

APP

语言结构化程序设计

· 李省耕 · 蔡启先

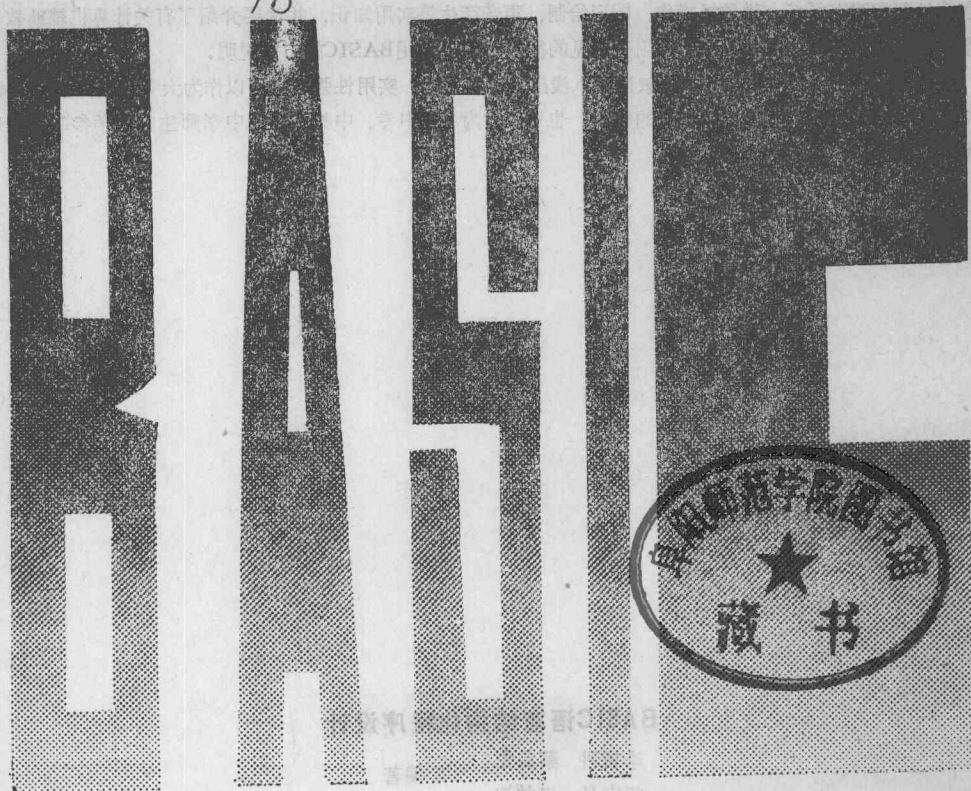
· 杨克昌 · 杨林初

等 编著

中南工业大学出版社

要 容 內

TP311
78



语言结构化程序设计

• 李省耕 • 蔡启先 • 黄贤运
• 杨克昌 • 杨林初 • 方建超 编著

中南工业大学出版社

本书以结构化程序设计思想贯穿全篇，重点介绍各种机型上通用的基本BASIC及其程序设计。同时以我国广泛使用的APPLE II、中华学习机CEC-I为基础，兼顾IBM PC、COMX PC I、LASER310等机型，叙述了字符处理和汉字系统、磁盘机操作、图形绘制、声音产生等实用知识，并简要介绍了有关计算机辅助教学的概念、软件编制和应用实例。书末还附有常见的微机机型中使用BASIC的方法说明。

本书文字通俗易懂，图文并茂，叙述深入浅出，详略得当，实用性强。它可以作为大专院校（特别是师范院校）、电大、业大和计算机培训班的教材，也可以作为各类中专、中技和普通中学师生的教学参考书或自学用书。

BASIC语言结构化程序设计

李省耕 蔡启先 等 编著
杨克昌 杨林初

责任编辑：肖梓高

中南工业大学出版社出版发行
湘潭大学印刷厂印装
湖南省新华书店经销

开本：787×1092 1/16 印张：13.25 字数：339千字

1991年5月第1版 1991年5月第1次印刷

印数：0001—7500

ISBN 7-81020-364-9/TP·015

前 言

本书是为计算机算法语言课程而编写的教材，内容主要包括电子计算机的基本知识，BASIC语言及其程序设计。

编写本教材时，在参考同类教材的基础上，结合我国计算机事业的发展实际，在如下几方面进行了改革。

(1) 将算法语言教学从以介绍算法语言的语法为主转移到以介绍程序设计方法为主的轨道上来。注意对有代表性的算法的介绍，并由浅入深，循序渐进，逐步提高程序设计能力。

(2) 注意计算机技术的发展动向，全书以结构化程序设计思想贯穿始终，以N-S图描述算法，成功地用非结构化语言构造出结构化程序，使读者养成良好的程序设计习惯。

(3) 突出师范性。在选例上较偏重于计算机在教育上的应用，并专辟一章介绍计算机辅助教学。

(4) 加强通用性。注意兼顾文理各专业的需要，行文深入浅出，便于初学者阅读。

(5) 增强实用性。全书以当前使用最普遍的APPLE II机、CEC-I中华学习机上的BASIC语言为基础，兼顾到IBM PC, COMX PC1, LASER 310等常用机型。在教材安排上，第一章至第七章为基本BASIC部分，在各类机型上都能通用。第八章字符运算和汉字使用，注重使读者掌握汉字输入方法。第九章综合应用和第十章磁盘操作系统与文件，可以根据情况选学，以进一步提高学生应用计算机解决实际问题的能力。第十一章可以作为学生课外阅读或教师举办讲座之用。附录列出了几种常用微机的BASIC使用说明及有关资料，特别列出了这几种微机BASIC语句的区别对照表，为拥有不同机型的读者提供了方便。在第一章至第九章中，适当安排了一些要求较高，程序稍显复杂的例题和习题，供学有余力的读者选学。总之，本书作为教材，使用灵活，可以根据课时情况，专业需求和设备条件进行取舍，而不致产生内容断裂之感。

