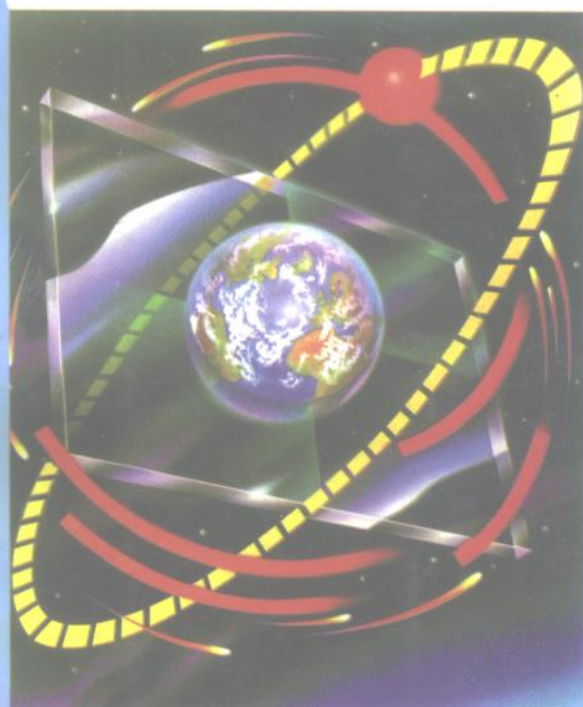


常用程序设计语言 简明操作方法



BASIC PASCAL C语言 C++ FORTRAN 汇编



王汝传 编著
西南交通大学出版社

常用程序设计语言 简明操作方法

王汝传 编 著

西南交通大学出版社

内 容 简 介

本书重点介绍常用程序设计语言 BASIC、PASCAL、C、C++、FORTRAN 和汇编语言上机操作的指南,不但详细叙述各种不同程序设计语言以及它们不同编译系统上机的操作方法和步骤,而且还给出出错信息和一些语言的库函数以便使用人员查询。本书是一本很实用的微机操作手册。

本书可作为大专、中专各类专业计算机应用实验课教程,也可作为计算机各种应用培训班和各类工程技术人员及自学者参考用书。

JS/38/05



常用程序设计语言简明操作方法

王 汝 传

责任编辑 唐 晴

*

西南交通大学出版社出版发行

(成都 二环路北一段 610031)

郫县印刷厂印刷

*

开本:787×1092 1/16 印张:15.375

字数:371 千字 印数:1~3000 册

1997 年 11 月第 1 版 1997 年 11 月第 1 次印刷

ISBN 7-81057-056-0/T·233

定价:19.00 元

前 言

近年来,我国计算机的应用和普及得到了迅速发展。计算机已进入各行各业,并已开始进入家庭。计算机应用范围几乎遍及科研机构、高等学校、政府机关、企事业单位。尤其是微型计算机在我国社会和经济发展中起着越来越大的作用,已成为实现四个现代化不可缺少的工具。随着微型计算机日益普及,人们不仅需要了解和掌握微机应用的一些基本知识和理论,而且还需要掌握计算机常用程序设计语言操作方法。学习使用计算机程序语言是使用计算机和研制开发计算机软件的必经之路,而上机实践训练是掌握和应用程序设计语言不可缺少的环节。目前许多程序设计语言教材中因限于篇幅,上机操作和语言系统环境或无介绍或介绍得过于简略,在上机实践中因缺乏指导材料,常常遇到问题而束手无策,而一般语言使用大全之类资料又十分庞杂,急用时又难于得心应手。为此,根据多年的教学实践,编写了这本《常用程序设计语言简明操作方法》。

本书是将常用程序设计语言 BASIC、PASCAL、C、C++、FORTRAN 和汇编语言以及它们的不同编译软件系统上机基本操作方法、步骤和出错信息汇集一起。本书共分六章,第一章介绍 IBM PC BASIC、TURBO BASIC、TRUE BASIC 和 QUICK BASIC 上机操作方法和步骤;第二章介绍 TURBO PASCAL 上机操作方法和步骤;第三章介绍 TURBO C、MS-C 和 QUICK C 上机操作方法和步骤;第四章介绍 BORLAND C++ 上机操作方法和步骤;第五章介绍 FORTRAN 77 上机操作方法和步骤;第六章介绍汇编语言上机操作方法和步骤。

