 循环结构程序设计	114
5.1 WHILE 循环结构	114
5.2 FOR 循环结构	118
5.3 DO 循环结构	122
5.4 循环的嵌套	128
5.5 循环结构应用举例	130
习题	136
第6章 数组和记录	139
6.1 数组的概念和使用	139

6.2 一维数组	147
6.3 二维数组	157
6.4 记录类型	162
习题	168
第7章 函数与子程序	173
7.1 DEF 函数	173
7.2 模块化函数	176
7.3 子例程	180
7.4 独立模块的子程序	182
7.5 模块间的参数传递	185
7.6 变量的生存期和作用域	189
7.7 过程的嵌套调用与递归	192
7.8 程序举例	196
习题	201
第8章 文件	203
8.1 文件的概念	203
8.2 顺序文件	209
8.3 随机文件	223
8.4 二进制文件	230
习题	233
第9章 图形程序设计	235
9.1 PC 显示系统的组成和类型	235
9.2 基本画图语句	239
9.3 图形窗口、重定义坐标和动画	249
9.4 应用举例	256
习题	264
附录1 ASCII 码字符集	266
附录2 键盘扫描码表	267
附录3 QBASIC 关键字	268
参考文献	269

第 1 章 QBASIC 起源与发展

1.1 计算机和程序设计语言

1.1.1 计算机系统概述

1. 计算机系统组成

一台计算机首先由能看得见摸得着的物理实体组成,我们称为计算机的硬件(即构成计算机的物理实体)。同时我们把一台计算机所拥有的硬件的全体称为计算机的硬件系统。但是仅有硬件,计算机并不具有生命,计算机的生命是由一系列程序给予的,我们称为计算机的软件(即用来指挥控制计算机操作的一系列程序的集合),同时我们把一台计算机所拥有的软件的全体称为计算机的软件系统。一台计算机的完整的系统就是由计算机的硬件系统和软件系统组成的。如图 1-1 所示。

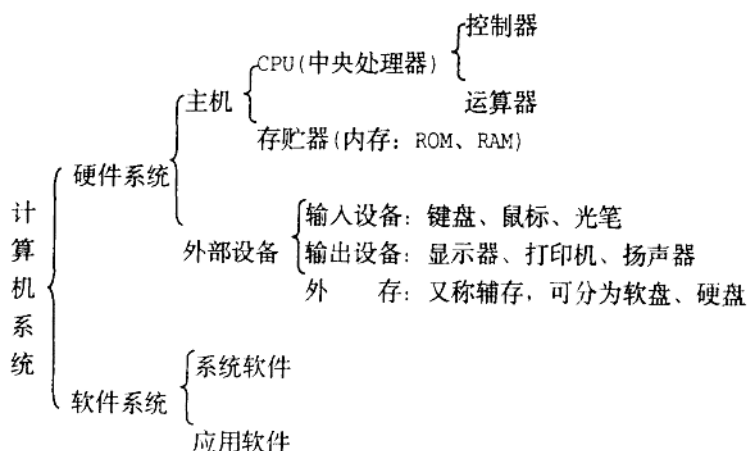


图 1-1 计算机系统结构

2. 硬件系统

计算机的硬件包括存储器、运算器、控制器、输入和输出设备等,即计算机的主机及其外部设备,统称为“硬设备”或“硬件”。

近年来,计算机技术发展很快,硬件配置不断升级。我们应当了解所用的计算机的档次和性能。计算机的性能指标主要包括以下几个方面。

(1)字长:指计算机能直接处理的二进制数据的位数。字长决定计算机运算的精度,字长愈大,精度愈高,直接寻址的能力愈强。根据字长,计算机分为 8 位机、16 位机、32 位机、64 位机等。