本书蔡启先编写第一章、第十一章；杨林初编写第二章、第九章；杨克昌编写第三章、第十章；黄贤运编写第四章、第七章；李省耕编写第五章、第八章；方建超编写第六章和附录部分。全书由李省耕、蔡启先、杨克昌负责统稿，最后由李省耕、蔡启先定稿。

在编写过程中，得到了各编者所在院校的大力支持，屈义安、黄根民对书稿提出了好的建议，在此一并表示感谢。

本书的编写，是教学改革上的一次尝试，不妥之处，恳请批评指正。

编 者

1990年6月

目 录

前 言

第一章 计算机、算法和程序设计	(1)
§ 1.1 电子计算机的发展及其应用	(1)
§ 1.2 电子计算机的组成与工作原理	(3)
§ 1.3 计算机中数的表示方法	(6)
§ 1.4 计算机语言和软件系统	(9)
§ 1.5 计算机算法	(10)
§ 1.6 流程图	(13)
§ 1.7 结构化程序设计与结构流程图	(15)
练习	(17)
第二章 BASIC语言概述	(18)
§ 2.1 BASIC语言的特点	(18)
§ 2.2 BASIC语言的基本符号	(19)
§ 2.3 BASIC程序的基本结构	(20)
§ 2.4 BASIC中的常量、变量和函数	(21)
§ 2.5 BASIC中的表达式和运算规则	(25)
§ 2.6 程序的输入与修改	(26)
练习	(30)
第三章 数据的输入和输出	(31)
§ 3.1 打印输出语句 (PRINT语句)	(31)
§ 3.2 赋值语句 (LET语句)	(34)
§ 3.3 键盘输入语句 (INPUT语句)	(37)
§ 3.4 读数语句 (READ语句) 和置数语句 (DATA语句)	(41)
§ 3.5 恢复数据区语句 (RESTORE语句)	(43)
§ 3.6 三种提供数据语句的比较	(44)
练习	(46)
第四章 程序转移和选择结构	(48)
§ 4.1 无条件转向语句 (GOTO语句)	(48)
§ 4.2 条件转向语句 (IF - GOTO语句) 和条件执行语句 (IF - THEN语句)	(49)
§ 4.3 选择结构	(53)
§ 4.4 应用举例	(55)
§ 4.5 IF - THEN - ELSE语句	(59)
§ 4.6 注释 (REM)、暂停 (STOP) 和结束 (END) 语句	(60)
练习	(61)
第五章 循环结构	(63)
§ 5.1 当型循环	(63)
§ 5.2 直到型循环	(66)
§ 5.3 循环语句 (FOR - NEXT语句)	(68)

§ 5.4 多重循环	(73)
§ 5.5 应用举例	(76)
练习	(81)
第六章 数组	(84)
§ 6.1 数组的概念和下标变量	(84)
§ 6.2 数组的基本操作	(86)
§ 6.3 数组的应用	(88)
练习	(100)
第七章 模块化程序设计	(102)
§ 7.1 概述	(102)
§ 7.2 转子语句 (GOSUB语句) 和返回语句 (RETURN语句)	(103)
§ 7.3 函数	(108)
§ 7.4 多分支控制语句 (ON-GOTO和ON-GOSUB语句)	(112)
练习	(117)
第八章 字符运算和汉字使用	(119)
§ 8.1 字符串变量和字符串运算	(119)
§ 8.2 字符串函数	(122)
§ 8.3 汉字系统简介	(124)
§ 8.4 应用举例	(128)
练习	(134)
第九章 综合应用	(136)
§ 9.1 图形	(136)
§ 9.2 声音	(149)
§ 9.3 程序调试	(151)
§ 9.4 综合应用举例	(156)
练习	(160)
第十章 磁盘操作系统和文件	(161)
§ 10.1 磁盘与磁盘操作系统	(161)
§ 10.2 磁盘文件	(162)
§ 10.3 DOS命令简介	(163)
§ 10.4 顺序文件	(168)
§ 10.5 随机文件	(172)
§ 10.6 应用举例	(174)
练习	(177)
第十一章 关于计算机辅助教学	(178)
§ 11.1 计算机辅助教学的概念	(178)
§ 11.2 CAI 课件的设计过程和设计方法	(180)
附录	(190)
附录A APPLE II机内ASCII代码表	(190)
附录B APPLE II保留字	(191)
附录C APPLE II错误信息表	(193)
附录D APPLE II打印机的使用	(194)
附录E IBM PC机操作使用提要	(195)
附录F COMX-PC1机操作使用提要	(197)
附录G LASER310操作使用提要	(201)
附录H APPLE II, LASER310, COMX-PC1三种微机基本BASIC差异对照表	(202)

第一章 计算机、算法和程序设计

电子计算机是人类科技与生产进步的产物。它的出现,大大促进了社会的现代化进程。今天,计算机的应用日益深入各个领域,成为人们处理烦杂事务,进行复杂运算的不可缺少的现代化工具。