作为一本实用性操作方法图书,它适合于每一个从事计算机应用与开发的人员,既可以作为程序设计初学者上机操作指南与参考,又可作为软件开发和设计人员简捷速查手册。本书还可以作为大、中专学生学习计算机程序设计语言课程上机指导教材,对于从事科研、教学和应用开发的广大科技人员也是一本较好的参考书,同时也可作为有关短训班人员及自学者上机操作参考书。

邮电部计算机教学指导委员会主任居悌教授对于本书的编写给予了大力支持和鼓励,并在百忙中审阅了书稿,同时提出了宝贵意见,在此表示深深谢意。此外,许多同行对本书编写提出了不少建议和修改意见,在编写过程中还参考了许多专家、学者的著作和成果,书中借鉴了他们的工作经验,在此一一表示忠心地感谢。

由于编写时间仓促,特别是编者水平有限,书中错误和不妥之处在所难免,敬请读者批评指正。

编 者

1996.11

目 录

第一章 BASIC 上机操作	1
1.1 概 述	1
1.2 IBM PC BASIC 上机操作	1
1.2.1 启动 BASIC	1
1.2.2 源程序的建立	2
1.2.3 源程序的编辑	3
1.2.4 源程序的运行和调试	5
1.2.5 源程序的保存和调用	6
1.2.6 常用 BASIC 命令简介	7
1.2.7 IBM PC BASIC 错误信息	9
1.3 TURBO BASIC 上机操作	10
1.3.1 TURBO BASIC 的进入与退出	10
1.3.2 源程序的输入	11
1.3.3 源程序的编辑	12
1.3.4 源程序的运行	15
1.3.5 源程序的保存和调入	15
1.3.6 TURBO BASIC 窗口功能简介	15
1.3.7 TURBO BASIC 错误信息	23
1.4 TRUE BASIC 上机操作	29
1.4.1 TRUE BASIC 进入与退出	29
1.4.2 源程序的输入	30
1.4.3 源程序的编辑	31
1.4.4 源程序的运行	32
1.4.5 源程序的调试	33
1.4.6 TRUE BASIC 系统命令	33
1.4.7 TRUE BASIC 错误信息	37
1.5 Quick BASIC 上机操作	40
1.5.1 Quick BASIC 的安装	40
1.5.2 Quick BASIC 的启动	41
1.5.3 Quick BASIC 的屏幕	43
1.5.4 菜单及其命令选择	45
1.5.5 对话框的使用	46

1.5.6	窗口管理	47
1.5.7	Quick BASIC 的退出	49
1.5.8	Quick BASIC 错误信息	49
第二章	PASCAL 上机操作	58
2.1	概 述	58
2.2	TURBO PASCAL 上机操作	58
2.2.1	TURBO PASCAL 运行环境	58
2.2.2	启动 TURBO PASCAL	59
2.2.3	主屏幕四部分简介	59
2.2.4	菜单命令操作	63
2.2.5	TURBO PASCAL 程序的编译、运行和调试	68
2.2.6	TURBO PASCAL 错误信息	71
第三章	C 上机操作	77
3.1	概 述	77
3.2	TURBO C 上机操作	77
3.2.1	TURBO C 2.0 的文件组合和配置	77
3.2.2	TURBO C 2.0 安装和启动	79
3.2.3	TURBO C 集成环境	81
3.2.4	TURBO C 命令行	93
3.2.5	源程序的输入、编译和运行	97
3.2.6	TURBO C 编译错误信息	98
3.2.7	TURBO C 2.0 库函数	103
3.3	MS-C 上机操作	104
3.3.1	系统安装	104
3.3.2	编辑源程序文件	104
3.3.3	源程序的编译	104
3.3.4	链 接	104
3.3.5	运 行	105
3.3.6	MS-C 错误信息	105
3.3.7	MS-C 5.0 库函数	120
3.4	Quick C 上机操作	138
3.4.1	Quick C 的集成方式操作	138
3.4.2	Quick C 的命令方式操作	139
第四章	C++ 上机操作	140
4.1	概 述	140
4.2	BORLAND C++ 上机操作	140
4.2.1	BORLAND C++ 的安装	140
4.2.2	BORLAND C++ 的启动和退出	142
4.2.3	BORLAND C++ 集成环境使用简介	143

