外语院系适用

计算机辅助外语教学
BASIC 编程基础

陈庆昌 编

上海交通大学出版社

内 容 提 要

本书介绍BASIC语言编程基础，着重于与外语教学及研究有密切关系的字符串处理。各章节的内容安排循序渐进、由简到繁。本书可作为外语院系学生的教科书，也可作为外语工作者初学BASIC编程的入门书。书中的程序均可直接在APPLE-Ⅱ上运行。

计算机辅助外语教学
BASIC编程基础

上海交通大学出版社出版
（淮海中路1984弄19号）
新华书店上海发行所发行
上海交通大学印刷厂印装

开本 787×1092毫米 1/32 印张 9 字数198,000
1987年12月第1版 1987年12月第1次印刷
印数：1—5,000
ISBN7-313-00032-4/TP31 科技书目·154-285

定价：1.50元

目　录

第一章　计算机简介

§1·1　计算机组成……1
§1·2　计算机语言……3

第二章　扩展 BASIC

§2·1　程序及语句……6
§2·2　程序的键入……9
§2·3　键盘操作命令……11
§2·4　调试……16
习题一　……18
§2·5　数据……19
§2·6　PRINT 语句(一)……24
§2·7　常量、变量、表达式……26
§2·8　LET 语句……31
§2·9　语句命令……37
§2·10　PRINT 语句(二)……38
习题二　……43
§2·11　INPUT 语句……46
习题三　……53
§2·12　GOTO 语句……54
§2·13　IF 语句……56

习题四 ……70
§2·14 END语句…… 71
§2·15 子字符串、字符串函数、LEN 函数……72
习题五 ……92
§2·16 TAB 函数、SPC 函数、PRINT 语句(三)
……93
§2·17 VTAB语句、HTAB语句、POS(X)
函数……97
§2·18 单下标变量、DIM 语句…… 103
习题六 …… 110
§2·19 FOR 循环结构 …… 111
§2·20 多重循环…… 123
习题七 …… 139
§2·21 READ语句、DATA语句、RESTORE
语句…… 141
§2·22 VAL 函数、STR$ 函数…… 149
§2·23 CHR$ 函数、ASC 函数…… 156
习题八 …… 164
§2·24 GOSUB 语句、RETURN 语句…… 165
§2·25 逻辑运算符 NOT、AND、OR…… 176
§2·26 ON 语句…… 181
习题九 …… 183
§2·27 标准函数…… 183
习题十 …… 192
§2·28 GET 语句 …… 194
§2·29 双下标变量…… 203
习题十一 …… 208

§2·30 REM语句…… 209
§2·31 ONERR GOTO语句、RESUME语句… 209
§2·32 STOP 语句…… 212
§2·33 TRACE 语句、NOTRACE语句…… 213
§2·34 DEF 语句…… 216
§2·35 PEEK 函数、POKE 语句…… 219
§2·36 CALL 语句…… 224

第三章 文件

§3·1 盘片的格式化…… 229
§3·2 文件名…… 231
§3·3 程序文件…… 232
§3·4 有关文件的其他 DOS 命令…… 234
§3·5 顺序文件…… 236
§3·6 顺序文件的扩充、修改与读出…… 240
§3·7 随机文件…… 250
附 录 练习参考答案…… 255

第一章　计算机简介

§ 1.1　计算机组成

计算机一般由运算器、控制器、内存贮器、输入和输出设备等五个基本部分构成。其中运算器和控制器又合称为中央处理器(central processing unit，简称 CPU)，这是一种大规模集成电路芯片，包含着指令系统，是计算机的“大脑”，以它为核心，加上由大规模集成电路做成的内存贮器和输入、输出接口 (interface) 等，构成了计算机的主机。以主机为中心，配以各种不同的输入、输出设备、外存贮器以及其他的外围设备就构成硬件 (hardware)系统。

计算机的内存贮器 (internal storage)，简称存贮器或内存，是计算机的“记忆器” (memory)，供存放数据和程序。它能(以二进制的形式)保存大量的代码，并且根据需要，既可以存入(或叫写入)备用也可以取出(或叫读出)使用。

存贮器由许多单元 (location) 组成，每一个这样的单元存放一个(计算机处理的) 信息单位 (single unit of nformation)。这样的单位叫做“字”。每一个“字”都是一个二进制数所表示的代码，又叫字节 (byte)，每个字节都由一定数目的“0”与“1”构成，每一个这样的“0”或“1”都叫做位 (bit)，每一个字节所包含的的位数叫做字

