



计算机应用基础(下)

TURBO BASIC 语言

主 编 黎国权
副主编 王元安
主 审 张宗文



地 质 出 版 社

计算机应用基础(下)

TURBO BASIC 语言

539

Q/1-2

计算机应用基础(下)

TURBO BASIC 语言

主 编 黎国权
副主编 王元安
主 审 张宗文

地 质 出 版 社

· 北 京 ·

(京)新登字 085 号

内 容 简 介

TURBO BASIC 语言是 BORLAND 公司于 1987 年推出的 BASIC 语言新版本。它提供了良好的程序设计环境,并将编辑、编译、运行、调试等功能结合成一个完整的软件集成开发环境,是一种既适合初学者又适合有经验的编程者的语言。它不但与标准的 BASIC 语言兼容,而且有更多的扩展功能,是当今国际上最为流行的一种 BASIC 语言。书中阐述了该语言的基本知识、基本语句、分支结构、循环结构、过程和多行函数、数组、子程序、图形、磁盘文件等内容。通过该书的学习,可以了解结构化程序设计知识和设计思路,以及软件的设计和运行环境,掌握上机调试简单程序的技能。

图书在版编目(CIP)数据

计算机应用基础(下):TURBO BASIC 语言/黎国权主编.——北京:地质出版社,1994.11
ISBN 7-116-01067-X

I. 计… II. 黎… III. BASIC 语言 IV. TP312

中国版本图书馆 CIP 数据核字(94)第 12861 号

地质出版社出版发行

(100013 北京和平里七区十楼)

责任编辑:杨友爱 冯士安

北京公安大学印刷厂印刷 新华书店总店科技发行所经销

开本:787×1092 1/16 印张:9.5 字数:220000

1994 年 11 月北京第一版·1994 年 11 月北京第一次印刷

印数:1—6000 定价:9.50 元

ISBN 7-116-01067-X

G·083

前 言

《计算机应用基础》是中等专业学校非计算机专业的一门技术基础课。本书是根据国家教委对初中后四年制、五年制非计算机专业《有关计算机教学的两个项目大纲和两个课程大纲》的内容与要求而编写的。上册主要是对两个项目(标准键盘英文打字训练和计算机通用软件使用训练)进行教学和练习。下册则选择了BASIC语言这门课程,并按国家教委制定的TURBO BASIC语言的大纲进行了编写。教学时数约为60—70学时。

BASIC语言因具有简单易学、使用方便和会话性强等特点而备受广大读者的欢迎,尤其在中小学生以及大中专学生中广为流行,它成了我国中小学生普及计算机教育的主流语言。目前使用的BASIC语言版本很多,常用的有BASICA、GW BASIC、TRUE BASIC、QUICK BASIC、TURBO BASIC,不久前又推出了VISUAL BASIC。在诸多的BASIC语言版本中,目前国际上最为流行的是具有结构化程序设计功能的TURBO BASIC语言。它提供了良好的程序设计环境,并将编辑、编译、运行、调试等功能结合成一个完整的软件集成开发环境,是一种既适合初学者又适合有经验的编程者的语言,它不但与标准的BASIC语言兼容,而且有更多的扩展功能。

本书主要阐述TURBO BASIC语言的运行环境、基本知识、基本语句、分支结构、循环结构、数组、子程序、函数与过程、数据文件以及图形等内容。通过该书的学习,可以了解结构化程序设计知识和设计思想以及软件的设计与运行环境,掌握上机调试简单程序的技能。书中文字通顺,语言精练,选例适当,通俗易懂。既可以作为中专、普通中学、职业中学的教材,也可以作为初学者的自学读物,以及计算机工作人员的参考书。

本书由地质矿产部南京地质学校、长春地质学校、昆明地质学校、赣州地质学校(南方工业学校)、郑州地质学校、湖北地质学校、浙江地质学校等七所中专学校合编。赣州地质学校黎国权任主编,昆明地质学校王元安任副主编,南京地质学校张宗文任主审。参加下册编写的人员有湖北地质学校邓爱平(编写一至四章),南京地质学校徐悦(编写第五、九章)、陆敏(编写第六章),昆明地质学校王元安(编写第七章)、方洱(编写第八章)。

由于时间仓促,作者水平有限,书中有不妥之处,敬请读者批评指正。

作者

1994年11月

目 录

前 言

第一章 TURBO BASIC 语言特点与工作环境介绍

- 第一节 TURBO BASIC 语言特点 (1)
- 第二节 TURBO BASIC 界面 (2)
- 第三节 TURBO BASIC 的菜单系统 (3)
- 第四节 TURBO BASIC 的编辑器 (5)