4.2.4	源程序的建立和编辑	157
4.2.5	源程序的编译、链接、运行和调试	159
4.2.6	BORLAND C++ 错误信息	164
4.2.7	BORLAND C++ 库函数	179
第五章	FORTRAN 77 上机操作	191
5.1	概 述	191
5.2	MS-FORTRAN 上机操作	191
5.2.1	MS-FORTRAN 3.30 编译系统的组成	191
5.2.2	编辑 FORTRAN 源程序文件	191
5.2.3	编译源程序	193
5.2.4	链接和运行	196
5.2.5	编译元命令	196
5.2.6	MS-FORTRAN 5.0 的使用	200
5.3	IBM-FORTRAN 上机操作	204
5.3.1	IBM FORTRAN 2.0 编译系统的组成	204
5.3.2	FORTRAN 源程序文件的建立	205
5.3.3	编译源程序	205
5.3.4	链接和运行	206
5.4	FORTRAN 77 错误信息	208
第六章	汇编语言上机操作	216
6.1	概 述	216
6.2	汇编语言程序上机操作过程	216
6.2.1	建立汇编源程序	217
6.2.2	汇编程序的启动	217
6.2.3	链接	218
6.2.4	程序的运行	219
6.2.5	程序的调试	219
6.3	8086/8088 指令系统	227
6.4	汇编程序出错信息	236

第一章 BASIC 上机操作

1.1 概 述

BASIC 语言是微型计算机上最常用的且易于初学者掌握的程序设计语言。BASIC 是英语 Beginner's All—Purpose Symbolic Instruction Code 的缩写,意思是初学者通用符号指令码。

BASIC 语言的编译系统根据其类型不同分成两种形式:一种是解释型 BASIC 系统,另一种是编译型 BASIC 系统。解释型 BASIC 系统是逐条地把程序中语句翻译成机器指令,边解释边执行,这类 BASIC 系统有磁带 BASIC,磁盘 BASIC,高级 BASIC 和 GWBASIC。编译型 BASIC 与其他编译系统一样,先对 BASIC 语言程序进行编译然后生成可执行程序,编译型 BASIC 执行速度远大于解释型 BASIC。这类 BASIC 系统有 MS-BASIC, TURBO BASIC 和 QUICK BASIC。此外,有的 BASIC 版本既可以提供解释模式,又可以提供编译模式,如 TRUE BASIC。

1.2 IBM PC BASIC 上机操作

1.2.1 启动 BASIC

在 IBM PC 系统上启动 BASIC 是非常容易的。一般步骤如下:

(1) 把 PC-DOS 系统盘插入驱动器 A(这个软盘上装有 BASIC 系统或 GWBASIC 系统),关上小门。

(2) 打开计算机电源开关。在计算机经过 3~25 秒自检之后,它自动把 DOS 软磁盘中磁盘操作系统装入内存。

(3) 在回答屏幕提问,输入日期、时间后,在提示符 A> 后键入命令:

A> GWBASIC(或 BASIC 等)↵

几秒钟后,BASIC 解释程序调入内存,同时在屏幕上显示为:

GW - BASIC 3.23

(C)Copyright Microsoft 1983,1984,1985,1986,1987,1988

60300 Bytes free

OK

!LIST 2RUN 3LOAD "4SAVE" 5CONT 6. LPT1 7TRON 8TROFF 9KEY 10SCREEN

第一行是版本信息,第三行是自由空间,然后给出提示符 OK,告诉你系统进入 BASIC 状态,在屏幕最下面一行是显示出十个功能键 <F1> ~ <F10> 所代表功能,如 3LOAD 表示功能键 F3 代表 LOAD 命令,只要按下 F3 键,相当于键入 LOAD 命令。这样可节省用户键入的时