(2)运算速度:即每秒钟能执行多少条基本指令。目前高档微机的运行速度可达几百万次/秒。巨型机可达几亿、几十亿次/秒,有的甚至达到几万亿次/秒。

(3)内存容量:即内存存储器中能存储信息的总字节数。286 微机配置的内存存储器一般为 640K~1M 字节,386 微机为 1M~4M 字节,486 微机为 4M~8M 字节。586 微机为 8M~128M 字节。随着内存价格的降低,内存容量还会增加。

(4)主频:指计算机的时钟频率。它在很大程度上决定了计算机的运算速度。主频的单位是兆赫兹(MHz)。目前微机的主频范围大体为:286 机 6~20 (MHz),386 机 16~23 (MHz),486 机 25~100 (MHz),586 机 60~166 (MHz),目前市场上已经出现 450 (MHz)微机。

(5)存取周期:存储器完成一次读写信息操作所需的时间称存储器的存取(访问)时间。连续两次读写之间的时间称为存储器的存取周期。微机的存取周期约在几个纳秒到 100 纳秒(1 纳秒= 10^{-9} 秒)。

(6)外存储器容量:过去用的 5 英寸软盘一般为 360K 或 1.2M 字节,3 英寸盘多为 720K 或 1.44M 字节。硬磁盘最早为 10M~20M 字节,现在已达十几 G($1G=2^{30}$)字节。显然,愈大愈好,可存储信息越多。

除以上 6 项指标外,还要注意机器的兼容性,可靠性,可维护性,允许配置的外部设备的最大数目等。

运行一个程序时,首先应考察计算机的硬件配置能否满足要求。

3. 软件系统

广义地说,软件泛指程序、运行程序需要的数据以及与程序有关的文档资料。软件着重研究如何管理机器和使用机器的问题,也就是研究怎样通过软件的作用更好地发挥计算机的功能。一个不包含任何软件的计算机称为“裸机”,在“裸机”上只能运行机器语言程序。显然,它的功能是有限的,操作也是很很不方便的。

计算机软件可分为两大类:系统软件和应用软件。系统软件用来实现某一方面的系统功能,通常是计算机厂商提供的,常见的有:操作系统、高级语言编译程序、网络系统软件等。一般来说,用户使用系统软件,但很少能随意修改它。应用软件是为了实现某一领域的特定功能而编制的,过去多为计算机用户根据本领域的需要而自行开发的,如人事管理程序、工资管理程序、学生成绩管理程序、人口统计程序、桥梁应力计算程序等。近年来,为了减少重复开发,提高软件质量,常由一些软件公司或行业主管部门组织力量根据多数用户需求进行统一开发,然后作为商品软件供应市场。

应用软件种类非常多,常见的有编辑类应用软件,如 EDIT、WordStar、CCED、WPS、WORD 等;图形图像处理类应用软件,如 AutoCAD、PhotoShop、CorelDraw 等;工具类应用软件,如 PCTOOLS、NORTON、ARJ 等。

要使计算机充分发挥其效能,除了要有好的硬件以外,还要有丰富多样的软件。随着计算机应用的日益深入广泛,计算机软件的研究与应用已经越来越显示出它的重要性。

4. 操作系统

运行任何程序,最重要的软件支持环境是操作系统(Operating System,简称 OS)。它

是所有软件的核心。现在几乎所有的计算机都配置了操作系统。它是一个庞大的程序，控制所有的在该计算机上运行的程序并管理这个计算机的所有资源。它的指导思想是：充分利用计算机的全部资源，最大限度地发挥计算机系统的作用。通俗地说，它如同大乐队的指挥，使各部分协调有效地工作。

要对用户作业以及计算机资源进行有效的管理，使它们协调无误和高效率地工作，是一件极为复杂的工作。它要根据计算机运行时出现的瞬息万变的情况，迅速地、准确地作出决策和发出各种命令，使计算机执行相应的工作。操作系统的基本特征是“多任务并行和资源共享”。操作系统是一个庞大的程序，它包括以下五个部分。

