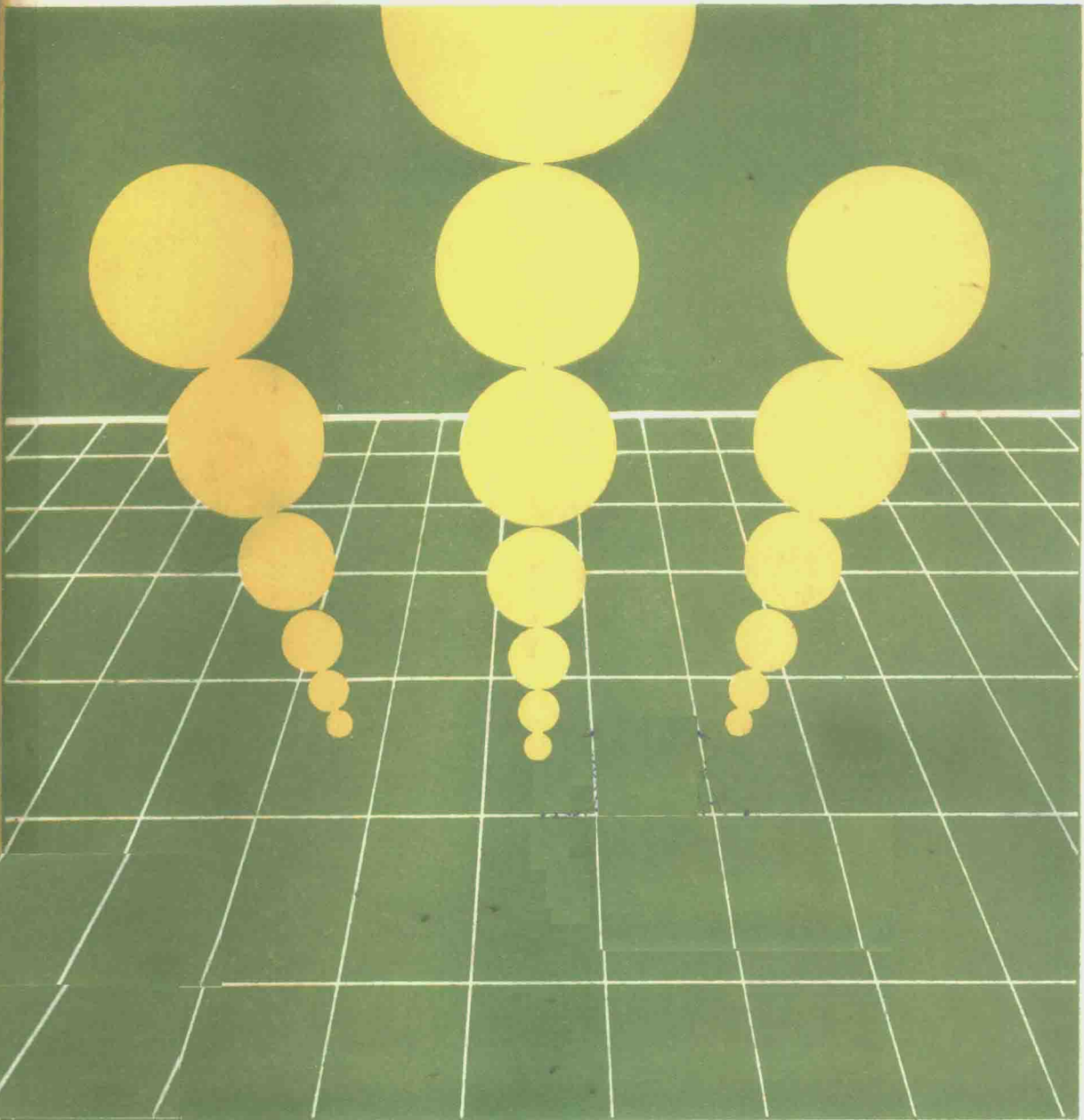


高等学校试用教材

计算机应用基础

(化学类专业用)

程光钺 陈庚华 忻新泉 编著



高等教育出版社

高等学校试用教材

计算机应用基础

(化学类专业用)

程光钱 陈庚华 忻新泉 编著

高等教育出版社

本书系根据 1985 年 4 月原教育部在杭州召开的理科化学系计算机教学学术讨论会上制订的“计算机应用基础(化学类专业用,四年制)”教学大纲编写的。本书分为 BASIC 语言、计算方法基础、应用三部分,共 14 章。书末有附录及参考文献。

本书 BASIC 语言部分的术语主要参考中华人民共和国国家标准 GB 4144-84(程序设计语言,最小 BASIC)。在这部分提供了足够的习题、思考题以辅助学习,同时举出足够的程序和相应的流程图,以便学习程序设计。在计算方法部分,侧重算法原理、程序设计原理和提供程序。对于数学基础稍差的读者也能了解原理,学会使用程序。在化学应用部分,广泛介绍了计算机解决化学各领域问题的情况,并介绍了利用数据库处理化学问题的捷径,开拓了读者的眼界。

根据目前各校的条件和要求不尽相同,本书内容分为必读部分和选读部分,选读部分加了*号。各校根据自己的情况选用。

本书适于化学类专业作教材,也可供其他专业和有关科技人员参考。

高等学校试用教材
计算机应用基础

(化学类专业用)