间。表 1.1 列出 10 个功能键的功能表。

表 1.1 10 个功能键之功能

键名	功 能	键名	功 能
F1	LIST(列程序清单)	F6	"LPT1" (规定每行打印字符数)
F2	RUN (运行程序)	F7	TRON (跟踪显示程序运行行号)
F3	LOAD (调程序文件)	F8	TROFF (取消跟踪显示)
F4	SAVE (存储程序文件)	F9	KEY
F5	CONT (继续运行程序)	F10	SCREEN 0,0,0 (决定屏幕状态)

至此,你可以键入 BASIC 命令,输入 BASIC 源程序,运行 BASIC 程序。

退出 BASIC 回到 DOS 状态命令是: `SYSTEM`
A >

1.2.2 源程序的建立

在提示符 OK 出现以后,用户就可以将编好的源程序输入到计算机内。例如:已知圆半径 $R = 10$,求圆面积 S 的程序。

```
10 LET R = 10
20 LET S = 3.1416 * R * R
30 PRINT "S=";S
40 END
```

输入步骤如下:

(1) 清除内存,键入命令:

```
OK
NEW
```

(2) 用户可以按行号 10,20,30,40,一行行地从键盘上敲入。

(3) 每一行敲完后要必须按一下回车键“`↵`”。

如 `10 LET R = 10`,这时计算机才可收到 `10 LET R = 10` 这个信息。此时光标自动移到下一行首址。

注意行号、语句关键字及变量之间需保留空格。

在输入程序时,我们可以使用 BASIC 提供的自动产生行号命令 AUTO。

命令格式: `AUTO[(起始行号),(增量)]`

命令功能:自动编行号,缺省时表示语句标号从 10 开始,增量为 10 自动编行号。

说明:

(1) AUTO 命令使用于输入程序,它使用户不必打入每一行号,同时还提醒用户是否有重行。

例: OK

AUTO_↵

10 LET R = 10_↵

20

使用 AUTO 命令后,只要机器不中断,机器会自动产生行号 10,20,30,⋯,如果内存中已有某行存在,则在行号后显示一个“*”号,警告用户这时输入的一切内容将替换已存在的程序行。

(2) 起始行范围在 0 至 65529 之内,缺省值是 10。

(3) 增量是每一行号增量,缺省值是 10。

例: AUTO 20,20_↵

这条命令产生行号是 20,40,60,⋯

AUTO 100,_↵

若上次产生行号增量为 20,这条命令产生行号是 100,120,140,⋯,

AUTO 0,5_↵

这条命令产生行号是 0,5,10,⋯

如果要想中止 AUTO,可按 <Ctrl> + <Break> 控制键。

无论是采用 AUTO 命令,还是自己键入行号,当完成上述求圆面积的源程序输入后,可键入 LIST 命令,屏幕上列出输入到内存的四行程序清单。说明源程序的建立工作完成。

1.2.3 源程序的编辑

对源程序的编辑加工包括修改、删除、插入等操作。

1. 基本编辑方法

(1) 当输入一个程序行在按回车键之前发现错误时,可用退格键“←”将光标移到有错误的字符下面,然后从这个字符以右,重新键入正确字符序列。

例: 下一行语句

20 □ LET □ Y = SIM(2X)

└─光标

在按回车键之前发现 SIM 的 M 有拼写错误,2 与 X 之间缺少乘号,为此,按五次退格键后,20 行变为

20 □ LET □ Y = SI[^]

└─光标

键入正确字符序列后再按回车键后为:

20 □ LET □ Y = SIN(2 * X)_↵

该行两处错误修改完毕。

(2) 按回车键后才发现错误,可保留错误所在行行号,重新键入正确的一行。

(3) 要删除一程序行,只需键入该行的行号并按回车键该行即被删除;如果删除一段程序,使用 DELETE 命令是很方便的(详见 § 2.2 六、6),例如:DELETE30-210_↵,命令执行以后,就从内存中删除 30~210 行程序段。

(4) 要插入新的程序行时,只需在程序当前末行键入所要插入的行号及语句内容,例如下面的程序段中:

```
10 LET T = (3 + 5) * 3 / 2
20 LET Y = SIN(2 * X)
30 PRINT T, Y
```

