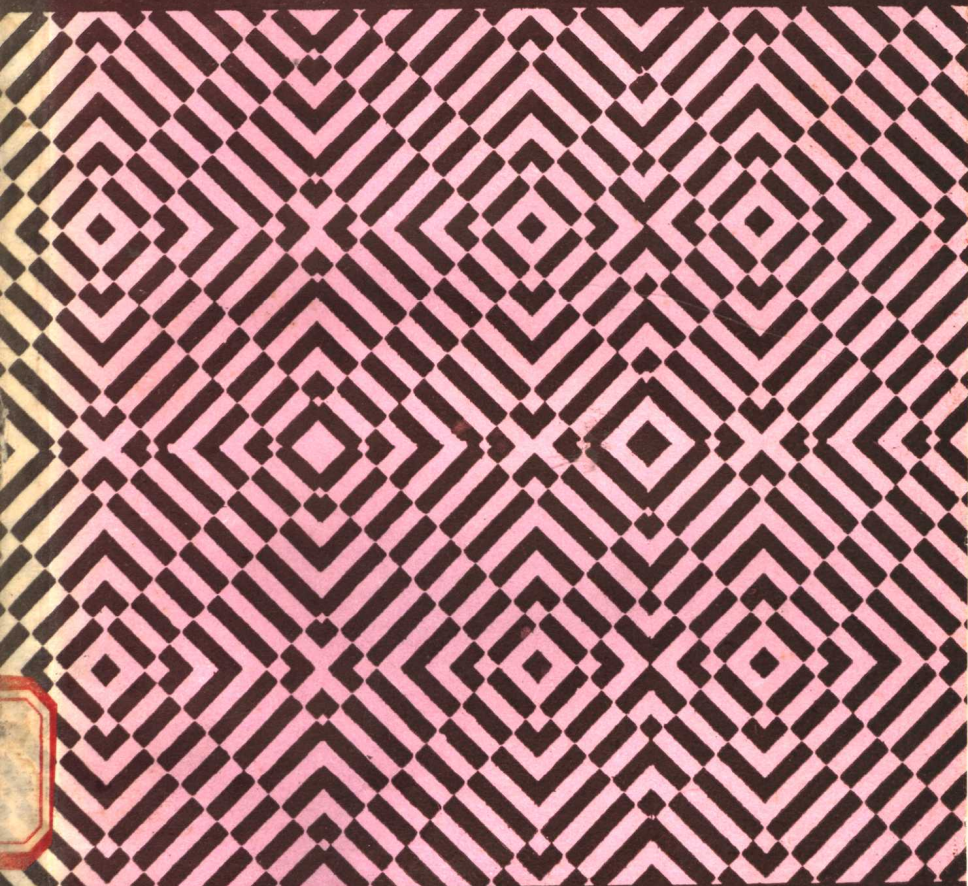


# 微机辅助管理软件初阶

李玉琴 王立威 王露露 合编



上海科学技术文献出版社

SHANGHAI KEXUE JISHU WENXIAN CHUBANSHE

# 微机辅助管理软件初阶

李玉琴 王立威 王露露 合编

上海科学技术文献出版社

**微机辅助管理软件初阶**

**李玉琴 王立威 王露露 合编**

\*

**上海科学技术文献出版社出版发行**

(上海市武康路2号)

**新华书店经销**

**昆山亭林印刷厂印刷**

\*

**开本 787×1092 1/32 印张 7.5 字数 181,000**

**1989年5月第1版 1989年5月第1次印刷**

**印数: 1—3,300**

**定价: 4.60元**

**ISBN 7-80513-350-6/T·121**

**《科技新书目》187-299**

## 序 言

近年来,微型计算机已在国内较普遍应用,全国的装机量已达10余万台,其中一半以上是用于企事业辅助管理及事务处理。相当多的管理人员已认识到计算机在管理工作方面可以发挥人所不及的作用。