为了会使用计算机,必须首先了解计算机。什么是电子计算机?人们是怎样利用电子计算机来工作的?本章将介绍有关方面的基本知识。

§ 1.1 电子计算机的发展及其应用

1.1.1 电子计算机的产生和发展

人们在长期的生产实践中,创造并逐步发展了计算工具。

我国很早就有了结绳记事的传说,春秋时出现了“筹算法”(用竹筹记数),唐末创造了算盘,这是人类经过加工制出的最早的计算工具。随着生产的发展,计算日趋复杂,产生了诸如机械计算机(1642年)、计算尺(1654年)、手摇计算机(1887年)乃至电动计算机。然而,这些计算工具的致命弱点是不能自动连续地进行计算,也不能存放大量的中间结果,即计算不能程序控制。尽管以后出现的计算机在这方面有所改进,但是,它们在运算量、运算精度、运算速度等方面远远满足不了生产实践的要求。

1946年,世界上第一台电子计算机“ENIAC”正式交付使用。ENIAC共用了18000多只电子管,1500多只继电器,耗电150kW,占地167m²,重量达130多吨。如此庞然大物,每秒只能进行5000次运算。ENIAC的诞生被称为新的工业革命的开始,誉为本世纪最卓越的科技成就之一。

从ENIAC问世至今四十多年来,计算机的发展突飞猛进。据国外报道,电子计算机每五至八年运算速度就提高十倍,而体积、成本却降低为十分之一。它经历了四个发展阶段:电子管、晶体管、集成电路、大规模集成电路计算机。目前计算机的发展已进入第五代。

前四代计算机都是以美籍数学家冯·诺依曼(Von Neumann)提出的“存储程序”的结构思想为基本原理。这个原理认为,计算机内部以二进制数表示信息;人们把要运算的步骤事先编成指令(以二进制表示),将指令和要运算的数据输入到计算机里存储起来,这就是所谓“存储程序”。计算机根据存好的程序,一步一步进行操作,对数据进行处理及输入输出。这种以二进制和程序存储控制为基础构想的计算机,被称为冯·诺依曼型计算机。

第五代电子计算机是一种非冯·诺依曼型的,它采用了超大规模集成电路或其它新的物理器件,具有全新的工作原理和系统结构,工作过程更近于人脑的思维方式,被称为“推理”型的或者“智能”型的计算机。第五代计算机尚处于研制阶段。当前主要使用的还是第三、四代计算机。

大规模的集成电路技术可以把一个小型计算机的运算器与控制器制作在一块很小的芯片

上, 成为一个微处理器, 并以它为元件构成微型计算机。微型计算机体积小, 性能价格比较高, 品种灵活多样, 应用范围大, 因而被迅速推广到各个领域。近年来, 又出现了在功能上与部分中小型计算机相匹敌的超微型计算机。微型计算机是现代电子计算机中一个兴旺的大家族。

我国从1956年开始计算机的研制。1958年制成第一代电子管数字计算机。1965年研制成功大型通用晶体管计算机。1971年完成第一台集成电路计算机。1974年推出小型系列化计算机DJS-130。1983年, 每秒计算一亿次的“银河”巨型机诞生, 标志着我国计算机事业进入世界先进行列。近年来, 国产长城等系列的PC微型计算机及普及型的中华学习机正在逐步成为我国微型计算机市场的主流。

当前计算机发展的总趋势, 可概括为“四化”, 即“微型化”, “巨型化”, “网络化”和“智能化”以及相应的软件工程开发。计算机的飞速发展, 正在改变着人类的生产方式和生活方式。计算机知识正在成为新一代知识分子知识结构和智能结构的一个重要组成部分。

1.1.2 电子计算机的特点

电子计算机有以下几个特点:

(1) 运算速度快。国外巨型机已达每秒十几亿次。一般微型计算机也能达到几十万次至几百万次。

(2) 精度高。一般计算机可以有十几位有效数字, 微型计算机往往具有7至9位有效数字。

(3) 具有“记忆”和判断功能。可以存贮程序、数据等信息, 能进行逻辑判断并作出相应的后续处理。

(4) 自动按程序所控制的步骤进行, 无需人工干预。

因此, 电子计算机是一种能高速度地进行运算和逻辑判断, 具有存贮信息能力的自动电子装置。

1.1.3 电子计算机的应用

电子计算机最初主要被用作数值计算, 即输入、处理的对象及输出的结果都是数值, 处理的方法是数学计算。但是, 人们很快就发现, 计算机除能进行数值处理外, 还可以处理诸如字母、符号、文字、表格、图形、资料乃至语言、声音等, 随之也发展了各种非数值的算法。比如排序、检索、字符串匹配等等。这样, 大大拓宽了计算机的应用领域。可以说, 无论何处, 或迟或早都是计算机大有作为的天地。

目前, 计算机的主要应用有:

(1) 科学计算或称数值计算。如人造卫星轨道的计算, 天气预报的计算, 导弹弹道的计算等几乎很难用人力解决的繁杂计算问题都得用计算机处理。

(2) 自动控制。计算机能对生产设备乃至整个生产系统进行自动化控制。它能自动检测设备工作情况, 自动调整设备使工作于优化状态, 自动排除故障等。广泛应用于科研、军事、工业农业生产中。

(3) 数据处理。利用计算机对大批数据进行归纳、整理、分类统计等, 将人们从烦琐的办公事务中解放出来。如银行、商业上的记帐, 算帐, 对帐; 工农业产品的合理分配; 工业