(1)CPU 管理模块：它的任务是控制 CPU 每个瞬时的的工作，接收“中断”信号，为用户担任 CPU 处理，充分发挥 CPU 的效能。它是操作系统的核心。

(2)存储管理模块：它的任务是合理地为用户分配存储空间，有效地管理内存。

(3)设备管理模块：它的任务是有效地管理计算机的各种设备，使之有效地进行工作。

(4)文件管理模块：它的任务是管理外存中的程序(包括系统担任的程序和用户程序)和数据，在需要时及时将它们调入内存使用。

(5)作业管理模块：它的任务是合理调度用户作业。所谓用户作业是指用户请求计算机所做的一个完整的工作。操作系统中用来控制作业执行的一组控制程序称为作业管理程序，包括批处理作业或分时作业。

目前在 IBM PC 以及其兼容机上使用较多的操作系统是 DOS(Disk Operating System 意为“磁盘操作系统”)。DOS 提供许多条操作命令，如列文件目录命令、复制命令、格式化命令、删文件命令……使用 DOS，必须熟悉有关的命令。

80 年代中，微软(Microsoft)公司推出的 Windows，1995 年又推出 Windows 95，这是一种基于图形界面的操作系统，使用菜单和鼠标驱动，用户不必去记 DOS 操作命令，把枯燥、难掌握的操作变成形象化、吸引人的操作，Windows 必将代替 DOS 成为微机的主流操作系统。

过去大多数软件是在 DOS 环境下运行的，近来已推出一些专为 Windows 环境设计的软件，因此，在运行一个软件之前，应弄清该软件是“for DOS”的还是“for Windows”的。

1.1.2 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。计算机语言就是用于人与计算机之间通信的人工语言。

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。在计算机语言的发展过程中，不同的阶段语言是各有特点的。

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都是采用二进制(“0”和“1”)方式进行的，即计算机只能接受和识别“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”或“1”组成的信息)组成一条指令。例如,某种计算机规定以 1011011000000000 这样的指令作为“加法”指令,让计算机执行一次加法;以 1011010100000000 作为“减法”指令,让计算机执行一次减法操作。以上两条指令只是第 6 和第 7 位不同(以最左边的位作为第 0 位)。可以看出 16 个 0 和 1 最多可以组成 2^{16} 个不同的指令或信息。

给计算机一个 1011011000000000 指令,实际上就是把 16 个“电子开关”作为一组,使其中序号为 0, 2, 3, 5, 6 的开关“导通”,其它 11 个开关的状态为“截止”,这个电子电路发生的电信号将使两个数据相加。在设计计算机时已经预先规定好每一条指令与电路所产生的操作之间的对应关系。这如同打电话时,拨打指定的电话号码就能找到指定的人,而不会接通别人的电话。

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”。例如前面介绍的 1011011000000000 和 11011010100000000 就是两条机器指令。一条机器指令产生一个相应的机器操作。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……),以实现不同的任务。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。正如同用算盘答题一样,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),它们的指令可能是不同的。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 机器上的机器语言编写了一个程序,拿到 B 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。只有在计算机产生的初期,计算机专门人员用机器语言编写程序。

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”。“ $\text{SIN}(a+b)+\text{COS}(a-b)+3.6$ ”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), 相对的机器语言则称为“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上都通用的。

世界上出现最早的高级语言是 FORTRAN(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)。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, 比较通用的有:

FORTRAN	适用于数值计算
COBOL	适用于商业和管理领域
PASCAL	最早出现的结构化语言, 适于教学和事务处理
PL/1, ALGOL 68	大型通用语言
C/C++	非常适用于编写系统软件, 但也适用于编写各类应用软件
BASIC	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	一种工程化的大型语言
LISP, PROLOG	用于人工智能领域, 特别适合编写专家系统