程光铎 陈庚华 忻新泉 编著

*

高等教育出版社出版
新华书店北京发行所发行
水利电力出版社印刷厂印装

*

开本 787×1092 1/16 印张 17 字数 388 000

1987 年 4 月第 1 版 1987 年 4 月第 1 次印刷

印数 00 001—12 150

书号 13010·01389 定价 2.50 元

前 言

本书是根据一九八五年四月原教育部制定的教学大纲编写的。按照教学大纲要求,“计算机应用基础”课程是四年制本科化学类专业学生的必修基础课。学时数为 50, 其中上机不少于 15 学时。考虑到现行教学计划安排和各校具体情况不同,本教材除大纲的基本内容外,还编写了部分打有星号(*)的较深入的章节,供各校选用。书中习题分为两类:A 类是基本习题,B 类是选作习题。

一九八六年三月,国家教委在杭州举办了化学类专业的“计算机应用基础”讲习班,使用本教材初稿作为讲义。有关笔者分别讲授了相应章节。与此同时,国家教委和高教出版社组织了由北京大学、复旦大学、中山大学、中国科技大学、浙江大学、上海交通大学和北京师范大学等校有关教授、专家参加的审稿会,对本教材的结构和内容,提出了许多宝贵的意见。本书笔者根据审稿会和讲习班的意见,又作了进一步的修改,编成本书。

全书分三部分,执笔者分别是:第一部分 BASIC 语言,陈庚华(浙江大学);第二部分计算方法,程光钺(四川大学);第三部分计算机在化学中的应用,忻新泉(南京大学)。

本书有关 BASIC 语言部分主要参考中华人民共和国国家标准 GB 4144-84(程序设计语言最小 BASIC),其中某些术语与目前常见的一些参考书略有不同。

本书在编写过程中得到各方面的支持、帮助和鼓励,在此表示衷心感谢。我们特别感谢参加本书审稿会的张懋森教授、陈志行教授、韩世钧教授、范勋培副教授、俞蒙槐副教授、田梦庐同志、朱自刚同志、刘万琪同志;以及对本书给予大力支持的陈祖福同志。

由于我们水平有限,加以时间匆促,书中难免有缺点错误,敬请读者批评指正。

目 录

绪论	1
一 计算机系统简介	1
二 计算机程序设计语言	3
三 计算机应用简介	4
四 计算机的发展	4

第一部分 BASIC 语言

第一章 BASIC 语言的基本成分与结构	6
§ 1.1 BASIC 程序的特点	6
§ 1.2 基本字符	7
§ 1.3 常数	8
1.3.1 数值常数	8
1.3.2 串常数	8
§ 1.4 变量	8
1.4.1 数值变量	9
1.4.2 串变量	10
§ 1.5 函数	10
1.5.1 标准函数	10
1.5.2 用户定义函数	12
§ 1.6 表达式	12
1.6.1 数值表达式	12
1.6.2 串表达式(string expression)	13
1.6.3 逻辑表达式(logic expression)	13
* § 1.7 串函数	15
1.7.1 取出字符串部分字符的函数	15
1.7.2 测字符串长度函数(LEN 函数)	16
1.7.3 数值与字符串相互转换函数	16
习题	17
第二章 赋值、输入输出与读数据语句	19
§ 2.1 引例——溶液 pH 值的计算	19
§ 2.2 赋值语句(LET 语句)	20
2.2.1 语法形式	20
2.2.2 功能	20
§ 2.3 输入语句(INPUT 语句)	21
§ 2.4 读语句和数据语句(READ 语句和	

DATA 语句)	22
2.4.1 语法形式	22
2.4.2 特点	22
§ 2.5 再置语句(RESTORE 语句)	23
§ 2.6 打印语句(PRINT 语句)	24
2.6.1 语法形式和功能	24
2.6.2 打印格式	25
§ 2.7 输入输出的程序设计	30
习题	32
第三章 控制语句	36
§ 3.1 引例——爆炸极限	36
§ 3.2 转移语句(GOTO 语句)	37
§ 3.3 条件语句(IF-THEN 语句)	39
3.3.1 语法形式	39
3.3.2 逻辑运算符的应用	42
* § 3.3.3 IF-THEN-ELSE 语句	44
§ 3.4 暂停语句(STOP 语句)	44
§ 3.5 开关语句(ON-GOTO 语句)	44
§ 3.6 子程序	45
§ 3.7 注释语句(REM 语句)	47
§ 3.8 控制语句的程序设计	47
习题	50
第四章 循环	54
§ 4.1 引例	54
§ 4.2 维数语句(DIM 语句)	55
§ 4.3 循环语句和出口语句 (FOR-NEXT 语句)	56
4.3.1 语法形式	56
4.3.2 循环结构的特点	59
§ 4.4 多重循环	60
§ 4.5 循环的程序设计	65
习题	70
第五章 图形输出	74
§ 5.1 文本方式下的图形显示	74

§ 5.2	图形方式与屏幕坐标	75
5.2.1	图形方式	75
5.2.2	屏幕坐标	76
§ 5.3	图形语句	77
5.3.1	图形显示语句	77
5.3.2	颜色语句	77
5.3.3	画点语句	77
5.3.4	画线语句	78
§ 5.4	图形输出的程序设计	81
习题		85
第六章	程序的调试与考核	86
§ 6.1	程序的静态检查	86
6.1.1	编程中的几个问题	86
6.1.2	静态检查	88
§ 6.2	动态调试	90
6.2.1	分段调试与整体调试	90
6.2.2	常见错误	90
6.2.3	程序的修改	90
§ 6.3	程序的考核	90