长。对于一种机器来说，字长是固定的；但不同的机器，字长可能不同，有4、8、16、32位等不一。字长是机器的一项重要技术指标。APPLE-Ⅱ的字长是8位。

在一个存贮器内，存贮单元都是统一编号的，每一个存贮单元都有一个固定的号码，称为该贮存单元的地址(address)，存贮单元不同，地址就不同。要往(或从)一个存贮单元写入(或读出)数据时，都必须给出该存贮单元的地址。地址的号码都是连续的并且从0开始。例如，假定有N个存贮单元，那末它们的编号是0～N－1。

习惯上，每1024（256×4）个存贮单元称为1K，存贮器的容量是机器的又一项重要技术指标。就APPLE—Ⅱ来说，通常是64K，它们的地址号码是0～65535（总共65536个），用十六进制（hexadecimal）表示是$0000～$FFFF。

有一种存贮器，用户只能从它的任何一个地址读出数据，却不能写入新的数据，这种存贮器称为只读存贮器ROM（read only memory），其中的数据，通常是由制造的厂家存入的，用户不能更改。

还有一种是随机存贮器RAM（random access memory），我们既可以读出其中的数据，也可以存入新的数据。

我们可以把存贮器想像成是一排排盒子(存贮单元)，每个盒子上都有一个不同的号码(地址)，每个盒子可以存放一个信息单位(相当于一个字符)；盒子分成8个小格（位），每一格放一个“0”或“1”，这8个“0”和“1”的不同组合，表示着存在该盒子里的不同信息单位。

输入设备是指把数据输入计算机的设备，它把外部数据变成一种计算机能接受的电子信号送入计算机。例如，外部

数据经常是利用一个类似打字机的键盘（keyboard）键入的。这时，键盘便是一种输入设备，每按下一个键，代表键上字符的电子信号便输入计算机。

输出设备是输出计算机数据的设备，其作用与输入设备相反，它把数据从内部的电子信号变成外界能应用的某种形式，最常见的输出设备是打印机和显示器。

一个键盘输入设备往往与一个显示器或一台打印机组合成一套输入／输出设备（I／O），这样的组合体通常叫做终端（terminal）。

在外围设备（peripheral device）中，除了输入／输出设备外，往往还有外存贮器，如：磁盘（magnetic disk）、磁带（magnetic tape），用来存贮暂时不用的程序和数据。

§ 1.2 计算机语言

要计算机完成一项运算，就必须向计算机发出指令（instruction）。编写任何指令都必须遵循一定的规则，违反了这些规则，计算机就不能“理解”发出的指令，更无从执行这些指令。

当我们在编写指令或程序（program）时，我们是与计算机打交道，或者是说，与计算机进行对话。就像人们在相互交往中需要使用自然语言（natural language）一样，与计算机进行对话，就必须使用计算机语言（computer language）。计算机语言种类繁多。

有一种计算机语言叫机器语言(machine language)，它是计算机在工作过程中实际使用的语言。这种语言对计算机

来说是一系列电子脉冲（electronic impulse），但对编写程序的人来说却是一连串的二进制数，即一系列的“0”与“1”。每一种计算机都有它自己的机器语言，因此可以说，有多少种计算机就多少种机器语言。但有一点是共同的，即：不论哪一种计算机都只懂得自己的一种机器语言。所以每一条指令都必须或者直接用机器语言写；或者先用其他语言写，然后翻译成机器语言。两者必居其一。

机器语言是计算机的最根本的语言，因此，我们把它叫做低级语言（low-level language）。对于用户来说，低级语言是很难懂的，要求他们用低级语言来编写指令或程序往往是不切合实际的。为了降低编程的难度、推广计算机的应用，计算机专业人员又针对各种使用领域，创造了各种高级语言（high-level language）。

高级语言，比起机器语言来，大大地接近了人的自然语言或数学语言，但它们也有一个共同点，即用任何一种高级语言写的指令或程序，都必须先翻译成机器语言，然后才能被计算机执行，而且不同的高级语言其翻译的过程也各不相同。好在这翻译工作是由计算机自动完成的，而且也不向我们显示它的“译文”，我们也无须过问。