自从有了高级语言以后, 一般的科技人员和大、中学生以及职工, 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。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。正如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, 同时也生产出了该机器所使用的各种语言的编译程序(或解释程序), 它们存储在磁盘上, 作为计算机系统的软件提供给用户或由用户来购买。

以上列出的语言属于“过程化语言”。编程序以解决一个任务时, 必须写出每一个具体步骤, 既不仅要指出“做什么”, 又要指出“怎么做”, 计算机才能去执行, 一步一步地实现目标。目前已出现一些“非过程化语言”, 这些语言功能比较强, 只需指出“做什么”即可,

不必指定执行的每一细节。例如完成任务“将全校三年级学生中各门功课平均成绩高于 80 分的名单打印出来”，仅需很少的操作计算机就会逐个算出三年级学生的平均分数，与 80 分相比较，并将高于 80 分者打印出来，用户不必过问计算机是如何实现的。这对用户来说方便程度又进了一步。数据库语言 dBASE、FoxBASE、FoxPro、SQL、QBE 等都属于非过程化语言。

现在用来编程的高级语言，大部分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，再掌握非过程化语言是轻而易举的。

3. 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：①要做什么事；②按什么顺序进行的。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，然后使计算机执行此程序。

下面是一个 QBASIC 程序(包含 4 个语句)：

```
LET a=3
LET b=4
PRINT SIN(a+b),COS(a-b)
END
```

这个程序的作用是：将 3 赋给变量 a，将 4 赋给变量 b，然后输出 (a+b) 的正弦值和 (a-b) 的余弦值。执行的顺序是：先对变量赋值，然后输出求出的结果。

程序的规模可大可小，上面是一个很小的程序，最短的程序只有一行，程序的规模和内容完全取决于所要解决的问题。

什么叫“程序设计”(Programming)? 设计、书写及检查调试程序的过程称为程序设计。应该指出，程序设计不仅指在纸上书写程序的过程，它应当包括：分析问题确定解题方法、写出具体操作步骤，即设计算法、正式运行、整理结果、书写程序文档等全过程。

如果要处理一个数值计算任务(例如农作物的产量预测)，应按如下步骤进行。

(1) 首先进行任务分析(分析任务的要求、要给出什么结果、提供什么资源、有无解决的可能)；

(2) 建立数学模型(例如列出一组联立方程式)，在建立数学模型时可能要对问题作一些近似简化处理；

(3) 决定用什么方法去解该数学模型(例如，联立方程组可以用不同的方法求解)；

(4) 根据所选用的方法写出操作步骤，所列出的具体操作步骤称为“算法”(algorithm)；

(5) 选择一种适用的计算机语言，并根据前一步骤所确定的算法编写出程序，这一步又称为“编码”(Coding)；

(6) 上机运行程序。编好程序还不能说已完成了程序设计任务，因为还未验证程序是否正确，必须上机运行程序，并且用各种不同的数据测试在不同的情况下能否得到正确结果；

(7)根据运行情况分析是否有逻辑错误,以及结果是否有错(可能是确定解题方法有错,也可能算法有错,或编程序时出错)。如果有错,则返回前面去修改;

(8)修改后再运行,直到得到正确结果为止;

(9)最后分析整理结果,写出程序文档。

有人说,现成的软件用起来很方便,并且不用自己去编写程序,那么为什么还要学习程序设计呢?在这里,应当说明几点:

(1)要让计算机进行一系列的操作,必须编写程序(如果不编写程序,只能进行单步的简单操作)。任何软件的主要部分都是程序。只不过那些商品软件是由别人编好了程序让用户直接使用而已。不要误认为不需要程序就能使计算机完成各种复杂的任务。有的人看到有些软件功能愈来愈强,感到计算机神秘莫测,而忘记了“计算机是根据人的意志进行工作的”这一根本思想,而人正是通过程序指挥计算机工作的。