第二部分 计算方法基础

第七章 计算方法与程序设计的质量要求

§ 7.1	计算量	93
§ 7.2	存储量	95
§ 7.3	误差	97
§ 7.4	程序结构	99
§ 7.5	综合示例: 几种排序方法的比较	99
7.5.1	遴选法排序	99
7.5.2	遴选互换法排序	103
7.5.3	互换法排序	105
7.5.4	下沉法排序	106
7.5.5	小结	108
习题		109

第八章 方程求根

§ 8.1	根的初值和存在范围	110
8.1.1	根据方程的数学性质进行判断	110
8.1.2	根据方程的物理意义进行估计	111
8.1.3	图解法	111
8.1.4	逐步法	112

§ 8.2	二分法	112
8.2.1	方法概述	112
8.2.2	二分法的框图与程序	113
8.2.3	应用示例: 实际气体的状态方程	116
§ 8.3	迭代法	117
8.3.1	迭代法的基本思想	117
8.3.2	迭代法的几何意义	117
8.3.3	迭代法的收敛条件	118
8.3.4	迭代法的框图与程序	120
8.3.5	应用示例: 弱酸溶液的 pH 值	121
8.3.6	应用示例: 沉淀的同离子效应	123
* § 8.4	牛顿法	124
8.4.1	牛顿法的原理	124
8.4.2	牛顿法的几何解释	125
8.4.3	对牛顿法的几点注释	126
8.4.4	牛顿法的框图与程序	127
8.4.5	应用示例: 化学反应的平衡浓度	129
8.4.6	应用示例: 络合平衡计算	130
习题		131

第九章 线性方程组求解

§ 9.1	简单消去法	132
9.1.1	简单实例	132
9.1.2	增广矩阵	133
9.1.3	消元过程及其在计算机上的实现	134
9.1.4	回代过程及其在计算机上的实现	137
9.1.5	简单消去法的程序	138
9.1.6	应用示例: 蒸气压-温度公式的计算	139
* § 9.2	主元消去法	141
9.2.1	一般介绍	141
9.2.2	三种主元消去法	142
9.2.3	列主元消去法的框图	144
9.2.4	列主元消去法的程序	146
9.2.5	应用示例: 混合物组分的化学分析	146
习题		149

第十章 插值

§ 10.1	拉格朗日一元全节点插值	151
10.1.1	一般介绍	151
10.1.2	一元全节点插值的算法	152
10.1.3	一元全节点插值的程序	153
10.1.4	应用示例:	

	热容-温度数据的插值之一	154
* § 10.2	拉格朗日一元部分节点插值	155
10.2.1	问题的提出	155
10.2.2	一元部分节点插值的算法与框图	155
10.2.3	一元部分节点插值的程序	158
10.2.4	应用示例: 热容-温度数据的内插之二	159
10.2.5	应用示例: 一维数表的插值	161
* § 10.3	拉格朗日二元插值	162
10.3.1	二元插值的算法	162
10.3.2	二元插值的框图与程序	163
10.3.3	应用示例: 二维数表的插值	164
	习题	165
第十一章	拟合	167
§ 11.1	一元线性最小二乘法	167
11.1.1	一般介绍	167
11.1.2	一元线性最小二乘法的算法	168
11.1.3	相关系数	171
11.1.4	一元线性最小二乘法的程序	174
11.1.5	应用示例: 化学反应活化能与压力关系	175
* § 11.2	多元线性最小二乘法	175
11.2.1	多元线性最小二乘法的算法	175
11.2.2	多元线性最小二乘法的框图	177
11.2.3	多元线性最小二乘法的程序	178
11.2.4	应用示例: 气体混合物的粘度	180
11.2.5	权与加权线性最小二乘法	182
* § 11.3	非线性拟合简介	185
11.3.1	化为线性拟合的情况	185
11.3.2	应用示例: 气体反应动力学方程的建立	186
11.3.3	不化为线性拟合的情况	189
	习题	190
*第十二章	积分与微分方程的数值解	191
§ 12.1	梯形法求积	191
12.1.1	一般介绍	191
12.1.2	梯形法的算法	192
12.1.3	梯形法的框图与程序	192
12.1.4	应用示例: 圆周率 π 的计算之一	193
§ 12.2	辛普森法求积	195

12.2.1	辛普森法的算法	195
12.2.2	辛普森法的框图与程序	197
12.2.3	应用示例: 圆周率 π 的计算之二	198
§ 12.3	离散点下的求积	199
12.3.1	方法介绍	199
12.3.2	应用示例: 圆周率 π 的计算之三	199
12.3.3	应用示例: 等压加热过程的热效应	201
12.3.4	应用示例: 化学反应的反应热效应	202
§ 12.4	尤拉法求微分方程组数值解	204
12.4.1	一般介绍	204
12.4.2	尤拉法的算法	205
12.4.3	一阶微分方程组的数值解	207
12.4.4	高阶微分方程与方程组的数值解	208
12.4.5	尤拉法的框图与程序	209
12.4.6	应用示例: 放射性元素的衰变	211
§ 12.5	龙格-库塔法求微分方程组数值解	213
12.5.1	龙格-库塔法的算法	213
12.5.2	龙格-库塔法的程序	213
12.5.3	应用示例: 化学反应的动力学计算	217
	习题	220