企业的各种计划编制,生产预测;图书资料的管理与检索;学校的教学管理,成绩统计等等。

(4) 计算机辅助设计 (CAD)。由计算机部分或全部代替人工进行飞机、机械、建筑、电路、服装等设计,周期短,成本低,质量高。

(5) 计算机辅助教学 (CAI)。利用计算机可以代替教师进行讲解,模拟教学演示和实验过程,辅导学生课外练习等,本书第十一章将作系统介绍。

§ 1.2 电子计算机的组成与工作原理

1.2.1 电子计算机的组成

计算机并不神秘,它是人们按照实际需要而创造的。为了解计算机的构成,先来考察一个简单的运算过程。

设有电路如图1.1,试求其电阻的并联值。

假定用算盘来计算,有如下步骤:

(1) 确定计算方法。根据电路原理,并联电阻 R_1 和 R_2 的总电阻 R 为:

$$R = \frac{R_1 \times R_2}{R_1 + R_2}$$

(2) 将计算式、计算步骤、原始数据等写在纸上。本例中计算公式是:

$$R = \frac{R_1 \times R_2}{R_1 + R_2}$$

计算步骤是:先算 $R_1 \times R_2$,再算 $R_1 + R_2$,最后算出

$$\frac{R_1 \times R_2}{R_1 + R_2}。$$

原始数据是: $R_1 = 10$, $R_2 = 20$ 。

(3) 用算盘计算。先算出 $R_1 \times R_2 = 10 \times 20 = 200$,将中间结果200记于纸上。

再算出 $R_1 + R_2 = 10 + 20 = 30$,将中间结果30记于纸上。

最后算出 $R_1 \times R_2 / (R_1 + R_2) = 200 \div 30 \approx 6.67$,将最后结果6.67记在纸上。

可见,要完成这一计算,必须具备有:

(1) 能进行运算的装置——算盘。

(2) 能保存计算题目、原始数据、运算步骤、运算中间结果和最后结果的装置——纸。

(3) 能书写数据、运算步骤和计算结果的笔。

(4) 最主要的还要有能控制上述装置协调工作的人的大脑。

电子计算机的计算过程与算盘相仿,只是它由机器代替人,同样必需具备类似设备。即运算器(相当于算盘),存储器(相当于纸),控制器(相当于“人脑”),输入设备和输出设备(相当于笔)。

下面对这几个部分予以介绍。

1. 存储器 存储器是用来存放信息,包括数据、程序或者指令的装置。

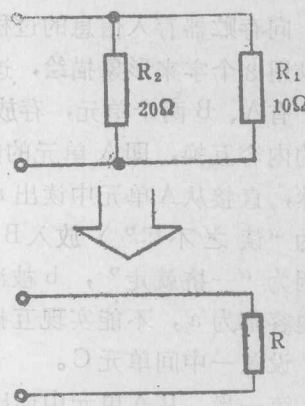


图1.1

存储器一般由千千万万个电路元件组成。它们有两个稳定的工作状态(导通或者截止),可以分别代表二进制的0和1。每一个这样的电路称为一个二进制位,叫作比特(bit)。利用比特位的状态(0或1)的组合,就可以用二进制码形式表示各种信息,实现信息的存贮。

为便于管理,人们将存储器划分为若干个存储单元。每个存储单元内存放的信息称为一个“字”(word);一个字可以含有一个或几个“字节”(byte);一个字节又包含若干“位”(bit)。

通常一个字节由八位组成。8位微型计算机字长8bit,即一个存储单元只含一个字节;16位机字长16bit,即一个存储单元有2个字节。存储器的容量以字节为单位衡量。8位机存储容量最大可达64k字节(这里1k字节代表 $2^{10}=1024$ 个字节),16位机存储容量可达兆字节以上。

为了向指定的存储单元存贮和提取信息,需要对存储单元编号,如同旅馆中的房间都有房间号码一样。存储单元的编号称为地址。存取信息,可以按存储单元的地址查找指定的存储单元。显然,存储单元的地址和存储单元的内容(存放的信息)是不同的两个概念。

向存储器存入信息的过程称为“写”,取出信息的过程称为“读”。存储器的工作特点可以用8个字来形象描绘,这就是:“读之不尽,一挤就走”。现举例说明。

有A、B两个单元,存放内容分别为a和b,现要把它们的内容互换,即A单元的内容为b,B单元的内容为a。

显然,直接从A单元中读出a(这时A单元的内容仍为a,因为“读之不尽”)放入B中,则B单元中的内容变为a

(因为“一挤就走”,b被冲掉了)。此时A、B二单元中的内容都为a,不能实现互换。正确的作法是:

设置一中间单元C。

第一步,从A单元中读出a存入C单元,即 $A \rightarrow C$ 。

第二步,从B单元中读出b存入A单元,即 $B \rightarrow A$ 。

第三步,从C单元中读出a存入B单元,即 $C \rightarrow B$ 。

过程示意图1.2。最后A、B、C三单元中的内容分别为b、a、a。

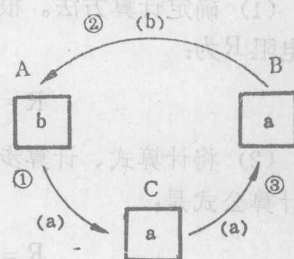


图 1.2

存储器分两种。一种是随机存储器(RAM)。这种存储器的存储单元的内容是可以随时写入和读出的,又称读写存储器;但断电之后,存储内容即自动消失。这类存储器是用户可以使用的空间。另一种存储器叫作只读存储器(ROM)。它存储的内容已经固化,不能改变;用户只能读出,而不能对它进行写操作。断电后,它里面的信息不会失去。这类存储器常用于存放管理计算机的系统软件。