为了给变量 X 赋值,只需在 30 行后面键入一行:

```
15 LET X = 45 * 3.14159 / 180
```

使用命令 LIST,程序段就以行号递增顺序显示。

2. 一般编辑方法

一般编辑方法就是使用 BASIC 的各种编辑键及编辑命令对源程序进行灵活、快速编辑的方法。

例:设有如下程序段:

```
10 LET X = 45 * 3.14159 / 180
20 LET Y = SIM(2X)
:
310 IF A = 5 THEN 40 ELSE 70
—(光标)
```

经检查,20 行语句有如前述的错误:SIM 拼写错误及 2 与 X 之间缺少 * 号;310 行在 0 与 I 之间缺空格及在 A 字后多了空格,且需要删除该行中的 ELSE 170。

修改操作步骤如下:

(1) 按“↑”键一次移光标到 3 处,按“→”键 3 次右移光标到 I 处,然后按“Ins”键和空格键各一次,光标随字母 I 及其以后的字符向右平移一格,所需空格已插入。

(2) 按“→”键四次移光标到字母 A 后空格处按“Del”键一次删除空格,此时光标在“=”号下方。

(3) 按“→”键 11 次移光标到字母 E 的下方按复合键 <Ctrl> + <End> 一次或按“Del”键 7 次或空格键 7 次就能删除 ELSE 170,前两种方式删除后光标未移动;最后一种方法删除后光标右移了 7 格,最后按回车键。

(4) 键入命令:

```
EDIT 20
```

后,屏幕显示该行语句:

```
20 LET Y = SIM(2X)
```

↑ ↑ ↑
光标 应改为 N 插入 *

(5) 按“→”键 11 次移光标到字母 M 的下方,再按“N”键后, M 就改成 N,同时光标右移至左圆括号的下方,按“→”两次移光标到字母 X 的下方;然后按“Ins”键及“*”键各一次之后,乘号“*”已插入 2 与 X 之间,光标在 X 下方;按“End”键移光标至本程序行末尾,最后按回车键,两行修改完毕。

从上述实例可以看出, BASIC 对源程序编辑允许在全屏幕范围内进行,但必须以行为单位进行,即每一行错误修改完毕以后,均应按回车键,编辑工作才有效。

表 1.2 列出编辑键及其对应功能。

表 1.2 编辑键之对应功能

编辑键	功 能
[↑]	光标上移一行
[↓]	光标下移一行
[←]	光标左移一个字符
[→]	光标右移一个字符
[Ins]	插入/替换方式转换,在插入方式(第一次按该键)下,可在光标前插入字符,再按一次 Ins 键或按任一光标移动键都能退出插入方式,此时键入的字符代替光标所在的字符
[Del]	删除光标处的一个字符
[←]	删除最后键入的字符,即删除光标左边的那个字符(在键盘最上一行的[←]键)
[Esc]	删除一行,如果这一行是程序行,那么它并不从内存程序中清除
[Home]	把光标移至屏幕左上角
[Ctrl] + [Home]	清除屏幕,并把光标移至屏幕左上角
[End]	光标移至行末。从这个位置起所键入的字符都加在该行后面
[Ctrl] + [End]	从当前光标位置开始到行末的内容都被清除
[Ctrl] + [→]	光标右移一个字
[Ctrl] + [←]	光标左移一个字
[Ctrl] + [Break]	返回命令级,不保存对当前正在编辑行所做的任何改动
[↹](Tab)	移动光标到下一个表(Tab)站,每 8 个字符位置有一个表站,即位置 1,9,17 等等

1.2.4 源程序的运行和调试

1. 源程序的调试

调试也称动态调试,是将编辑好的源程序上机试算,以便查找错误。

动态调试手段通常有下面五种方法:

- (1) 在程序执行前发出跟踪命令 TRON,以供观察程序执行的路径,即执行各行行号序列。
- (2) 在源程序中安排若干个 STOP 语句,以便逐段分析执行情况。
- (3) 在源程序中设置若干个 PRINT 语句,以便了解中间变量取值情况。
- (4) 使用 LIST 语句,即 LIST n1 - n2 命令,可根据需要在屏幕上列出 n1 ~ n2 行的程序段。