但是,要在管理工作中用好计算机,必须配备适应管理工作要求的管理应用软件。各种管理软件之间,在规模和复杂程度上差异很大。大而复杂的管理软件是需要众多的计算机专业人员与各级管理人员互相配合进行开发的,而对于一些简单的辅助管理软件,假如能由直接从事管理的人员按照需要自己编制的话,是最好不过的。这样做,可使软件功能比较切合实际,在工作情况有所演变时,也易于及时作出改进。

微型计算机已为编制辅助管理软件提供了多种编程工具,其中常用的有BASIC语言、COBOL语言及dBASE数据库管理系统等。这些编程工具提供了一系列的编程规则,管理人员利用这些规则在较大程度上可以无视计算机内部结构来编制软件,使计算机按所要求的方式进行工作。现在,不少管理人员对微型计算机的应用产生了浓厚的兴趣,有的已初步掌握微型计算机上的基本工具与操作方法,这是值得提倡的。然而,一般管理人员即使掌握了这些工具的基本规则,要针对具体的工作目标,确当运用这些编程工具,尚需要具备一定的系统设计知识与实际编程经验。本书编写的目的是为从事计算机辅助管理的人员介绍这方面重要的基本技术,作为他们编制辅助管理应用软件的

初阶。

本书分三部分,分别介绍如何在 IBM PC 及其国产兼容机上利用 BASIC、COBOL 和 dBASE III 软件工具编制管理软件。着重于下列常用的基本功能模块的编制: 屏幕菜单、数据输入、数据文件的设计与组织、索引文件的建立、数据的排序、数据的检索与修改、数据的处理与统计。并结合典型实例予以说明。在附录中还介绍了软件结构化的设计思想、不同语言之间的调用与连接。力求使读者通过本书掌握一定的编程方法,并能比较正确运用上述一些软件工具。

本书的第一部分和附录 II 由李玉琴编写,第二部分及附录 III 由王立威编写,第三部分及附录 I 由王露露编写,王春元审核了全书。由于编写时间仓促,有不妥与错误之处,希望读者不吝指正。

编 者

1988 年 1 月

# 目 录

## 第一部分 PC—BASIC 程序编制方法

|                                  |    |
|----------------------------------|----|
| 第一章 编制 BASIC 程序的初步知识 .....       | 1  |
| 第一节 IBM—PC 机上配置的 BASIC 及其运行条件... | 1  |
| 第二节 PC—BASIC 的基本功能 .....         | 4  |
| 第三节 简单 BASIC 程序的编制方法 .....       | 12 |
| 第二章 人机对话式的程序设计 .....             | 16 |
| 第一节 菜单程序设计 .....                 | 16 |
| 第二节 数据输入时的查错与修改 .....            | 22 |
| 第三章 数据处理 .....                   | 25 |
| 第一节 数据文件 .....                   | 25 |
| 第二节 数据的排序、造索引及查寻 .....           | 41 |
| 第四章 数据输出 .....                   | 56 |
| 第一节 BASIC 中输出语句及其应用 .....        | 56 |
| 第二节 屏幕显示和报表输出 .....              | 59 |

## 第二部分 PC—COBOL 程序编制方法

|                                       |    |
|---------------------------------------|----|
| 第一章 COBOL 程序设计基本知识 .....              | 64 |
| 第一节 COBOL 标准 .....                    | 64 |
| 第二节 PC—COBOL 的特性 .....                | 65 |
| 第三节 结构程序设计在编写 COBOL 源程序中的应<br>用 ..... | 67 |
| 第四节 程序的模块化 .....                      | 70 |

|                              |     |
|------------------------------|-----|
| <b>第二章 字符处理与报表处理</b> .....   | 71  |
| 第一节 字符串的分离技术 .....           | 71  |
| 第二节 字符查找 .....               | 81  |
| 第三节 表处理 .....                | 82  |
| <b>第三章 文件系统</b> .....        | 96  |
| 第一节 顺序文件的处理 .....            | 97  |
| 第二节 随机文件的处理 .....            | 104 |
| 第三节 索引文件的处理 .....            | 118 |
| 第四节 打印输出文件的处理 .....          | 126 |
| <b>第四章 其它编程技术</b> .....      | 131 |
| 第一节 “菜单”技术 .....             | 131 |
| 第二节 链接技术 .....               | 133 |
| 第三节 COBOL 语句调用机器级语言子程序 ..... | 136 |