2. 运算器 运算器在控制器的指挥下,从存储器取得数据,并对数据进行运算,然后将运算的结果存入存储器。这里,“运算”不仅是算术运算,还包括对数据进行比较、判断等加工操作。运算器一般由加法器和若干个寄存器组成。

衡量运算器性能的主要指标是运算速度和处理字长等。运算速度是指运算器在单位时间内执行指令的平均条数。一般地,运算器做一次运算要好几个时钟周期。因此时钟频率越高,运算速度就越快。8位机时钟频率约1~4MHz,16位机约为4~10MHz。字长越长,意味着运算器同时处理的信息量大,运算精度高。

3. 控制器 控制器向机器各部分发出控制命令,使整个机器自动地协调工作。这些控制命令的主要内容就是告诉计算机下一步干什么,要操作的数据在何处。命令是人们事先

编制好并已存入机内存贮器的程序。计算机工作时，由控制器把它们逐条读出，控制其它部分按一定顺序，高速度地进行操作。

4. 输入设备和输出设备 输入设备是用来向主机输入数据和程序的设备。常见的输入设备有终端键盘，盒式磁带机，磁盘机等。盒式磁带或磁盘可贮存信息，是计算机的外存贮器。

输出设备是用来输出计算结果或其它信息的设备。常用输出设备有显示终端(显示器)，打印机，磁带机和磁盘机(又作输入设备)等。内存贮器的信息可通过输出设备存入外存贮器中。当然，也可从外存贮器中取出信息放入内存贮器中。

通常，由于运算器和控制器制成在同一块芯片上，它们在计算机中又起着核心作用，故它们合称为“中央处理单元”，即CPU(Central Processing Unit)。内存贮器和CPU合称为“主机”。输入输出设备及外存贮器统称“外部设备”，简称“外设”。APPLE II型机和中华学习机的键盘与主机往往制作在一个机体外壳内。

以上这些设备(主机、外设)称为计算机的“硬件”系统，它们是计算的物质基础。

1.2.2 电子计算机的基本工作原理

计算机硬件系统的五个组成部分之间的联系见图1.3所示，其中实线代表数据传送路径，虚线表示控制路线。可见电子计算机能完成的主要工作有：

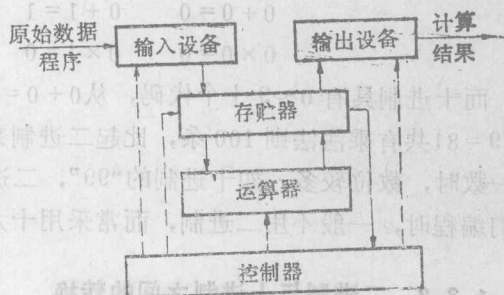


图 1.3

- (1) 输入设备到内存贮器的数据输入。
- (2) 内存贮器到输出设备的数据输出。
- (3) 主机内部的数据传输。
- (4) 由运算器完成的对数据进行算术运算，逻辑比较，移位等处理工作。
- (5) 控制器从内存中读取指令，分析指令，并指挥各部分协调工作。

例如，要用计算机来求出图1.1所示并联电路的等值电阻R，具体工作过程如下：

第一步，分析问题，得到数学模型或算法，编制出有关语言源程序，这部分工作由人完成。

第二步，输入程序和数据。

第三步，由人发出执行命令，启动计算机开始自动工作。先由控制器从内存中逐条取出程序中的指令，进行分析，并按指令要求向有关设备传送命令，指挥有关设备传送数据和处理数据。

第四步，运算结束，按程序要求将有关数据输出给外设，如在屏幕上显示出R的数值，并最后给出运行完毕的信息。

显然，计算机的自动工作有赖于人们事先编制的程序，光有计算机硬件还不行，必须有告诉计算机如何工作的软件，有关程序和软件的概念，将在§1.4中给予介绍。

以二进制形式存放数据，以存储的程序进行控制，这就是冯·诺依曼型计算机的基本工作原理和主要特征。

§ 1.3 计算机中数的表示方法

十进制是人们日常生活中用得最多的计数制。有时，人们也使用其他进制。如六十进制， $1\text{h} = 60\text{min}$ ， $1\text{min} = 60\text{s}$ 。十二进制，一打为12个，一年等于12月，1英尺 = 12英寸等。在计算机内部采用二进制计数；人们编写汇编语言程序时，一般使用16进制；而编写高级语言程序时，又常采用十进制。下面就来讨论这些进制及它们之间的换算关系。

1.3.1 二进制

二进制是逢二进一。它只有0和1两种代码。计算机所贮存和处理的信息，不管它外部的表达意义是什么，它在机器内部都是以二进制数来表示。这是因为，首先，二进制很容易物理实现。元件的截止与导通，脉冲的有或无，都能方便地用1和0来表示；其次，二进制的算术运算十分简单，加法和乘法运算各只有4条运算法则：

$$\begin{array}{llll} 0+0=0 & 0+1=1 & 1+0=1 & 1+1=10 \\ 0\times 0=0 & 0\times 1=0 & 1\times 0=0 & 1\times 1=1 \end{array}$$

而十进制具有0~9十个代码，从 $0+0=0$ 到 $9+9=18$ 共有加法法则100条，从 $0\times 0=0$ 到 $9\times 9=81$ 共有乘法法则100条，比起二进制来要复杂得多。当然二进制也有它的缺点，表示同一数时，数位较多。如十进制的“99”，二进制表示为“1100011”，读写很不方便，所以，人们编程时，一般不用二进制，而常采用十六进制或十进制。