(2)学习程序设计,才能使人了解计算机是怎样工作的,例如知道数据是怎样存储和输入输出的,知道赋值和运算的方法,知道图形是怎样画出来的……在使用计算机时才不至于知其然而不知其所以然。

(3)程序设计是计算机应用人员的基本功。一个有一定经验和水平的计算机应用人员不应当只满足于能使用某些现成的软件,利用菜单和鼠标去选择现成的方案。还应当具有自己开发的能力。现成的软件不可能满足一切领域的多方面的需求。计算机应用人员应当能够根据本领域的需要进行必要的开发工作。

4. 翻译程序和翻译方式

前已述及,用高级语言的源程序必须翻译成二进制的目标程序后,计算机才能识别和执行。将高级语言翻译成机器语言程序有两种方式。

(1)编译(Compile)方式

用一个称为“编译程序”的软件进行编译工作。编译过程包括翻译和查错两个功能,具体包括:词法分析、语法和语义分析、生成目标程序、优化目标程序。如果发现语法有错,就向用户报告出错信息,不生成目标程序。用户必须修改程序,再次进行编译。如还有错,还要修改和编译,直到不出现语法错误为止,才能够最后生成目标程序。

(2)解释(Interpreter)方式

解释也是将高级语言源程序翻译为机器指令,但它与编译方式不同,不是把整个高级语言源程序一起翻成一个目标程序。而是翻译一句,执行一句,不产生整个的目标程序。如果程序没有错误,则一直进行到全部执行完毕。如果在运行过程中发现程序有错,则马上中止“解释”工作,需要修改程序后再重新运行。也就是说第一次运行将“前功尽弃”,白白浪费了机器时间。例如整个程序有100条语句,前面98条语句都能正确执行,第99条语句有错,就得不到正确结果,需要重新修改、重新运行。

打个比方:编译方式相当于笔译,得到一篇完整的译文。解释方式相当于口译,说一句译一句,发现说得不对就停止翻译并且全部重来。解释方式使用灵活方便,占内存较少(因为它没有目标程序,因而省了存储空间),但占机器时间多,效率较低。编译方式得到的目标程序经过优化,执行效率高,但占内存多,使用不大方便。编译方式得到的目标程序,以

后每次只需直接运行即可，不必再重新编译，也就是“一次编译，多次运行”，当然如果源程序有修改，就需要重新编译以得到一个新的目标程序，否则运行的仍是修改前的目标程序。

编译或解释工作都是在操作系统的管理下进行的。用户要计算机运行自己编写的程序，先要编写好源程序，再经过编译(或解释)，得到机器指令，在操作系统的统一调度与安排下由计算机执行。从表面上看，好像用户在直接操纵计算机，其实，用户与计算机隔了两层，用户是通过编译(或解释)程序、操作系统与计算机打交道的，也就是说，在操作系统的统一安排下执行机器指令。

现在，绝大多数语言的翻译工作都采用编译方式。早期的 BASIC 采用解释方式，后来的 BASIC 已有了编译方式，有的 BASIC 同时提供解释方式和编译方式。QBASIC 采用解释方式执行程序。

1.2 BASIC 语言的发展

1.2.1 BASIC 语言的产生

在 1964 年以前，计算机数量少而且价格昂贵。人们想要利用计算机运行自己的程序，必须先将程序做成穿孔卡片，交给操作员。操作员将数十、数百个作业作为一批依次输入给计算机，计算机分别运算后输出结果。用户不能当时得到结果，一般要等到第二天才能得到正确的结果。这种成批处理多个作业的方式称作“批处理”方式。这种工作方式对用户使用计算机是很不方便的。人们希望自己上机操作，边运行边修改，当场得到结果。后来出现了“分时系统”，即一台计算机同时为多个用户服务，每个用户“分享”一部分“时间片”。分时系统的出现要求有与之配合的交互式的计算机语言。