经过查错、修改、运行,获得满意结果的程序,并正式投入运行之前,应当取消跟踪,删除为调试而设置 STOP 和 PRINT 语句。

取消跟踪命令为:TROFF(或按 F8 键)。

2. 源程序的运行

源程序运行就是计算机执行内存中 BASIC 程序,如果程序正确,则输出相应的正确结果,否则,则输出错误信息。

- (1) 运行全部程序,命令为:

RUN↵

例:运行上例求圆面积程序

OK

```
RUN↵  
S=314.16  
OK
```

(2) 运行从指定行到结尾的程序段,命令为:

```
RUN<行号>↵或 GOTO<行号>↵
```

1.2.5 源程序的保存和调用

上面讲的源程序建立,指的是把程序从键盘上逐字打入机器内存中。为了使程序永久保存下来,可以把已经建立的源程序作为程序文件,从机器内存中调出,保存到磁盘中,使用时再调入内存。具体步骤和命令是:

1. 首先把经过 DOS 格式化好的软磁盘插入驱动器 A。
2. 完成 BASIC 源程序的输入工作,如上面求圆面积的 BASIC 程序例子。
3. 键入存盘命令:

```
SAVE"<文件标识符>"[,A][,P]↵
```

若选用 A 参数,则表示以 ASCII 码的形式存放在磁盘上,对于这种文件,不但可以在 BASIC 中显示、打印和重新修改程序,而且还可以在 DOS 中显示、打印程序,在行编辑中重新修改程序。若选用 P 参数,则表示以二进制形式存放在磁盘上,无法在 BASIC、DOS 或行编辑中显示。打印和修改程序,若两个参数 A 和 P 全省略,则表示以 BASIC 设定的一种特殊代码形式,只能在 BASIC 中显示、打印和修改。

如上例,键入命令:

```
SAVE "Area"↵
```

此时,以“Area.BAS”为文件名求圆面积的程序文件存入 A 磁盘上。“.BAS”是系统自动设置的 BASIC 文件的扩展名。

4. 调用存在盘上的程序文件,可用 LOAD 命令。

例:将 A 盘上 Area.BAS 文件调入内存。

(1) 调入内存不执行程序

```
OK  
LOAD "Area"↵  
OK  
LIST↵  
10 R=10  
20 S=3.1416*R*R  
30 PRINT "S=";S  
40 END  
OK
```

(2) 调入内存马上执行程序

```
OK  
RUN "Area"↵  
S=314.16  
OK
```

1.2.6 常用 BASIC 命令简介

1. 列程序清单命令。

格式: LIST<行号 1> - <行号 2>[, <文件名>]

示例: LIST 在屏幕上列出整个程序。

LIST<行号 1> - <行号 2> 列出从行号 1 至行号 2 之间的程序。

LIST<行号 1> - 列出从行号 1 开始的所有程序。

LIST - <行号 2> 列出从程序首行至行号 2 之间的程序。

LIST<行号 1> - <行号 2>, "<文件名>" 将行号 1 至行号 2 之间的程序以指定文件名存盘。

2. 从打印机打印出程序清单。

格式: LLIST [<行号 1> - <行号 2>]

示例: LLIST 打印出全部程序。

LLIST<行号 1> 打印出行号 1 程序行。

LLIST<行号 1> - <行号 2> 打印出从行号 1 至行号 2 之间的程序。

LLIST<行号 1> - 打印出从行号 1 开始的所有程序。

LLIST - <行号 2> 打印出从首行开始至行号 2 为止的所有程序。

3. 自动产生行号。

格式: AUTO [<行号>[, <增量>]

示例: AUTO 产生的行号从第 10 开始, 行号依次为 10, 20, 30, ...。

AUTO 100, 50 产生的行号从 100 开始, 以后行号增量为 50, 即行号依次为 100, 150, 200, ...。

AUTO 500, 如果在此之前已使用了 AUTO 命令, 则此时产生的行号从 500 开始, 行号增量为上次使用的增量。

AUTO, 20 产生的行号为 0, 20, 40。

按 <Ctrl> + <Break> 就退出 AUTO 状态。

4. 重新编行号。

格式: RENUM [新行号][, 老行号][, 增量]