第二章 TURBO BASIC 基础

- 第一节 TURBO BASIC 的程序结构 (8)
- 第二节 TURBO BASIC 的字符和保留字 (9)
- 第三节 TURBO BASIC 的常量和变量 (10)
- 第四节 函数 (12)
- 第五节 表达式和运算符 (16)

第三章 最基本的 BASIC 语句

- 第一节 结束(END)语句和赋值(LET)语句 (20)
- 第二节 打印(PRINT)语句 (21)
- 第三节 PRINT USING 语句 (22)
- 第四节 键盘输入(INPUT)语句 (23)
- 第五节 读数(READ)语句和数据(DATA)语句 (24)
- 第六节 恢复数据(RESTORE)语句 (26)
- 第七节 三种提供数据的语句比较 (26)
- 第八节 程序设计初步 (27)

第四章 分支程序设计

- 第一节 结构化流程图 (31)
- 第二节 转向(GOTO)语句 (33)
- 第三节 单一条件的选择结构——IF 语句 (33)
- 第四节 多条件的选择结构——IF 块语句 (33)
- 第五节 通用测试选择结构——SELECT 语句 (38)

第五章 循环程序设计

- 第一节 FOR—NEXT 循环结构 (41)
- 第二节 循环嵌套 (48)
- 第三节 DO—LOOP 循环结构 (51)

第六章 数组

- 第一节 数组介绍 (63)
- 第二节 数组的说明与使用 (65)

第三节	应用举例	(70)
第四节	数组处理补充概念	(81)
第七章	子程序、函数和过程	
第一节	子程序	(85)
第二节	自定义函数	(86)
第三节	过程	(93)
第四节	综合应用举例	(97)
第八章	数据文件	
第一节	顺序文件的操作	(109)
第二节	随机文件的操作	(114)
第三节	文件综合应用举例	(118)
第九章	图形	
第一节	显示模式与图形窗口	(120)
第二节	绘图语句及其应用	(123)
附录 I	TUBRO BASIC 错误信息	(133)
附录 II	TUBRO BASIC 过程与函数摘要	(144)

第一章 TURBO BASIC 语言特点 与工作环境介绍

第一节 TURBO BASIC 语言特点

在 IBM PC/XT、AT 及兼容机上可以运行多种 BASIC 语言,较常用的有 GW BASIC、PC BASIC、TRUE BASIC、QUICK BASIC、TURBO BASIC 及不久前刚推出的 VISUAL BASIC。它们都各有特点,在特定的时期赢得了大量的用户。目前,较常用的 BASIC 语言,主要是 TRUE BASIC、QUICK BASIC 和 TURBO BASIC,而 VISUAL BASIC 则会在未来得到很快的发展。不管是何种 BASIC 语言,我们都可以把它们简单地分为解释 BASIC 和编译 BASIC 两类。更有发展前途的是编译 BASIC, QUICK BASIC、TURBO BASIC 及 VISUAL BASIC 均为编译 BASIC。

解释 BASIC 的优点在于可以进行人机对话。编程者最喜欢解释程序,是因它响应快速,不需要编译。解释程序总是将用户的程序准备好后输入 Run,让结果显示在屏幕上。然而,解释程序并不是没有缺点。例如,在存储器中总是需要一个解释拷贝,这样会占用较多的内存空间。另外,它可能使好的编程风格得不到体现。解释程序最大的缺点是慢,它花费了大量的时间告诉机器做什么,而它们本身并不做什么。在执行一个语句时,解释程序必须首先检查每个语句的内容。然后进行所需的操作,循环体中的语句要反复检查。

编译 BASIC 克服了解释 BASIC 的缺点,同时保持与解释 BASIC 的兼容性。

TURBO BASIC 就是一种较好的编译型 BASIC 语言,它有一些突出的优点:

1. 与 PC BASIC 的兼容性好,用 PC BASIC 编制的绝大多数程序,不经任何修改就可在 TURBO BASIC 中运行。
2. 提供了良好的程序设计环境。将编辑、编译、运行、跟踪等功能结合成一个完整的工作环境,大大缩短了调试程序的时间。
3. 扩充了语言能力。

除了 IF/THEN/ELSE, WHILE/WEND, FOR/NEXT 等控制结构外,增加了 IF/END IF 等块结构语句。在 TURBO BASIC 中可以定义多种函数和过程,其中的变量可以是全局变量、局部变量或静态变量。调用函数或过程时,允许传递参数,允许递归调用;增加了一种二进制格式的数据文件,提高了处理各种文件的能力;改进了图形功能,可以重新定义屏幕上图形坐标系统,使得函数图形的绘制工作变得非常容易,可以用图案填充封闭区域,可以用虚线绘图。