在高级语言中，用得最广泛的语言之一是BASIC语言。它是Beginner's All-purpose Synbolic Instruction Code的缩写，最初是在60年代中期由Dartmouth学院创建的。当时的目的是使低年级学生容易学习编程。但由于它有易学，易用的优点，结果在计算机解题中，尤其是在人机对话系统中得到了非常广泛的应用，目前可以说是小型机或微机中用得最广的语言。

BASIC语言，自从创建以来，已有不少修改和改进。

制造计算机的厂家，往往根据自己机器的特点，对BASIC语言进行某些修改和补充，使得在不同机器上使用的BASIC都稍有不同，结果在BASIC语言中出现了某些“方言”（dialect），使BASIC有许多不同的版本（version）。但差异毕竟是小的，不会构成使用BASIC语言的困难。

BASIC本身又分为基本BASIC、扩展(浮点) BASIC(是前者的扩充)以及整数BASIC。在编制英语教学与研究的软件时，我们要使用扩展BASIC(包括基本BASIC)，这就是本书所要介绍的主要内容，并且侧重于编制英语教学软件中的应用。

第二章　扩展BASIC

§ 2.1　程序及语句

以下我们简单地称之为语句（statement）。每个语句由下列四个部分组成：

1. **语句行号（line number）或叫标号**　这是语句的开首部分，它是一个无符号整数，表示该语句在程序中被执行的次序。在同一程序中，一个行号只能使用一次，不能有重复的行号。

允许使用的最小行号是0，但允许使用的最大行号却要视机器内存容量的大小而定，通常可以在说明书中查到。APPLE-Ⅱ机的行号范围是0～36999。当然，第一个语句的行号可以大于0，最后一个语句的行号可以小于36999，见上例。此外，行号无须连续，间隔也不强求相同，计算机一律按行号由小到大的顺序依次执行各行语句。上例中的程序也可写成下列形式，其执行结果不变。

```
100    LET  A=70
120    LET  B=80
135    LET  C=84
145    LET  D=90
160    LET  E=A+B+C+D
180    LET  F=E/4
200    PRINT  F
```

图 2.1.2

一般地说，行号之间留有一定的间隔是适当的。这样便于修改程序时插入新的语句。

每个语句的长度（字符数）不得超过255（包括空格在内）。在这个前提下，几个相联的语句可以使用同一个行号，也就是说，同一行号后可以接写几个语句，但语句间必须加用一个冒号。在执行时，计算机将从左到右依次执行各语

句。例如，上列程序也可改写成下列形式，其执行结果仍然不变。

```
100    LET  A=70 : LET  B=80 : LET
       C=84 : LET  D=90
145    LET  E=A+B+C+D : LET
       F=E／4
200    PRINT  F
```

图 2.1.3

2. 语句定义符(keyword) 这是从键盘输入的字母组合(单词或是一种特定形式的缩写)，在某些机器上必须使用大写字母，APPLE-Ⅱ便属此列。定义符接在行号之后，与行号之间可留空格，也可不留。例如图 2.1.1 中的 LET、PRINT 就是定义符。

定义符规定了计算机将进行何种操作。例如 PRINT 是计算机进行打印(数据)的操作。BASIC 中允许使用的定义符是有限的，以后我们将逐一介绍。切不可随意地拿英文字当作定义符使用。实际上，定义符是表示某种操作的“代码”。定义符原先作为一个英文字时的含义恰好指明了此种操作，如：

PRINT　　打印，
LET　　令，

这就为我们学习 BASIC 语言提供了方便。

3. 语句体 它接在定义符之后，相互间可留或不留空格。它是定义符规定的操作所要直接涉及的内容，例如图2.1.1中的

A=70
E=A+B+C+D

及第7语句中的F都是语句体。语句体与定义符有着非常密切的关系，介绍各种定义符及其相应的语句体便是本章的基本内容。

4. **语句结束符** 在一个语句键入完毕时，应按一次回车换行符RETURN键，以表示该语句到此结束。这时，我们可以看到屏幕上的光标移到了下一行的头上，但并不出现"RETURN"这个词，也不可以键入字母组RETURN来代替按一次RETURN键。

每一种语句都有一定的格式，这种格式就叫句法(SYNTAX)。如果格式错了，那末当计算机执行到这个语句时就会中止执行程序，并在屏幕上显示错误信息，在APPLE-Ⅱ上是：