第三部分 应用

第十三章	计算机在化学中的应用概况	222
§ 13.1	计算机与化学	222
13.1.1	边缘学科的发展	222
13.1.2	计算机对化学的影响	222
13.1.3	计算机化学使用的语言问题	223
13.1.4	模拟与模型	224
§ 13.2	数值运算在化学中的应用	225
13.2.1	实验数据处理	225
13.2.2	无机化学中的应用	226
13.2.3	分析化学中的应用	226
13.2.4	物理化学中的应用	227
13.2.5	有机化学中的应用	229
13.2.6	高分子化学中的应用	229
13.2.7	理论化学中的应用	230
13.2.8	结构化学中的应用	230
13.2.9	谱学中的应用	230
13.2.10	其它方面的应用	230
§ 13.3	非数值运算在化学中的应用	231

13.3.1	化学数据库	231	14.2.5	使用 dBASE II 时的注意事项	243
13.3.2	化学教学	231	§ 14.3	化学数据库的建立和使用	243
13.3.3	图形与化学	233	14.3.1	建立 dBASE II 数据库的 一般过程	243
13.3.4	化学情报检索	234	14.3.2	定义化学元素性质数据库结构	243
13.3.5	合成化学	234	14.3.3	数据的添加、删除和插入	245
13.3.6	人工智能与专家系统	235	14.3.4	数据的修改	246
13.3.7	文字处理	235	14.3.5	数据的检索和排序	246
§ 13.4	实验室自动化	235	14.3.6	读数和打印输出	247
13.4.1	联机	235	14.3.7	计数和求和	248
13.4.2	接口	236	§ 14.4	物性数据库	248
13.4.3	采集数据	236	14.4.1	建立 VAPOUR 数据库	248
13.4.4	单板机	237	14.4.2	应用程序文件	249
§ 13.5	化学中应用小结	237	14.4.3	应用程序(计算化合物蒸汽压)	250
*第十四章	化学数据库与 dBASE II	239	§ 14.5	文献库	251
§ 14.1	数据库	239	§ 14.6	dBASE II 的发展版本	252
§ 14.2	dBASE II 简介	240	附录一	APPLE-II 上机操作	253
14.2.1	dBASE II 命令语句	241	附录二	IBM-PC 上机操作	257
14.2.2	dBASE II 表达式	241	附录三	ASCII 代码表	263
14.2.3	文件类型	242	参考文献		264
14.2.4	dBASE II 的功能	242			

绪 论

电子计算机的问世是二十世纪最重大的科技成就之一。它极大地增强了人类认识世界和改造世界的能力,并正在深刻地影响着人类社会的各个领域。目前电子计算机的生产、普及和应用程度,已成为衡量一个社会现代化水平的重要标志之一。

根据工作原理,电子计算机分两大类,一类是电子数字计算机(digital computer)另一类是模拟计算机(analog computer)。前者是目前广泛使用的计算机,它以数字形式的量值在机内进行运算;后者则以连续变化的物理量(如电流、电压)表示被运算量。

按使用目的,电子计算机又可分通用电子计算机(general purpose computer)和专用计算机(special purpose computer)。前者是为处理范围广阔的各种问题而设计的计算机,后者是为处理某类专门问题而设计的计算机。

本书所讨论的是通用电子数字计算机,以下简称计算机。

一、计算机系统简介

一个完整的计算机系统是由硬件(hardware)和软件(software)组成。

1. 硬件:它是数据处理的实体设备。硬件主要由五部分组成:运算器、控制器、存储器、输入和输出设备,它们之间的相互关系示于图1。

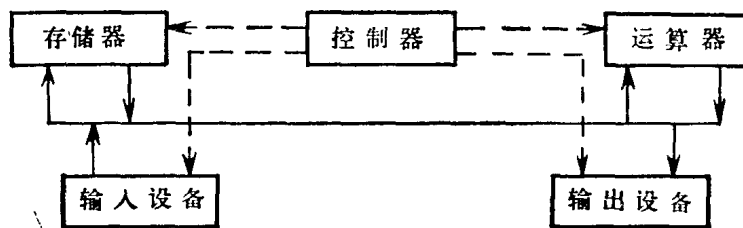


图1 计算机硬件结构

——> 数据信息流向
---> 控制信息流向

下面简要介绍硬件的各主要部分。

(1) 存储器(storage, memory): 存储器是能把数据放入、保存并可从中取出数据的计算机部件, 其中的各种数据都是以二进制存放的。存储器分成一个个存储单元, 每个存储单元有编号, 称为单元地址。一个存储单元可存放一组二进制的数位。二进制数制中的一个数字位(0 或 1), 称为一个比特(bit)。一般, 计算机是以 8 位二进制数位为一个字节(byte), 以 B 表示。1024 个字节($1024=2^{10}$) 称为 1K 字节, 以 1KB 表示。1000KB 为 1 兆字节, 以 1MB 表示。在存储器中所能容纳的数据量, 称为存储容量(storage capacity), 其单位是字节(B)。存储容量是评价一台计算机性能的主要指标之一。如一台存储容量为 8 位 64K 的 APPLE-II 机, 它可存储 65536