1.3.2 二进制与十进制之间的转换

将十进制整数化为二进制数，采用“除2取余，自下向上排列”的法则，即将十进制整数一次又一次地被2除，直至商为0，每次相除得到的余数从下向上顺序排列成横行。最下边的余数（最后得到的余数）是最高位，就得到用二进制表示的数。如，

$$\begin{array}{r|l} 2 & 215 \text{ (1)} \\ \hline 2 & 107 \text{ (1)} \\ \hline 2 & 53 \text{ (1)} \\ \hline 2 & 26 \text{ (0)} \\ \hline 2 & 13 \text{ (1)} \\ \hline 2 & 6 \text{ (0)} \\ \hline 2 & 3 \text{ (1)} \\ \hline 2 & 1 \text{ (1)} \\ \hline & 0 \end{array} \quad \begin{array}{l} \uparrow \text{最低位} \\ \\ \\ \\ \\ \\ \\ \downarrow \text{最高位} \end{array}$$

即， $(215)_{10} = (11010111)_2$ 。

括弧外的注脚表示括弧中的数是何种进制的数。

将二进制数化为十进制数，采用求“位码与权的乘积的和”的方法，简称“按权相加法”。

位码又称加权系数，一个二进制数的位码，是指对应于这个二进制数的各位上的数码。所谓权，是指各数码所在的位。在二进制中依照二进制数码自右至左的排列，各位的权分别是 2^0 ， 2^1 ， 2^2 ， 2^3 ，…，这里2是进制的基数，指数0，1，2，3，…，随数码的位而定。例

如, $(101)_2$ 是一个三位二进制数 (读作一零一) 它的右边第 1 位位码是 1, 权是 2^0 。右边第 2 位位码是 0, 权是 2^1 , 第 3 位位码是 1, 权是 2^2 。

按位码与权之积相加, 可将二进制数化为十进制数。

如, $(101)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (5)_{10}$

$(10110)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (22)_{10}$

1.3.3 十六进制与十进制之间的转换

十六进制是逢十六进一, 它有 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F 等 16 个代码, 其中 A~F 相当于十进制中的 10~15。

十进制整数转换为 16 进制, 可以采用与十——二进制转换类似的方法, 即除 16 取余, 从下向上顺序排列。

如,

16 10879 (F)	↑ 最低位
16 679 (7)	
16 42 (A)	
16 2 (2)	
0	
	↓ 最高位

即 $(10879)_{10} = (2A7F)_{16}$ 。

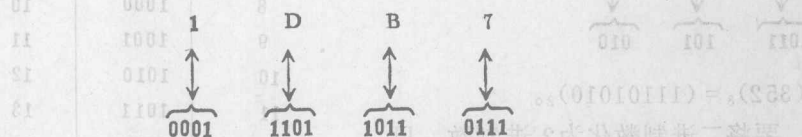
将十六进制数化为十进制数, 同样采取按权相加的办法。由于基数是 16, 故权是 $16^0, 16^1, 16^2, \dots$ 。同时注意位码 A~F 化为十进制数的 10~15。

如, $(1DB7)_{16} = 1 \times 16^3 + 13 \times 16^2 + 11 \times 16^1 + 7 \times 16^0 = (7607)_{10}$

式中十六进制的 D、B 分别相当于十进制中的 13、11。

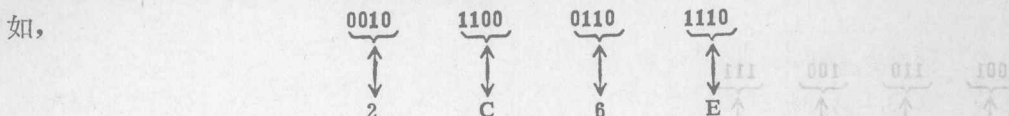
1.3.4 二进制与十六进制之间的转换

由于数 16 与数 2 之间的关系为 $16^1 = 2^4$, 因此 1 位十六进制数相当 4 位二进制数。所以, 将十六进制数化为二进制数, 只要将每一位上的十六进制代码用 4 位二进制数码对应表示就行了。如,



即 $(1DB7)_{16} = (0001110110110111)_2 = (1110110110111)_2$

同样, 要将二进制数化为十六进制数, 只要从个位向左每四位一分节 (不够四位的用 0 补足四位), 每节用 1 个十六进制数码表示即可。



即 $(10110001101110)_2 = (2C6E)_{16}$ 。

1.3.5 任意进制及其转换

引用前述关于基数、权与位码的概念, 一个 n 位的 N 进制整数 A, 可以用下式表示:

$$[A]_N = a_{n-1} \times N^{n-1} + a_{n-2} \times N^{n-2} + \cdots + a_1 \times N^1 + a_0 \times N^0$$

$$= \sum_{i=0}^{n-1} a_i \times N^i$$

其中 N 为基数， N^i 为该数从右向左（第0位起始）第 i 位的权， a_i 为第 i 位上的位码。
 N 进制数只有 N 个代码，即 $0, 1, \dots, N-1$ 。如8进制数只有 $0, 1, 2, \dots, 7$ 这8个代码。

1. N 进制与十进制之间的转换 要将 N 进制整数化为十进制数，采取“按权相加”法。
 如将8进制数化为十进制数：

$$(207)_8 = 2 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 = (135)_{10}$$