4. 其它优点:是编译型语言,而且自动支持 8087 协处理器,程序执行速度快,可以支配 640K 内存;每个数组可长达 64K,允许有多个 64K 的数组;可以使用动态数组;可以同时说明数组的上、下界;可以直接调用磁盘上的 .COM 文件;可以调用 DOS 和 BIOS 中的中断;可以在程序中插入机器代码;程序中可以随意加入空行和注释以便提高程序的可读性;程序

中可以使用标号,行号可有可无,行号的实质是标号。采用 IEEE(美国电气电子工程师协会)标准的浮点格式,而不是 MICROSOFT(微软)公司的浮点格式,增加了一种长整型的数据类型。

TURBO BASIC 提供了丰富的编辑命令,它们是 WORDSTAR 的子集,能方便地在程序中进行字符串查找替换,可以把程序中的任何一段单独存入磁盘,也可以把磁盘文件插入到程序中的指定位置。

TURBO BASIC 是一个独立的系统,在 DOS 的提示符后输入 TB,并击回车键就可以调用它,进入 TURBO BASIC 后,DOS 就把控制权交给它了。所以在 TURBO BASIC 中,看不到 DOS 的提示符,也不能使用 DOS 的命令。

TURBO BASIC 将编辑、编译和运行工作融为一个整体,在它提供的工作环境中就可以做这三件事,因此大大缩短了程序调试的时间。

第二节 TURBO BASIC 界面

一、系统盘上的文件

系统盘上有下列文件:

TB.EXE 包含研制环境、编辑程序、编译程序和运行时间库,当敲入 TB 并回车后,TURBO BASIC 被启动。

*.BAS 为 TURBO BASIC 程序实例。

TBHELP.TBH 具有为使用 TURBO BASIC 的帮助屏幕。

README.COM 使用该程序读 README 文件。

README 文本文件包含了最新改动信息。

二、安装及设置环境

对于软盘体系而言,将 TB.EXE 随同许多 DOS 实用程序一道拷入一张保护磁盘中,使用驱动器 B 存入源程序和用户将产生的可执行的程序。

对于硬盘体系而言,在硬盘中建一子目录\TB 或其它目录,将 TURBO BASIC 系统盘上的所有文件拷入所建子目录中,则可以使用 TURBO BASIC,具体操作如下:

```
C>MD\TB
C>CD\TB
C>COPY A: *.* *
C>TB
```

TURBO BASIC 的环境设置提供了对 TURBO BASIC 工作环境的定义,例如 CGA 方式下是否抑制雪花干扰,是否使用 8087 或 80×87 协处理器,编译后是否在磁盘上产生 EXE 文件等,这些操作可通过下拉式菜单实现。

三、启动 TURBO BASIC

启动 DOS 后,通过软盘或硬盘就可启动 TURBO BASIC,当前盘符及子目录置成包含 TB.EXE 文件的盘符及子目录,键入 TB 并回车,即可启动 TURBO BASIC,进入 TURBO BASIC 的工作屏幕,如图 1-1 所示。