字节($=64 \times 1024$),即 8×65536 个二进制数位。一般以 64KB 表示该机的存储容量。

存储器分两类:一类是内存储器,简称内存;另一类是外存储器,简称外存。

内存的元件一般有磁芯和半导体两种。早期为磁芯,现已用半导体。半导体存储器分随机存储器(Random Access Memory 简称 RAM)和只读存储器(Read Only Memory 简称 ROM)。RAM 能直接存取数据,而 ROM 只能读不能写^①。不同计算机的 RAM 和 ROM 的容量是不同的。如 IBM-PC 机, RAM 是 64KB(可扩充到 640KB), ROM 是 40KB(可扩充到 256KB)。APPLE-II 机的 RAM 是 48K, ROM 是 12K。

外存的容量比内存大,但存取速度慢。常用的外存有磁带、磁盘等。

存储器的一个重要特点是新的信息没有存入前,原有的信息能反复读取,不会被破坏。一旦新的信息送入,原有的信息立即消失。

(2) 运算器(Arithmetic Logic Unit, 简称 ALU), 又称算术逻辑部件。算术运算是按算术规则进行运算, 如加、减、乘、除等。逻辑运算是根据逻辑规则进行运算(有关逻辑规则将在第一章 § 1.6 节介绍)。

运算速度是计算机的又一性能指标。通常有以下几种表示法:每秒加法的次数,每秒存取的字节数,机器执行一条指令所需的时间(又称指令周期)等。例如,APPLE-II 机的运算速度是每秒做 50 万次加法,银河计算机的运算速度是每秒一亿次。

(3) 控制器(control): 控制器是整个计算机的指挥系统。它通过向机器的各个部分发出控制信息来指挥机器自动地、协调地工作。控制器是根据指令(instruction)指挥机器工作的。指令是规定机器的操作类型、操作数地址的一组代码。指令包括两部分:一是指明所进行的是何种操作,称为操作码(operation code),二是指出操作数在机内存放的位置即地址,称为地址码(address code)

人们用计算机进行计算或数据处理,事先应编好程序(program)。一个程序是由一系列指令组成。因此,用计算机解题的过程,实际上是执行指令的过程。具体地说,把解题所需要的程序(即一系列指令)和数据送入存储器,计算机依次从给定的存储单元取出指令、执行指令,然后取下一条指令再执行。如此周而复始地进行,直至程序执行结束,计算结果也就求得了。

运算器和控制器合称为中央处理机(Central Processing Unit, 简称 CPU), 这是计算机的核心。CPU 有多种型号,如 APPLE-II 微型机的 CPU 型号是 6502。IBM-PC 是 Intel 8088。

(4) 输入与输出设备(Input/Output Device, 简称 I/O)。输入输出设备是用来将数据送入计算机或从计算机取出数据的设备。各种输入和输出设备统称外围(或外部)设备。主要的输入设备有:卡片阅读机和纸带阅读机(早期使用)、键盘、模/数转换器(它能将连续变化的模拟量如电压,转换成数字量)、光笔等。

常用的输出设备有:显示器、各种打印机、绘图仪、数/模转换器等。

^① 将数据长期地或暂时地记录在存储器中称为写或存,从存储器获得数据称为读或取。

很多外存储器如磁盘驱动器、磁带机等,同时具有输入输出的功能。

2. 软件:软件是相对于硬件而言的,它包括机器运行所需要的各种程序及其有关资料。软件分两大类:系统软件(system software)和应用软件(application software)。

系统软件是用来提高计算机的使用效率、扩大计算机的功能、简化程序设计和使用方法的整套程序。如编译程序和解释程序(在下面讨论),操作系统(把硬件、软件和用户的程序进行统一管理和调度,使整个计算机协调工作等),诊断程序(检查硬件各部分是否可靠?有否故障及故障部位等),控制程序等。系统软件具有公用性。

应用软件一般指的是针对某类应用性问题而建立的程序系统。应用软件常常是在系统软件的基础上设计的,要在系统软件的支持下运行。

软件的发展十分迅速,五十年代软件还仅仅作作为一种工艺,而今,软件已形成为一门新兴的产业部门。

软件和硬件是相辅相成的两个方面。一台性能优良的计算机,不仅要有可靠而先进的硬件,还必须具有高质量的软件系统。

二、计算机程序设计语言

前已指出,在使用计算机前,人们须将编好的程序输入计算机内,计算机才能按程序的安排进行工作。对程序的设计、书写和调试称为程序设计。表达程序要通过一定的语言(language)。程序设计语言就是为了表达计算机程序而制定的语言。

1. 机器语言和汇编语言:早期所用的程序设计语言与某台或某类计算机的结构有密切关系。这种语言称为低级语言(low-level language),又称面向计算机语言(computer-oriented language)。最初人们直接用机器指令编写程序。其指令都是机器指令的低级语言称为机器语言(machine language)。用机器语言编写的程序在计算机上能直接执行。但它的最大缺点是程序全由一大堆数字所组成,不直观、不形象,难记、难懂,不易交流。此外,不同的计算机也互不通用,这对计算机的普及和应用带来很大障碍。