要将十进制整数化为 N 进制数，采取“除基数取余，从下向上顺序排列”的方法。如将十进制数化为八进制数：

$$\begin{array}{r} 8 \overline{) 123} \quad (3 \\ 8 \overline{) 15} \quad (7 \\ 8 \overline{) 1} \quad (1 \\ 0 \end{array} \quad \begin{array}{l} \uparrow \text{最低位} \\ \downarrow \text{最高位} \end{array}$$

$$\text{即 } (123)_{10} = (173)_8。$$

2. 2^n 进制与二进制之间的转换 基数为
 $N = 2^n$ 进制的数与二进制数之间有着特别方便的对应关系。即1位 2^n 进制数相当 n 位二进制数。所以将 2^n 进制整数化为二进制数，只要将每一位上的 2^n 进制代码用 n 位二进制数码对应表示就行了。例如，八进制数化为二进制数，由于 $8 = 2^3$ ，故它们之间有1位对3位的关系。如，

$$\begin{array}{ccc} \begin{array}{c} 3 \\ \updownarrow \\ \underbrace{\hspace{1cm}} \\ 011 \end{array} & \begin{array}{c} 5 \\ \updownarrow \\ \underbrace{\hspace{1cm}} \\ 101 \end{array} & \begin{array}{c} 2 \\ \updownarrow \\ \underbrace{\hspace{1cm}} \\ 010 \end{array} \end{array}$$

$$\text{即 } (352)_8 = (11101010)_2。$$

同样，要将二进制数化为 2^n 进制数，只要从最低位向左每 n 位一分节（不够 n 位的用0补足 n 位），每节用1个 2^n 进制代码相应表示即可。例如，二进制数化为八进制数：

$$\begin{array}{cccc} \begin{array}{c} 001 \\ \updownarrow \\ 1 \end{array} & \begin{array}{c} 110 \\ \updownarrow \\ 6 \end{array} & \begin{array}{c} 100 \\ \updownarrow \\ 4 \end{array} & \begin{array}{c} 111 \\ \updownarrow \\ 7 \end{array} \end{array}$$

$$\text{即 } (1110100111)_2 = (1647)_8。$$

表1.1

十进制数	二进制数	八进制数	十六进制数
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

表1.1列出了 $0 \sim 15$ 的十进制数与二、八、十六进制数之间的换算，供读者转换时参考。

§ 1.4 计算机语言和软件系统

语言是人们交换思想的工具。除了人们相互交流用文字或说话的语言外，还有所谓音乐语言、舞蹈语言，……这些语言都具有其特定的表达形式。计算机是在人的指挥下进行工作的，也有特定的语言表达形式，这就是计算机语言。利用计算机语言，可以编写计算机程序。程序就是人们解决问题的工作顺序。指挥计算机做某种特定动作的命令叫指令。所谓计算机程序，就是告诉计算机如何处理问题所写出的一系列指令的集合。计算机程序必须用计算机所能接受的语言来书写。

目前计算机常用的语言有三种：机器语言，汇编语言和算法语言。

1.4.1 机器语言

计算机的控制器只能识别 0 或 1 两种状态，即以二进制数构成的数据。机器语言是用一串二进制数码来表示的机器指令系统。不同类型的机器具有不同的指令系统。如 APPLE II 机的指令系统有 56 条基本指令。通常一条机器指令由操作码与地址码组成。如指令 0110010100110000，操作码是 01100101，表示加法，地址码是 00110000。指令的意思是把运算器中的数加上内存中地址为 00110000 号的存储单元中的数，结果仍保留在运算器中。

用机器语言编写的程序称为手编程序或机器指令程序。机器语言是计算机能懂得的唯一语言，因此，手编程序在机器上运行时速度最快，但由于其难读难记易错，再加上不同机型所用的机器语言往往不同而不便推广。所以，对于机器语言，除专业技术人员，一般人难以问津。

1.4.2 汇编语言

既然每条机器指令都有其特定意义，那么就能用一些简单符号来表达这些意义。基于此，人们创立了符号语言，又称汇编语言。

符号语言是将指令的功能用英语含义的缩写符号来表示。例如，上面提到的那条加法指令，用符号表示就是 ADD \$30，其中 ADD 是英文 Addition（加）的省写，它代替了机器语言中的操作码 01100101。\$30 表示十六进制数，它相当于二进制数 00110000，这样写地址码较紧凑。

显然，符号语言比机器语言好懂好记，但编写起来非常不容易。例如，要实现一次乘法，就必须进行多次移位与加法操作。至于其他计算，则更为复杂，使它的通用性大大受到限制。

用符号语言编写的程序称为符号程序。要使符号程序变为机器能接受的机器指令系统，必须有一个对照翻译的过程，这个过程称为汇编。具有汇编功能的机器指令程序叫作汇编程序，其翻译过程如图 1.4。

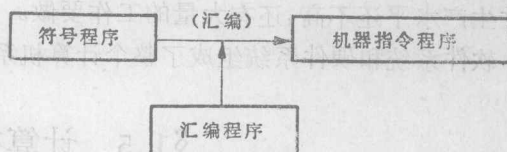


图 1.4

1.4.3 算法语言

算法语言的最大特点是规定了一些与英文原意基本上一致的词汇（称为保留字），从而

象自然语言一样来描述算法。如用“READ”表示读取数据操作，用“PRINT”表示打印显示操作等，而且人们熟悉的数学表达方式几乎都可以直接拿来使用。因此，算法语言一经问世，就得到了普遍应用。

算法语言种类繁多。常见的有 BASIC，FORTRAN（多用于科学计算），PASCAL（多用于数学与结构化设计），COBOL（多用于商业、情报检索），LOGO（多用于中小学教学、绘图）等等，BASIC 语言是其中用途广泛最简单易学的一种。