FORTAN 语言是为工程计算使用的。使用它之前往往需要接受数周的专门培训。对一般人员来说，不容易掌握。G.Kemeny 和 Thomas E.Kurtz 决定创造出一种新的语言，以便为初学者使用。他们认为这种语言应该具有以下特点：

- (1) 初学者容易学习；
- (2) 是一种通用语言，可以编任何程序；
- (3) 可以进一步扩充功能；
- (4) 采用“交互工作方式”，使用户与计算机能随时对话，以方便使用；
- (5) 给用户友好而清晰的出错信息；
- (6) 对小程序能很快响应；
- (7) 用户不需要对硬件的深入理解；
- (8) 把用户与操作系统屏蔽开。

显然，它的初衷是“使初学者容易学习”，用今天的话来说就是：“方便用户，为群众着想”。这一指导思想，直到今天来看，仍然是十分可贵的，这正是它的独到和成功之处。G. Kemeny 和 Thomas E. Kurtz 把这种新的语言称为 BASIC(Beginner's All-purpose Symbolic

Instruction Code), 意为“初学者通用符号指令代码”。

BASIC 语言最初是为学院的大学生创造的, 目的是使大学生容易使用计算机。由于 BASIC 无可比拟的“容易学习”的特点, 使它很快从校园走向社会, 从美国走向各国, 风靡全世界, 成为初学者学习计算机程序设计的首选语言。

1.2.2 BASIC 的发展阶段

BASIC 语言的发展经历了以下三个阶段。

1. 初期的 BASIC

第一阶段是 BASIC 的初期(1964~70 年代初)。当时由于计算机硬件的限制(主要是内存容量小), BASIC 语言只是一种小型语言, 功能相对较弱。1964 年 5 月 1 日产生的最早的 BASIC, 只有 14 条语句, 后来发展到 17 条语句, 即“基本 BASIC”。今天看来, 初期的 BASIC 语言好像初学走路的孩子一样, 单薄幼稚, 但是 BASIC 的出现却使计算机由只能被专业人员使用发展到能为一般人使用。

这个时期的 BASIC 是在小型计算机(如 pop、Nova 等)上使用的, 多个用户同时坐在计算机前, 通过分时系统共同使用一台计算机, 他们自己输入程序, 即时运行程序, 现场修改错误, 立即得到结果。BASIC 语言的出现开辟了一个“大家都来学计算机”的新阶段。

在这个阶段中, BASIC 语言先后在各种计算机上实现, 采用的是编译方式。各个厂家在实现时都设法对 BASIC 的功能作扩充和改进, 先后出现了多种 BASIC 版本, 语句数量增加了, 运行速度提高了, BASIC 不仅能用于学习, 而且也能用于解决小型的问题。

美国标准化协会(American National Standard Institute, 缩写为 ANSI)于 1974 年开始对 BASIC 进行标准化的工作。它成立的 X3J2 委员会, 制定的第一个 BASIC 标准(即 Minimal BASIC, 小型 BASIC, 于 1978 正式批准), 就是以 1964 年初期的 BASIC 语言为蓝本的。

2. 第二代 BASIC

70 年代初, 是 BASIC 发展过程中的一个重要时期。BASIC 语言已成为一种相当稳定的通用语言, 正当它向多功能、标准化的方向发展时, 一个划时代的发明问世了——1971 年美国英特尔(Intel)公司研制出世界上第一个微处理器, 并由此开辟了微型计算机的时代。这真是对 BASIC 的天赐良机, 由于当时微机硬件配置较低(尤其是内存很小), 其它计算机高级语言难以在微机上实现, 而 BASIC 由于对硬件要求比较低, 很容易在微机上实现。这使 BASIC 语言成为与微机天然匹配的计算机语言。

第一个为微机配置 BASIC 语言的是今天世界闻名的全球首富、微软(Microsoft)公司的总裁比尔·盖茨(Bill Gates)。1974 年春, 他看到《电子学》杂志宣布了英特尔公司推出新的芯片 8080 时, 他敏锐地预感到“大型笨拙的机器的末日很快来临了”。当时很多人还没有看出这小小的芯片将会对计算机的未来发生历史性的变革。