后来人们把机器语言符号化,用较形象和直观的符号(如用英文单词的词头,像ADD表示加,SUB指减等)代替各种不同的机器指令,产生了所谓汇编语言(assembly language)。用这种语言编写程序(简称编程)比机器语言程序有了改进。但因它仍是面向计算机的语言,对不同的计算机仍互不通用,同时编程难度大,对不甚熟悉计算机的人们使用困难。为了克服这些缺点,使计算机为广大用户所掌握,五十年代起,发展了另一种程序设计语言,即所谓高级语言(high level language)。

2. 高级语言:高级语言是不反映某台或某类计算机结构的程序设计语言。使用高级语言编写程序,不必了解具体的计算机硬件结构与指令系统。用高级语言编程和调试比用机器语言和汇编语言要方便和容易得多。这种语言常与人们习惯用的自然语言(如英文)和数学语言相类似,例如用词义相近的英文单词对算法进行描述,直接使用一些数学运算符号或代数式等。因此,高级语言形象、直观,便于记忆和掌握。

目前已有的高级语言为数众多,其中使用最广泛的有FORTRAN、BASIC、PASCAL、

COBOL、ALGOL 等。每种高级语言常常有多种版本。

由高级语言编写的程序，称为源程序(source program)。当源程序输入机内，计算机不能直接识别，故无法执行。这时需要有个翻译程序(translating program)，自动地将源程序翻译成机器语言表示的目标程序(object program)，这时机器才能执行。

翻译程序有两种类型，一种是编译程序(compiler)，它把用高级语言编写的整个源程序，翻译成用机器指令表示的目标程序，然后执行，获得结果。FORTRAN、PASCAL、COBOL、ALGOL 等语言多采用此类方式。另一种是解释程序(interpreter)，它对源程序进行逐句翻译，边翻译边执行。BASIC 目前多采用此类方式。用汇编语言编写的源程序，在执行前也要逐句翻译成目标程序后才能执行。这种翻译程序称为汇编程序。

编译程序和解释程序各有特点，前者运算速度快，但占内存多。后者则反之。

三、计算机应用简介

计算机的应用现已从早期的单纯数值计算发展到数据处理、自动控制以至社会生活的各个领域。下面就计算机应用的主要三个方面作些简要介绍。

1. 科学和工程计算：由于计算机具有运算速度快、运算正确可靠、结果精度高等优点，它已成为在现代科学和工程中最有力的一种计算工具。如人造卫星运行轨道的计算，原子和分子结构的计算，石油、化工、建筑、水电、轻工以及交通运输等部门中各种工程设计的计算等等，几乎都是由计算机完成的。

2. 数据处理：在现代计算机应用领域中，数据处理约占 80%。这里所指的数据具有广义的含义，它除了一般的数字数据外，还包括各种文字、图表、文献、数学模型、共性规律等。如实验数据的处理，经济信息、企业管理、天文、气象、地质勘探等等的数据处理。又如数据库、文献和情报检索(将在第十一、十二章讨论)、各种模拟(如化学模拟、化工模拟、模拟计算等)、数字仿真、汉字信息处理、计算机辅助教学等等，都属数据处理的范畴。

3. 自动控制：用计算机控制生产过程，能使生产过程从局部自动化向综合自动化发展。若配以必要的输入输出设备与计算机联用，如模拟量输入输出装置、开关量输入输出装置等，就可实现对生产过程的检测、监视和控制，实现优质、高产和低耗。邮电系统用计算机能实现自动分拣包裹，转发电报、接转电话等。各种仪器通过数/模和模/数装置，能实现仪器的联机操作，如色谱-质谱仪，当样品进入后，就能定量计算出样品中有关物质的组成等数据。

四、计算机的发展

计算机自诞生以来，发展十分迅速。仅仅四十年左右时间，已经历了四代。各代主要特点简述如下：

第一代：计算机的基本元件是电子管。这时的计算机庞大笨重、运算速度慢、稳定性和可靠性差。软件方面采用低级语言，应用以科学计算为主。

第二代：由晶体管取代了电子管，注重输入输出设备。开始用高级语言。应用转向数据处理。

第三代：采用集成电路为基本元件，使计算机实现了小型化。运算速度、稳定性和可靠性大大提高。外围设备的品种和数量显著增加。高级语言迅速发展和普及。应用已遍及科学、工程

计算、数据处理和自动控制。

第四代:全面采用大规模或超大规模集成电路为基本元件,从而使计算机体积大大减小,出现了微型计算机并得到蓬勃发展。软件与硬件开始结合,应用进入社会的各个领域。

目前,计算机的发展趋势主要有以下几个方面:

1. 巨型计算机:这种计算机并不是指它笨重庞大,而是指其内存容量大、运算速度快。如银河计算机,每秒运算速度在一亿次以上。这类计算机主要用于军事技术和尖端科研。

2. 微型计算机(简称微机,microcomputer):它是由大规模或超大规模集成电路组成。微机有一定功能,还具有体积小、价格便宜、可靠性高、便于推广和应用等优点。它几乎能用于国民经济的各个部门,乃至家庭和日常生活之中。

3. 计算机网络:它是利用通讯系统把设在各地的计算机连接起来,并常以其中的一台计算机作为中心。使用时只要有一台终端设备,如打印机或荧光显示装置,通过通讯系统与网络连接;这样,用户在办公室或家里就能使用网络中的计算机。