## 第三部分 dBASE III 的程序编制方法

|                                      |     |
|--------------------------------------|-----|
| <b>第一章 dBASE III 程序设计的一般规则</b> ..... | 139 |
| 第一节 概 述 .....                        | 139 |
| 第二节 变 量 .....                        | 142 |
| 第三节 运算符 .....                        | 144 |
| 第四节 标准函数 .....                       | 146 |
| 第五节 表达式 .....                        | 147 |
| <b>第二章 顺序、判断、循环及过程</b> .....         | 147 |
| 第一节 顺 序 .....                        | 148 |
| 第二节 判 断 .....                        | 149 |
| 第三节 循 环 .....                        | 155 |
| 第四节 过程(子程序) .....                    | 158 |
| <b>第三章 dBASE III 文件及数据检索</b> .....   | 161 |

|                                            |     |
|--------------------------------------------|-----|
| 第一节 dBASE III 使用的文件及其形式 .....              | 161 |
| 第二节 数据检索 .....                             | 174 |
| 附录 I 应用程序基本设计方法 .....                      | 182 |
| 附录 II BASIC 与 dBASE III 之间的数据共享及程序调用 ..... | 209 |
| 附录 III COBOL 与 dBASE III 应用程序的连接 .....     | 224 |
| 参考书目 .....                                 | 228 |

# 第一部分 PC-BASIC 程序 编制方法

## 第一章 编制 BASIC 程序 的初步知识

在叙述具体的程序编制方法前,先简单介绍一下 PC 机上使用的几种 BASIC 程序的特点及其运行条件,以及如何进入 BASIC 编制程序。

### 第一节 IBM—PC 机上配置的 BASIC 及其运行条件

#### 1. 在 IBM—PC 机及其兼容机上配置的 BASIC 种类

##### (1) 磁带 BASIC

它是驻留在 32K ROM 中的 BASIC 解释程序(在国产化的 0520 系列中则无此常驻部分),它支持 BASIC 的所有基本功能,可以显示 ASCII 字符和部分图形符号;支持图形显示器,可以直接使用特定的语句作图;支持扬声器、光笔和游戏操纵杆等,但是它不支持磁盘驱动器。

##### (2) 磁盘 BASIC

它提供磁带 BASIC 的全部功能,支持磁盘文件系统,可使用磁盘记录用户的程序和数据,并且设置了一个记录日期和时间的内部“时钟”;支持 RS—232 异步通信和两台附加的打印

机。磁盘 BASIC 要求机器至少有 32K 随机存贮器(RAM)。

### (3) 高级 BASIC

它提供磁盘 BASIC 的全部功能,增加了事件陷阱的设置功能,允许用户捕获键盘、异步通信器等设备发出的中断;增加了作图和音乐语句。高级 BASIC 要求机器至少有 48K 随机存贮器。

磁盘 BASIC 和高级 BASIC 都必须在磁盘操作系统(PC—DOS 或者 CC—DOS)支持下才能工作。如在 CC—DOS 下运行,尚要求有足够的 RAM 区存放汉字字库。

### (4) 编译 BASIC

它具有高级 BASIC 的大部分功能。使用编译方式,执行速度比解释 BASIC 快三至十倍,程序占据的内存空间也比解释 BASIC 少,且便于用户程序的保密。

## 2. 各种 BASIC 的启动

### (1) 启动盒式磁带 BASIC

要求计算机的 A 驱动器中无盘片或者驱动器门开启;要求在硬盘 C 中无操作系统时接通计算机电源,在屏幕上显示:

The IBM Personal Computer Basic Version 1.10 Copyright IBM Corp,etc.  
33402 Bytes free