示例: RENUM 将全部行号重排, 使之从第 10 行开始, 行号以 10 为增量。

RENUM 300, 50 对整个程序重新编号, 第 1 行行号是 300, 行号以 50 为增量。

RENUM 1000, 900, 20 对程序 900 行以后各行重新编号, 从 1000 号开始, 并以 20 为增量。

5. 显示一行进行编辑。

格式: EDIT<行号>

6. 删除某些行。

格式: DELETE [行号 1][- 行号 2]

示例: DELETE 40 删除第 40 行。

DELETE 40 - 100 删除 40 行至 100 行。

DELETE - 40 删除 40 行前面所有行。

7. 清除内存中程序。

格式: NEW

8. 显示文件目录。

格式: FILES ["文件标识符"]

示例: FILES 显示当前盘上所有文件目录。

FILES "*" .BAS" 显示当前盘上具有扩展名为 .BAS 的所有文件目录。

FILES "B: * . *" 显示 B 盘上所有文件目录。

9. 装入一个文件到内存。

格式: LOAD "<文件标识符>" [,R]

示例: LOAD "MENU" 将当前盘上名为 MENU.BAS 的程序文件装入内存。此后可用 LIST 命令在屏幕上显示出来。

LOAD "INVENT",R 装入当前盘上 INVENT.BAS 文件并立即运行。

10. 将当前内存中程序存入磁盘。

格式: SAVE "<文件标识符>" [,A][,P]

示例: SAVE "INVENT" 将当前内存中的程序以压缩二进制格式存入当前盘,文件名为 INVENT.BAS。

SAVE "B: PROG",A 将当前程序以 ASCII 码格式存入 B 盘中,文件名为 PROG.BAS。

SAVE "SECRET",P 将当前程序以指定文件存入当前盘。以这种方式存入文件将被保护,不能再对它修改。

11. 更换文件名。

格式: NAME "<老文件名>" AS "<新文件名>"

示例: NAME "A: ACCTS.BAS" AS "LEDGER.BAS" 将 A 盘上名为 ACCTS.BAS 的文件名改为 LEDGER.BAS。

12. 删除文件。

格式: KILL "<文件标识符>"

示例: KILL "B: TEST.BAS" 删除 B 盘上名为 TEST.BAS 文件。

13. 合并程序。

格式: MERGE "NUMBRS" 将当前盘上名为 NUMBRS 的文件合并到当前内存中的程序。

14. 继续执行中断程序。

格式: CONT

15. 运行内存中程序。

格式: RUN [<行号>]