4. 人工智能:计算机的功能归结为一点,就是代替人的脑力劳动和体力劳动。人工智能是用计算机模仿人的感觉和大脑功能,即所谓智能模拟。目前,这方面的工作已有了相当的进展。如机器人、对语言的识别、数字仿真、专家系统等等。

5. 光电子计算机:近年来,正在研究用光导管代替传统的计算机电子元件。这种计算机一旦研制成功,对计算机的发展又将产生一个飞跃。

第一部分 BASIC 语言

第一章 BASIC 语言的基本成分与结构

第一个 BASIC 语言版本是 1965 年发表的,现已有多种版本。但从功能考虑,大体可分两大类:基本 BASIC 和扩充 BASIC。各种类型计算机所配的扩充 BASIC 都是在基本 BASIC 基础上作了扩充和提高,以便简化操作,提高计算功能。

本书是以 APPLE-II 机为主要对象,着重介绍这类机型所配的扩充 BASIC 语言及其程序设计。

本章讨论 BASIC 程序的特点与结构、BASIC 语言的词法——基本字符、常量与变量、函数、表达式。

§ 1.1 BASIC 程序的特点

与其它高级语言比较,BASIC 语言具有如下特点:简单易学、使用方便,可进行人机对话,有丰富的文字处理功能和多种打印格式,能适用于一般的科学计算、数据处理和管理等要求。对初学者尤为适宜^①。因此,在一般小型计算机或微机中都配有这种语言,成为国际上较为普及和流行的高级语言之一。

例 1.1 根据方程

$$V = \frac{nRT}{p}$$

编写计算体积 V 的 BASIC 程序。已知: $n=1\text{mol}$, $T=273.15\text{K}$, $p=101.325\text{kPa}$ 。

解: 下面是计算 V 的两个程序,它们是等价的。

程序 1

```
10 INPUT T,P
20 LET R=8.314
30 LET N=1
40 LET V=N*R*T/P
50 PRINT V
60 END
```

程序 2

```
10 INPUT T,P
20 R=8.314:N=1:V=N*R*T/P
30 PRINT V
40 END
```

^① BASIC 是 Beginner's All-purpose Symbolic Instruction Code 的缩写。意为初学者通用符号指令代码。

当执行上述程序后,分别输入 T 和 p 的具体值: 273.15 和 101.325, 机器即显示出 V 的计算值 22.4127224。在上述程序中, 程序 1 属基本 BASIC, 程序 2 属扩充 BASIC。显然, 程序 2 比程序 1 更简练。

扩充 BASIC 的主要特点是:

1. 一个 BASIC 程序是由行(又称程序行)的序列所组成, 用行号(line number)对行进行编序。除特殊规定外, 一般程序从第一行开始按序列顺序执行。

2. 每行以一个行号开始。行号是非零的正数, 它是由小到大顺序排列的。在编写程序时, 通常行号不宜连续, 便于中间增补。每种机器允许的最大行号是不同的。如 APPLE-II 机的行号最大是 63999。

每行可写一个或多个语句, 语句间用冒号分隔。

3. 语句包括关键字和语句体两部分。关键字(key term)通常是常用的英文拼写词或助记词的一个字符串(character string), 它给语句或语句成分以特有的标识。如 LET 表示赋值, INPUT 是键盘输入, PRINT 是输出, END 是结束。常用的关键字已列于表 1-1。

表 1-1 主要关键字

关 键 字	说 明	关 键 字	说 明
LET	赋值	SUB	子程序
INPUT	键盘输入(键入)	GOSUB	转于程序
READ	读	RETURN	返回
DATA	数据	ON	关于、当
RESTORE	恢复	DIM	数组说明
PRINT	输出	FOR	循环入口
DEF	用户定义函数	NEXT	循环出口
GO	去	STEP	步长
GOTO	转移	REM	注释
TO	到	STOP	暂停
IF	如果	END	结束
THEN	则		

语句体是计算机执行的具体内容, 如 $N=1$, 表示将 1 赋值给变量 N , 在框图中有时表示为 $1 \Rightarrow N$ 。应指出, 有些语句没有语句体。如程序 1 中 60 行的结束语句 END, 无语句体。

4. 当用户在键盘上写好一行后, 必须键入回车键。这样, 该行的内容才进入机内, 否则无效。回车键的键符随机而异, 如 APPLE-II 机是 RETURN, IBM-PC 机是 ↵。一般书写用“↵”表示。为简化书写, 在一般书(包括本书)中所有程序内的回车符“↵”均略去。

§ 1.2 基本字符

字符(character)是用来组织、控制或表示数据的字母、数字和计算机能识别的其它符号。多

数版本的扩充 BASIC 语言具有以下一些字符:

字母: 由 A~Z 26 个英文字母组成。其它字母如 π 、 α 、 β 、 γ 等不能作为 BASIC 字母。

数字: 0~9 共十个十进制数。

串字符: 如 + (加)、- (减)、* (乘)、/ (除)、^ (乘方)、< (小于)、= (等于)、> (大于)、" (引号)、; (分号)、: (冒号)、\$ (货币号)、空格、? (问号) 等。