OK

表示已进入磁带 BASIC,可编写或执行磁带 BASIC 控制下的应用程序。

### (2) 启动磁盘 BASIC 和高级 BASIC

首先引入磁盘操作系统(PC—DOS 或 CC—DOS),然后把含有磁盘 BASIC 及高级 BASIC 的盘插入 A 驱动器,并键入:

A>BASIC<CR>

或 A>BASICA<CR>

前者就进入了磁盘 BASIC, 而后者则进入高级 BASIC。

在执行进入磁盘 BASIC 或高级 BASIC 命令中, 还可以设定所要执行应用程序的文件名, 选择可同时打开的文件个数, 选择随机文件缓冲区的大小, RS-232 异步通信缓冲区的大小和 BASIC 工作区的大小, 其选择方法如下:

A> BASIC[A] ["d: 文件名"] [/F: 文件数] [/S: 随机缓冲区大小] [/C: 异步通信缓冲区大小] [/M: BASIC 最大工作区]

/F: 指定 BASIC 中可同时打开的文件个数, 如不指定, 系统设定最多可打开三个文件。可设置同时打开 3 至 15 个文件。

/S: 指定随机文件缓冲区的大小, 系统中设置为 128 字节, 可设定的最大字节为 32767。

/C: 指定异步通信缓冲区的大小, 系统设定为 128 字节, 可指定的最大字节为 32767。

/M: 指定 BASIC 工作区, 可以设置的最大工作区是 64 K 字节。

如果在引导 DOS 后键入:

A> BASICA "B:MUSIC"/F:5/S:256/C:1024/M:32767  
<CR> 首先把高级 BASIC 解释程序装入内存, 设置同时可打开的文件数为 5, 随机文件缓冲区 256 字节, 异步通信缓冲区 1024 字节, BASIC 最大工作区 32767 字节, 然后把 B 驱动器上的 BASIC 文件 MUSIC.BAS 装入内存, 并且执行此程序。

### (3) 编译 BASIC 的使用

BASIC 程序的编译过程与其它高级语言程序的编译过程相类似。首先建立源程序文件, 然后调用编译程序对源程序编译产生目标文件, 如果目标文件无语法性错误, 则用连接程序对

各目标文件模块进行连接,即可产生可执行的(.EXE)文件。由于 BASIC 有解释程序,源程序的编辑、修改、调试都极为方便。BASIC 的源程序可以在 BASIC 解释程序控制下构成,也可使用行编辑 EDLIN 建立。在 BASIC 解释程序控制下输入程序文本,并进行调试,再修改调试中发现的错误。调试完毕,用 SAVE 命令以 ASCII 字符形式把程序存盘,就产生了我们编译中使用的源程序。由于编译程序要求 BASIC 源程序使用.BAS 为扩展名,在用 EDLIN 建立源文件时要指定文件扩展名.BAS。

源程序构成后,调用 BASCOM 对源程序进行编译。如源程序文件为 B:MUSIO.BAS,则键入: BASCOM"B:MUSIO"  
<OR>,即对 MUSIO 编译产生 MUSIO.OBJ 目标文件,然后调用 LINK 程序对 MUSIO.OBJ 进行连接(键入 LINK MUSIO  
<OR>),产生 MUSIO.EXE 可执行文件,整个编译过程就结束了。

## 第二节 PC-BASIC 的基本功能

为了便于阅读后面章节中的程序,本节中把 BASIC 中主要控制键、函数、数值精度及屏幕坐标等功能简单介绍一下。

### 1. 控制键与功能键

#### (1) BASIC 命令功能键

BASIC 命令功能键 F1~F10 见表 1.1。

可以用 KEY n,x\$ 语句重新定义功能键,其中 n 为 1~10, x\$ 是欲定义的字符串。

#### (2) 语句功能键

语句功能键(同时把 ALT 键和某一字符键按下)见表 1.2。

#### (3) 编辑键

编辑键功能见表 1.3。

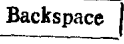
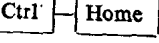
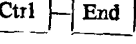
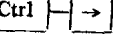
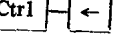
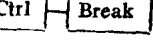
表 1.1 BASIC 命令功能键