用算法语言编写的程序称为源程序。源程序必须通过“翻译”才会变成机器能执行的机器指令程序（称为目标程序），这个过程比汇编过程要复杂的多。翻译有两种方式，一种将整个源程序翻译为目标程序。这个过程由编译程序来完成，其过程见图1.5。

另一种是将算法语言源程序逐句翻译，翻译一句，执行一句，边解释，边执行，直至源程序结束。一般BASIC语言采用这种方式，其过程见图1.6。

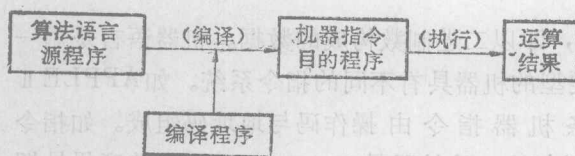


图 1.5

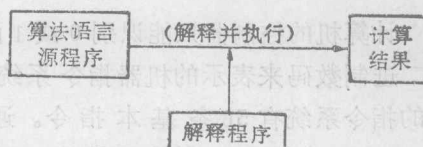
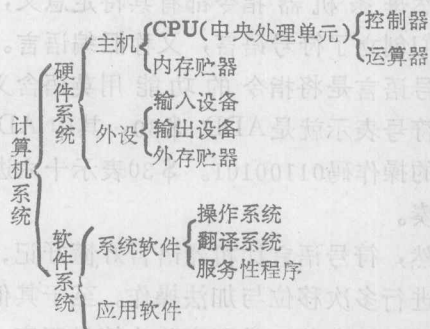


图 1.6

机器语言和汇编语言是面向机器的语言，通常称为低级语言，算法语言是面向问题的语言，通常称为高级语言。

1.4.4 计算机系统的组成

计算机必须配备具有各种专门用途的程序如汇编程序、解释程序等，还要有管理和指挥整个计算机转换信息，分配内存单元，与外设联系，CPU管理等所谓操作系统。有检查维护机器的故障诊断程序，故障维修程序，……这些统称为系统软件。此外，还有为解决具体问题由用户编写的各种各样的应用程序，它们一起构成了计算机的软件系统。显然，没有软件支持的计算机硬件，什么事也不会做。可以说软件是计算机的灵魂。软件系统越完善，计算机就越能发挥更大的作用。当前，国际上对软件研制十分重视，有所谓“三分硬件，七分软件”之说。我国的软件生产水平还不高，还有大量的工作要做。



软件系统和硬件系统组成了整个计算机系统，其组成见右上表所示。

§1.5 计算机算法

为了解决某一具体问题，人们必须事先研究解决这一问题的方法和应该采取的步骤和计划。例如，建筑房屋，就得有规划设计，有造屋的各个工序的统筹安排。所谓算法，就是对

解决某一问题的方法和步骤的精确描述。不仅建房有算法，解数学题有算法，广义地说，任何一个问题的进行过程都有它自己的算法。本节只讨论计算机算法，即如何使用计算机一步一步地进行工作的具体过程，先干什么，接着干什么，再干什么……。当然，设计计算机算法，应是计算机能执行的。

下面举例说明如何设计一个计算机算法。

例1.5.1 统计100个学生某门功课的学习成绩，打印出这门课的总分和平均分。

对这道题，粗略的算法是：

第一步，计算出100个学生的成绩总分并且打印出来；

第二步，求出平均分并且打印出来。

在此基础上，进一步写出下面的算法：

- (1) 将第1个学生的成绩输入计算机；
- (2) 将第2个学生的成绩输入给计算机；
- (3) 将以上两个学生的成绩相加；
- (4) 将第3个学生的成绩输入给计算机；
- (5) 将它和前两个学生的成绩和相加；
- (6) 将第4个学生的成绩输入给计算机；

.....

(198) 将第100个学生的成绩输入给计算机；

(199) 将它和前99个学生的成绩和相加；

(200) 打印出这100个学生的成绩总分；

(201) 将这100个学生的成绩总分除以100得到平均分；

(202) 打印出平均分。

这个算法是可以实现的，但非好的算法。若统计1000个学生的成绩，算法就会写得更长。考察上述算法，发现，每个学生的成绩加入总分，这种操作是重复的。可以让计算机进行“循环”，重复同一操作，直至加完第100个学生的成绩为止。因此，算法可以写为：

(1) 设一“计数变量” N ，用于记录进行累加的成绩个数。显然， N 的初值为0，即 $N = 0$ ；

(2) 设“累加变量” A ，用于存放每一次加进去之后的成绩和。显然，未计算之前， A 的初值为0，即 $A = 0$ ；

(3) 输入一个学生的成绩给“成绩变量” C ；

(4) 将 C 和 A 的值相加，和放在变量 A 中，即 $C + A \rightarrow A$ ；（这个过程完成了将输入的学生成绩进行累加的操作）；

(5) 使 N 的值加1，即 $N + 1 \rightarrow N$ （表示已累加了一个学生的成绩）；

(6) 若 $N < 100$ 则返回(3)继续执行，否则执行(7)；（ $N < 100$ 说明第100个学生成绩未加到，必须返回(3)继续输入下一个学生成绩。否则， $N \geq 100$ ，即这100个学生成绩已累加，应执行步骤(7)）；

(7) 打印出总分 A 的值（此时 A 中存放的是这100个学生的成绩总分）；

(8) 算出 $\frac{A}{N}$ ，结果存入 P 变量，即 $\frac{A}{N} \rightarrow P$ （这里 P 变量表示平均分）；