1975 年, 盖茨从一本《大众电子学》的封面上看到由 8080 芯片组装的小计算机“牛郎星 8800”。它没有键盘和显示器, 用 16 个开关来发布指令, 面板上装有几个闪光的小灯泡, 这就是“全部”了。没有软件, 无法运行程序, 几乎是一种摆设而不是实用工具。但是, 盖

茨却决定给它配上 BASIC 语言(尽管他根本还未看到真正的“牛郎星”计算机)。他用哈佛的一台大型计算机来模拟小“牛郎星”，与其中学伙伴一起将 BASIC 压缩成只有 4K 字节的小巧的解释程序。

盖茨将 BASIC 移植到微机上时，考虑到编译方式占内存量大，为了节省内存(这对当时的微机来说是极为重要的)，盖茨采用了“解释方式”。用解释方式运行程序虽然比编译方式慢得多，但对初学者来说，使用比较方便。盖茨在此基础上成立了世界上第一个微型计算机软件公司——微软公司。可以说，今日的“软件大王”盖茨是从 BASIC 起家而走向辉煌的。那时盖茨 19 岁。在盖茨的第一个微机 BASIC 的带动下，以后出现的微机都毫无例外地配置了 BASIC 语言。例如：Cromemco BASIC, Northstar BASIC, TRS-80 BASIC, Apple BASIC, GW BASIC, IBM BASIC(即 BASICA)。

各厂商在推出其 BASIC 时，对 BASIC 的功能都或多或少作了些扩展。同时，硬件技术发展迅速，也为软件的发展提供了宽阔的舞台。BASIC 的功能愈来愈丰富，它能处理文件，具有绘图和音乐功能。BASIC 语言不仅以它的简洁的语法和容易学习的特点广为流行，而且它的实用性使它成为广大专业程序设计人员和普通用户手中的有力工具。BASIC 已经由“小型、简单”的学习语言发展成为功能丰富的实用语言。它的许多功能已经能与其它优秀的计算机高级语言相媲美，有的功能(如绘图)甚至超过其它的语言。当时许多应用软件都是用 BASIC 语言编写的。

二十几年前，微机上配备的软件不多，人们要使用计算机解决问题，大都需要自己编写程序，而 BASIC 正切合广大初学者的要求。这个阶段，BASIC 推广应用达到全盛时期。凡是学习计算机的人，大多都首先学习 BASIC 语言，全世界 BASIC 语言的用户有 1 亿人。中国从 80 年代起普及计算机就是以 BASIC 语言为主要内容的。BASIC 受到千百万人的喜爱和欢迎。具有广泛群众基础的 BASIC 也帮助了微机的推广，而微机的推广又反过来进一步推动了 BASIC 的普及。微型计算机以体积小、功耗低、工作可靠、价格低廉等特点，迅速成了世界计算机市场的主流机型，它载着 BASIC 语言奔向人类社会的各个角落。

在 BASIC 语言竞相发展的过程中，出现了严重的“方言性”，即不同型号计算机的 BASIC 缺乏统一。在这一机器上能运行的 BASIC 程序，拿到另一机器上可能不运行或运行不好。虽然 1978 年通过了一个 BASIC 语言标准，但那个“Minimal BASIC”早已落后于形势的发展，人们迫切地期待着新的 BASIC 标准的诞生。

第二代 BASIC 的代表是 GW BASIC 和 MS BASIC(即 IBM PC 的 BASICA)。

3. 结构化 BASIC

70 年代，著名的荷兰计算机科学家 Edgar W. Dijkstra 在论文中指出：调试与修改程序的困难和程序中的 GOTO 语句数目成正比。他主张限制使用 GOTO 语句。进而提出了结构化程序设计思想，就是将大任务划分成小任务，每一个不同的小任务分别编写程序模块，最后组合在一起调试为一个程序，这种方法便于按功能设计程序模块。第二代 BASIC 语言显然不能满足结构化程序设计的要求。