| 功能键                  | 定 义      | 功能键                   | 定 义                |
|----------------------|----------|-----------------------|--------------------|
| <b>F<sub>1</sub></b> | LIST     | <b>F<sub>6</sub></b>  | RUN<CR>            |
| <b>F<sub>2</sub></b> | RUN<CR>  | <b>F<sub>7</sub></b>  | TRON<CR>           |
| <b>F<sub>3</sub></b> | LOAD"    | <b>F<sub>8</sub></b>  | TROFF<CR>          |
| <b>F<sub>4</sub></b> | SAVE"    | <b>F<sub>9</sub></b>  | KFY                |
| <b>F<sub>5</sub></b> | CONT<CR> | <b>F<sub>10</sub></b> | SCREEN 0, 0, 0<CR> |

表 1.2 语句功能键

| 功能键           | 定 义    | 功能键           | 定 义    |
|---------------|--------|---------------|--------|
| <b>ALT</b> -A | AUTO   | <b>ALT</b> -M | MOTOR  |
| <b>ALT</b> -B | BSAVE  | <b>ALT</b> -N | NEXT   |
| <b>ALT</b> -C | COLOR  | <b>ALT</b> -O | OPEN   |
| <b>ALT</b> -D | DELETE | <b>ALT</b> -P | PRINT  |
| <b>ALT</b> -E | ELSE   | <b>ALT</b> -R | RUN    |
| <b>ALT</b> -F | FOR    | <b>ALT</b> -S | SCREEN |
| <b>ALT</b> -G | GOTO   | <b>ALT</b> -T | THEN   |
| <b>ALT</b> -H | HEX\$  | <b>ALT</b> -U | USING  |
| <b>ALT</b> -I | INPUT  | <b>ALT</b> -V | VAL    |
| <b>ALT</b> -K | KEY    | <b>ALT</b> -W | WIDTH  |
| <b>ALT</b> -L | LOCATE | <b>ALT</b> -X | XOR    |

表 1.3 编辑键功能

| 编辑键                                                                                 | 功 能             |
|-------------------------------------------------------------------------------------|-----------------|
|    | 光标上移一行          |
|    | 光标下移一行          |
|    | 光标左移一字符         |
|    | 光标右移一字符         |
|    | 插入若干字符          |
|    | 删除光标处一个字符       |
|    | 反向删除一个字符        |
|    | 删除一行            |
|    | 光标移回屏幕左上角       |
|    | 光标移回屏幕左上角,并清除屏幕 |
|    | 光标移至逻辑行末        |
|  | 从光标处删到逻辑行末      |
|  | 光标右移一个字         |
|  | 光标左移一个字         |
|  | 终止编辑            |
|  | 光标移到下一个制表位置     |

## 2. 屏幕显示形式

### (1) 文本显示形式

在文本显示形式下,每幅屏幕可显示 ASCII 字符 25 行(汉字显示方式只能 11 行,但这些指标是对 PC—XT 及其兼容机上所配置的显示器而言,对于 0520CH 等高一档微机,显示器的分辨率有很大提高,汉字也可显示 25 行),每行可显示 40 个字符或 80 个字符(汉字方式均减半)。文本显示形式下,行编号为 1~25,列编号为 1~40 或 1~80。可以用 LOCATE 语句选择字符在屏幕上显示的位置,例如:

```
LOCATE 10, 20:PRINT"A"
```

即在屏幕第 10 行 20 列的位置上显示字符 A。

### (2) 图形显示形式

图形显示可分为中分辨率与高分辨率两种情况,在中分辨率方式下,屏幕横向为 320 点,纵向为 200 点,全屏幕为  $320 \times 200$  点。高分辨率方式下,横向为 640 点,纵向为 200 点,全屏幕为  $640 \times 200$  点。