还有一些字符将在以后有关章节陆续介绍。

§ 1.3 常 数

常数是其值固定不变的量。BASIC 中的常数有两种类型: 数值常数(numeric constant)和串常数(string constant)。

1.3.1 数值常数

数值常数简称数常数, 其值为数值。数常数又可分为整常数和实常数。

一、整常数(integer constant)即整数。整数的数值范围对 APPLE-II 机是 -32767~32767。在这数值范围之外, 机器不接受。

二、实常数(real constant)即实数。化学中绝大多数物性常数是实数, 如气体常数 $R=8.314 \text{ J} \cdot \text{mol}^{-1} \cdot \text{K}^{-1}$ 。玻尔兹曼常数 $k=1.380626 \times 10^{-23} \text{ J} \cdot \text{K}^{-1}$ 等。实数的数值范围和有效位数是随机而异的。如 APPLE-II 机的数值范围一般为 $3.0 \times 10^{-39} \sim 1.7 \times 10^{38}$, 小于 3.0×10^{-39} 按零处理, 大于 1.7×10^{38} 给出“溢出”(overflow)信息。实数在 APPLE-II 机中的有效位数是 9 位。

常数也可用科学计数法表示。科学计数表示又称比例表示, 其形式为:

Sd.....d.d.....dESd...d (称为显式点比例表示)

或

Sd.....dESd...d (称为隐式点比例表示)

其中 d 是十进制数字; S 是正负号(正号可省略); E 是字符 E, 它代表“乘 10 的幂”。如 1.380626×10^{-23} 和 101325 用科学计数法表示为 $1.380626\text{E}-23$ 和 $1.01325\text{E}+5$ (或 $1.01325\text{E}5$)。应注意 E 后的部分(称为指数部分)不能单独成为一个数。如 100 不能写成 E2, 应写成 1E2。对 APPLE-II 机, 指数部分最多只能有两位数。

1.3.2 串常数

串常数又称字符串(string)常数。它是用双引号括起的字符序列。如 "KOH"、"KMNO4" 等都是串常数。串常数的值是字符串, 它以双引号之间所有字符(包括空格)构成的字符串作为其值。在双引号之间的字符数目, 称为串常数的长度。如上面两个串常数的长度分别是 3 和 5。串常数的长度仅以一行的长度为限。例如 APPLE-II 机的串常数的允许长度是 256。

§ 1.4 变 量

变量(variable)是在程序执行中其值可变的量。每个变量都须有一个名字, 称为该变量的变量名。

在 BASIC 中的变量有两类:数值变量和串变量。它们又有简单变量和下标变量之分。

1.4.1 数值变量

数值变量简称数变量(numeric variable)。

数变量的值为数值。物质的性质在计算机内常可表示为数变量。例如沸点,若以数变量 TB 表示(TB 是该数变量的变量名),则不同物质在不同条件下其值可以不同。如水在 101.325 kPa 时 TB 的值(单位为 K)是 373.15;在 122.877 kPa 时 TB 的值是 323.15 等等。

数变量有两种:简单数变量和下标数变量。

一、简单数变量:它的命名规则是变量名由一个或两个字母、或由一个字母尾随一个数字来命名。如 A、A1、AA、YE、YC 等都是可允许的合法的变量名。变量名若多于两个字符,则只有前两个字符有效^①。如 TE1 和 TE2, ABCD 和 AB 被视为同名的变量。同名变量其值相同。

凡是不符合上述命名规则的变量名都是不允许(不合法)的,如

$$5Y, X-5, \pi$$

等都不能作为简单数变量名。APPLE-II 机规定,关键字也不能作为简单数变量名^①。

简单数变量的值可以是整常数或实常数。APPLE-II 规定:对整常数,简单数变量尾随%。例如 X%, F2%, KK% 等,它们的值都是整常数。实常数没有尾随%。

二、数组和下标变量:简单数变量只能一个一个命名。当一个源程序中变量很多时,仅用简单数变量不仅繁琐不便,有时还会出现变量名不够用或其他问题。如有 50 个温度数据,用简单数变量要写 50 个变量名。若以 T1, T2, ..., T50 表示,则不仅不便,而且其中 T1, T10, T11, ... 都是同名变量,无法彼此区分。这时我们最好把这些温度数据当作一组有序的整体,即当作一个数组,而把每个温度数据当作这个数组中的一个元素。

1. 数组(array):数组是按一定格式排列起来的一组同一属性的量。如 50 个温度数据或变量是属同一性质的量(其值为实数或整数的变量)。它们组成数组 T, 其中每个量依次表示为 T(0), T(1), ..., T(49)。又如

向量

$$Z = (Z_0, Z_1, \dots, Z_n)$$

矩阵

$$A = \begin{pmatrix} a_{00} & a_{01} & \dots & a_{0n} \\ a_{10} & a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots & \dots \\ a_{m0} & a_{m1} & \dots & a_{mn} \end{pmatrix}$$

等都是数组,可分别用 Z 和 A 表示。这里的 T、Z 和 A 都是数组名,其命名规则与简单数变量相同。

2. 下标数变量:数组中的每个元素称为数组元素,又称下标变量(subscripted variable)。下标变量的表示方法是在数组名之后缀以用圆括号括起来的下标。若有两个下标,则在其间用逗号隔开。例如上述数组 Z 的各下标变量表示为

$$Z(0), Z(1), \dots, Z(N)$$

^① IBM-PC 机变量名可多于两个字符,关键字也可作变量名。