示例: RUN 运行整个内存中 BASIC 程序。

RUN <行号> 从指定行号开始运行程序。

16. 退出 BASIC,返回 DOS。

格式: SYSTEM

1.2.7 IBM PC BASIC 错误信息

代码	出错信息	说 明
1	NEXT WITHOUT FOR	没有 NEXT 对应的 FOR。
2	SYNTAX ERROR	语法错误。
3	RETURN WITHOUT GOSUB	无与 RETURN 对应的 GOSUB。
4	OUT OF DATA	DATA 语句中数据不够。
5	ILLEGAL FUNCTION CALL	调用非法功能。产生原因:(1) 求负数非整数次幂。(2) 下标为负或过大。(3) 用 DEF USR 定义段地址前,调用了 USR 函数。(4) 在 GET 或 PUT 上是负记录号。(5) 函数或语句的变量不适当。(6) 企图删除不存在的行号。(7) 企图对受保护的 BASIC 程序清单或编辑。
6	OVERFLOW	溢出。
7	OUT OF MEMORY	存储容量不够,可用 CLEAR 语句重新安排堆栈或内存。
8	UNDEFINED OUT OF NUMBER	未经定义的行号。
9	SUBSCRIPT OUT OF RANGE	下标超出范围。
10	DUPLICATE DEFINITION	重复定维。
11	DIVISION BY ZERO	除数为零。
12	ILLEGAL DIRECT	非法立即方式命令。
13	TYPE MISMATCH	类型不匹配。
14	OUT OF STRING SPACE	字符串空间不够。
15	STRING TOO LONG	字符串长度超过 255 个字符。
16	STRING FORMULA TOO COMPLEX	字符串公式太复杂。
17	CAN' T CONTINUE	已不能用 CONT 命令继续执行。
18	UNDEFINED USER FUNCTION	未定义的用户功能。
19	NO RESUME	出错陷井程序中无 RESUME 语句。
20	RESUME WITH OUT ERROR	没有出错而用了 RESUME 语句。
22	MISSING OPERAND	表达式缺少操作数。
23	LINE BUFFER OVERFLOW	行缓冲器溢出,可把该行分为多个行。
24	DEVICE TIMEOUT	设备超时,未在预定时间发出信息。
25	DEVICE FAULT	设备硬件故障。
26	FOR WITHOUT NEXT	有 FOR 而无 NEXT。
27	OUT OF PAPER	打印纸用完。
29	WHILE WITHOUT WEND	有 WHILE 而无 WEND。
30	WEND WITHOUT WHILE	有 WEND 而无 WHILE。
50	FIELD OVERFLOW	文件字段溢出。
51	INTERNAL ERROR	BASIC 内部错误。
52	BAD FILE NUMBER	文件号错误。
53	FILE NOT FOUND	文件不存在。
54	BAD FILE MODE	文件处理方式错误。
55	FILE ALREADY OPEN	文件已打开。
57	DEVICE I/O ERROR	输入/输出设备错。
58	FILE ALRENADY EXISTS	NAME 语句中的文件名与当前盘中的文件名重复。

代码	出错信息	说 明
61	DISK FULL	磁盘满。
62	INPUT PAST END	文件已读完。
63	BAD RECORD NUMBER	记录号错。
64	BAD FILE NAME	文件名不符合规定。
66	Direct statement in file	装入或连接 ASCII 格式文件时,文件中立即语句错。
67	Too many files	文件太多,目录满。
68	Device unavailable	设备失效。
69	Communication buffer overflow	通讯缓冲器溢出。
70	Disk write protect	磁盘被写保护,不能写入
71	Disk media error	磁盘未准备好。
72	Disk media error	磁盘介质错。
73	Advanced feature	使用磁盘 BASIC 时用了 BASICA 的语句。
74	Rename across disks	不能在磁盘之间进行换名。
75	Path/file access error	路径/文件存取错误。
76	Path not found	路径没有找到。
	Unprintable error	不能显示的错误,没有合适的错误信息。
	Incorrect DOS version	错误的 DOS 版本。

1.3 TURBO BASIC 上机操作

TURBO BASIC 是美国 Borland 公司于 1987 年推出集编辑、编译于一体的软件,其编译速度快,效率高,使用方便。

使用 TURBO BASIC 至少应有以下三个文件:TB.EXE: TURBO BASIC 编译器;BHELP.TBH: 揭示屏幕帮助信息;TBINST.COM: 装配程序。

1.3.1 TURBO BASIC 的进入与退出

1. 进入 TURBO BASIC

在 DOS 状态提示符下,运行 TB 文件,即:

A> TB↵

此时出现起始屏幕,如图 1.1。

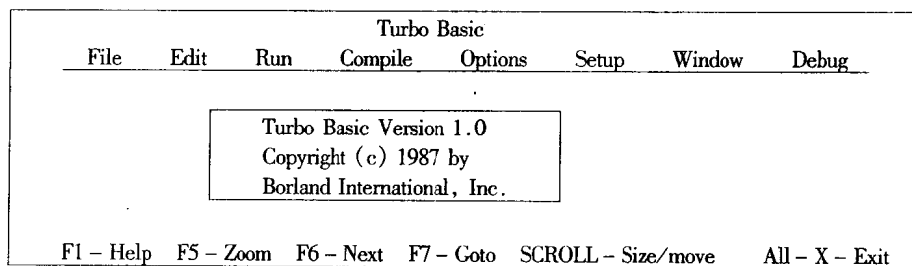


图 1.1