80 年代中期，美国国家标准化协会根据结构化程序设计思想，提出了一个新的 BASIC 标准草案。BASIC 语言的两位创始人 John G. Kemeny 和 Thomas E. Kurtz 也参与了这一标

], 该标准于 1987 年正式通过, 称为“Full BASIC(X3.113-1987 ANSI)”

七前后, 出现了一些结构化的 BASIC 语言, 主要有 True BASIC、Quick BASIC、Turbo BASIC 等。True BASIC 是 BASIC 语言的创始人 John G. Kemeny 和 Thomas E. Kurtz 于 1984 年推出的, True BASIC 意为“真正的 BASIC”, 以区别于各厂家自行开发的“街头 BASIC (Street BASIC)”。True BASIC 对 BASIC 语言作出了重大改进和发展, 它严格遵循 ANSI BASIC, 不仅完全适应结构化和模块化程序设计的要求, 而且保留了 BASIC 语言的易学易懂、程序易编易调试的优点, 它提供了解释工作方式和编译工作方式。它的两位创始人宣称: True BASIC 的出现将开始 BASIC 的新纪元。在我国有不少人学习并使用了 True BASIC。

Quick BASIC 是 Microsoft 公司于 1987 年推出的。Quick BASIC 提供了一个开发程序的集成环境, 用户编写、修改、编译、调试运行程序均可通过菜单进行操作, 十分方便; 它与 GW BASIC 和 BASICA 高度兼容, 提供了全局变量和局部变量; 程序模块化; 编译后产生一个可执行文件, 执行效率也很高。

在 Quick BASIC 的基础上, Microsoft 公司又在几年前推出了 QBASIC, 并随 MS DOS 5.0 版本一起发行。

第三代 BASIC 语言的特点是: 完全符合结构化程序设计的要求, 多具有编译工作方式。它们的出现大大提高了 BASIC 语言的档次与功能, 受到用户的欢迎。

1.2.3 继续学习

80 年代中, Microsoft 公司推出的 Windows, 又一次给广大计算机用户带来福音。它提供了图形方式的用户界面, 通过鼠标、窗口、菜单等操作计算机, 人们不必去记忆那些枯燥、复杂的 DOS 命令, 使操作变得直观、简单。Windows 迅速赢得了广大用户, 人们使用计算机变得更加容易了。

以前许多软件只能在 DOS 环境下运行, Windows 的出现要求新编制的软件能直接在 Windows 环境下运行, 并能充分享用 Windows 的资源。目前已出现了许多种标有“for Windows”的软件。第一个 for Windows 的 BASIC 语言是 Visual BASIC (意为“可视的 BASIC”, 即图形界面的 BASIC), 它是 Microsoft 公司在 1991 年开发推出的。

Visual BASIC 用于开发 Windows 应用软件, 用它可以设计出具有良好用户界面的应用程序。Visual BASIC 与 Quick BASIC 兼容, 用 Quick BASIC 编写的程序可以不加修改地运行于 Visual BASIC 环境下。

Visual BASIC 是一个强有力的软件开发工具。它一出现就受到普遍重视, 是很有前途的一种语言。现在, Microsoft 公司已推出 Visual BASIC 的第 6 版, Visual BASIC 是能迎接计算机程序设计挑战的最好例子。

除了 Visual BASIC 以外, 属于第四代 BASIC 的还有 GFA-BASIC, CA-REALIZER 等。它们都具有共同的特点:

- (1) 提供图形界面取代过去的字符界面;
- (2) 是“for Windows”的软件。

目前在我国, 第二、三、四代 BASIC 都在流行, 都有众多的用户。第一代简单的 BASIC