图形方式下,屏幕上各点可用坐标方式表示,见图 1.1。

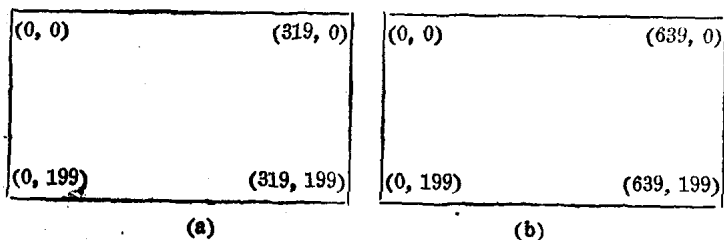


图 1.1

在中分辨率显示方式,屏幕左上角的坐标为(0, 0),即X坐标为0,Y向坐标也为0,右上角坐标为(319, 0),也即是X为319,Y为0,屏幕右下角坐标为(319, 199),即X为319,Y为199。这

两个值是横向坐标及纵向坐标的最大值。右图中显示的是高分辨率图形方式的坐标表示,横向坐标从0至639,纵向坐标从0至199。

在图形方式下,可用图形语句选择屏幕上坐标位置。

### (3) 彩色显示

在文本方式和图形中分辨率方式可以选择屏幕显示的颜色,文本显示方式有十六种颜色供选择,见表1.4。

表 1.4 文本显示方式的颜色

| 编号 | 颜色 | 编号 | 颜色   |
|----|----|----|------|
| 0  | 黑  | 8  | 灰    |
| 1  | 蓝  | 9  | 浅蓝   |
| 2  | 绿  | 10 | 浅绿   |
| 3  | 深蓝 | 11 | 浅青蓝  |
| 4  | 红  | 12 | 浅红   |
| 5  | 洋红 | 13 | 浅洋红  |
| 6  | 棕  | 14 | 黄    |
| 7  | 白  | 15 | 高亮度白 |

可以选择0~7号颜色作为屏幕底色,字符显示的颜色可选择0~15号。另可以选择16~31号,表示前16种颜色闪烁。

在图形方式下,一幅图形最多可用四色,究竟选择那四种,则由COLOR语句选择某一组混合颜色,每一组颜色为四种,0表示底色,另有三种(1, 2, 3)可供图形语句选用作图形显示色。混合色有两组,见表1.5。

### 3. 数据类型及其表示形式

数据一般分为整型、单精度型、双精度型、字符串型以及数组。PC-BASIC中用特定的变量尾标表示数据的类型。如果省略尾标,而又未用DEF类型语句定义过,则把变量作为单精

表 1.5 两组混合色

| 编号 | 混合色 0 | 混合色 1 |
|----|-------|-------|
| 1  | 绿     | 蓝     |
| 2  | 红     | 洋红    |
| 3  | 棕     | 白     |

度变量处理。

数据类型及尾标如下：

- 整型变量，尾标%，表示范围-32768至32767。在存储器中占据两个字节。整型变量可分为十进制、十六进制(用 &H 表示，如 &H1A 表示十六进制 1A)和八进制(用 &O 或 & 表示)。

- 单精度变量，尾标! (或可省略尾标)，可表示具有 7 位或 7 位以下有效数字，在存储器中占据 4 个字节，指数形式表示时使用 E(例如 -1.09E-06 表示数值  $-1.09 \times 10^{-6}$ )。

- 双精度变量，尾标#，可表示 8 位至 16 位有效数字，在存储器中要占 8 个字节，当采用指数形式表示时使用 D (例如 -1.091764D-06)。

- 字符串变量，尾标\$，字符串长度可为 0~255 个字符。

- 数组变量：数组变量可以是整型、单精度型、双精度型和字符串型，其类型表示同前所述可用变量尾标%，!，#，\$ 表示。数组变量的最大下标要用 DIM 语句加以说明，否则只能为 10，最小下标要用 OPTION BASE n 选择，n 可以是 0 或 1，当采用 OPTION BASE 说明时，最小下标为 0。

DIM 语句格式如下：

DIM 变量(<下标表>){, <变量>(<下标表>)...}

下标表的维数最多可达 255，每维元素最多允许 32767 个。