SYNTAX ERROR

要求予以改正。

每一个语句都会促使计算机进行某些操作，从而达到预期的目的，这种功用叫语义(SEMANTICS)。例如图2.1.1中第7语句的语义是：在屏幕上显示F的值。了解一个语句的语义，对正确、灵活地应用这个语句是大有裨益的。

句法和语义是BASIC语句的两个方面，学习BASIC语句就是要学习这两方面的内容。

§ 2.2 程序的键入

程序语句要通过键盘逐个字符地输入计算机，在按过一次RETURN键以后，该语句便存入计算机内存。若发现已输入的某个语句有错，那末只要重新键入正确的语句则可。此时，屏幕上虽然可能还保留着原来的错误语句，但在计算机的内存里已被新键入的正确语句所替代。例如，若先键入

语句

```
100  LETA =80
```

然后键入语句

```
100  LETA =90
```

则在计算机的内存里，前者已不复存在，由后者取而代之。同理，若要取消原有的某个语句，则只须键入该语句的行号并按一次RETURN键便成。我们可以把这种情形想像为用一个没有定义符及语句体的“空白”语句取代了原来的语句。例如，若要取消图 2.1.2 中的第 135 语句，则键入

```
135
```

并按一次RETURN键即可。

至于其他修改办法，则须参阅机器的说明书。无法一概而论。

各语句按行号由小到大依次键入，但也可以按其他任意的次序键入，计算机仍将自动地按行号由小到大的顺序执行各语句，也就是说，键入语句的先后次序与计算机执行各语句的先后次序无关。例如图 2.1.2 中的程序也可按下列次序键入：

```
200  PRINT  F
180  LET  F=E／4
160  LET  E=A+B+C+D
145  LET  D=90
135  LET  C=84
120  LET  B=80
100  LET  A=70
```

图 2.2.1

但各语句的执行次序仍然是 100，120，…，180，200；实

际上，整个程序与原来的程序完全相同。

§ 2.3 键盘操作命令

在BASIC语言中，键盘操作命令又叫键盘命令、直接命令或简称为命令（command）。每一条命令都由大写(有些机器许可用小写）的英文字母构成，从键盘输入。每键入一条命令并按一次RETURN键后，计算机便执行一定的操作（为书写的方便起见，以后我们用↵代表按一次RETURN键，但作为语句结束符的此项操作，则不予标出）。本节介绍键盘命令NEW、RUN、LIST、HOME。

NEW

通常，计算机的内存里只能容纳一个程序，不能有两个以上的程序并存。如果内存里已存贮着一个程序（即若干个语句），那末新键入的任何语句都将是对原有程序的修改或扩充。例如，当内存中已存贮着图2.1.2中的程序时，若键入下列语句：

```
80  LET  B=40 : LET  C=40／8
100  PRINT  C
```

那末，内存中的程序就被修改成：

```
80  LET  B=40 : LET  C=40／8
100  PRINT  C
120  LET  B=80
135  LET  C=84
145  LET  D=90
160  LET  E=A+B+C+D
```

```
180  LET  F=E/4
200  PRINT  F
```

而不可能产生只包含下列两个语句的新程序：

```
80  LET  B=40:LET  C=40/8
100  PRINT  C
```

每当我们不再需要当前的程序而要重新键入一个程序时，应先键入命令

```
NEW↵
```

命令NEW的功能是使计算机从内存中清除当前的程序及所输入的一切数据。通常，在着手键入新的程序之前，应首先键入此项命令。

在APPLE-Ⅱ上，命令NEW没有清屏的功能，所以使用此项命令后，屏幕上可能仍然残留着原先本已显示着的语句。但这些语句已经“无效”。例如，在键入图2.1.2中的程序后，若键入NEW↵，则屏幕上虽然还仍然显示着原来的各语句，但该程序实际上已被从内存中全部清除。

RUN

键入一个程序，仅仅意味着把一个程序存入计算机的内存。在命令计算机执行该程序之前，不可能获得任何执行结果。例如，图2.1.2中的程序，在键入

```
200  PRINT  F
```

后，我们还看不到F的值。

为了让计算机执行程序以取得所需的结果，则必须键入命令

```
RUN↵
```

命令RUN的作用是使计算机执行已输入的程序。