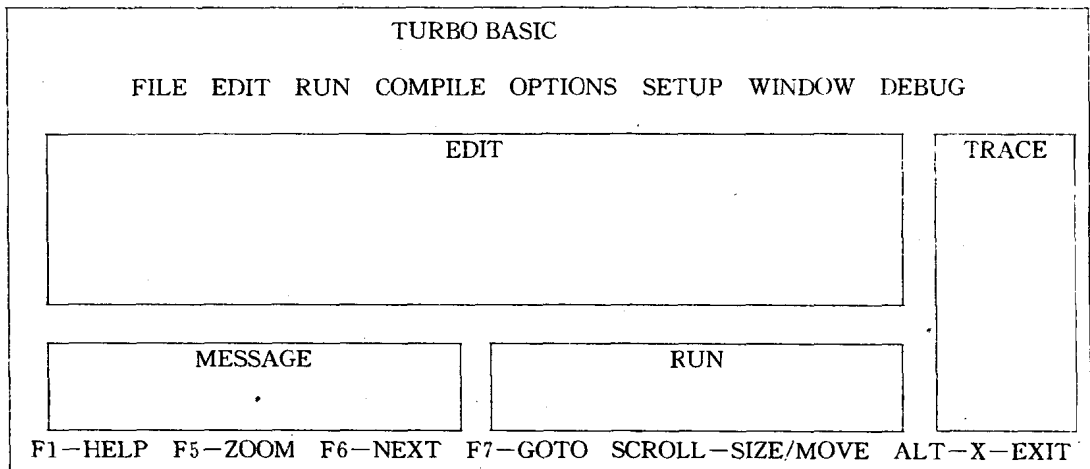


图 1-1 TURBO BASIC 工作屏幕

TURBO BASIC 的工作屏幕由三部分组成：顶端的一项是主菜单；中间是四个显示窗口；屏幕底行显示有关的帮助信息。

TURBO BASIC 的工作环境就表现为这样一个多窗口的显示屏幕和由下拉式菜单提供的命令系统。各窗口可以分别显示不同的信息，菜单系统提供各种命令和环境参数，并允许程序员修改环境参数。

在这个环境中可以编辑源程序，可以把源程序编译成目标程序，可以运行目标程序，还可以对正在运行的程序进行追踪。屏幕上有四个窗口，分别为 EDIT、MESSAGE、TRACE 和 RUN，其功能分别为：

EDIT：编辑窗口。在该窗口中可编辑修改文本文件或 BASIC 源程序。

TRACE：追踪窗口。程序追踪运行时，显示所遇到的行号、标号、函数名和过程名。

RUN：运行窗口。显示程序运行时间屏幕输出的结果。

MESSAGE：信息窗口。显示对程序编译时的有关信息。

屏幕的底部一行显示了一些帮助信息。这一行叫帮助行。其中，每一项都由两部分组成：前者表示产生这一功能需击的键位，后者表示功能。两部分的顏色不同，中间用“-”连接。如最后一项 ALT-X-EXIT 表示：按住 ALT 键后再击 X 键可退出 TURBO BASIC，回到 DOS。当进行不同的工作时，帮助行中提供的信息是不同的，它们提示现在要做某件工作时应该按什么键。

第三节 TURBO BASIC 的菜单系统

菜单系统由主菜单及各级子菜单构成，用于输入命令和设置环境参数。

主菜单含有八项，其中 EDIT、RUN、COMPILE 各是一条命令，其它五项是子菜单名，选用它们将调出下一级子菜单。有的子菜单还套有更下一级子菜单，即菜单系统具分级结构。但是最终的每一项都是命令或者对参数进行的设置。下面对菜单系统作简单介绍。

一、EDIT、RUN、COMPILE 命令

EDIT 调用 TURBO BASIC 的编辑器，光标将进入编辑窗口，可以输入、修改源程序。

RUN 运行内存中的程序,如果该程序尚未编译,则自动地编译后运行。

COMPILE 编译内存中的程序。编译生成的目标程序的三种可能:在内存中和在磁盘上生成可以独立执行的 .EXE 文件;在磁盘上生成链接文件;进行编译之前,要在 OPTION 的子菜单中进行选择,否则系统默认为编译生成的目标程序是放在内存中。

在执行 COMPIL 命令或 RUN 命令的过程中,如果发现程序中的语法错误,将会立即停止工作,自动地调用编辑器,光标出现在发生错误的语句处等待修改,而且在编辑窗口的顶端显示有关该错误的信息。

二、FILE 的子菜单——文件操作,共九项

1. Load: Load 将某个磁盘文件装入内存,选用此命令时,将要求输入一个文件名。如果不输文件名而直接按回车键,将显示工作盘上的全部 BASIC 程序文件名供选择。输入文件名后,按 Enter, TURBO BASIC 在记入磁盘的运行目录中查找文件;若找不到则产生一新文件。

2. New: 删除现存在编辑程序中的程序(文本)。

3. Save: 将内存中的源程序存入磁盘。使用此命令之前,编辑窗口中的文件名如果不是系统提供的 NONAME. BAS, 则立即进行存盘工作;否则存盘前提示你换一个新的文件名。若简单地按一下回车,则名为 NONAME. BAS 文件被存入。

4. Write to: 编辑之后,使用 Write to 选择将编辑后的文件用新的文件名写到磁盘上。

5. Main file: 当用大程序工作时,该选择可将大程序分解成易管理的块,若超过了编辑器所能容纳的字符,则不能用此选择了。分解成块的这些部分,可以作为独立的包含文件存储和编辑。这些程序块可在需要时又放到一起成为整体。

6. Director: 使用该命令可以得到一个目录清单,搜集子目录。

7. Change dir: 用来选择另外的目录。按 C 之后,屏上提示用户目录路径。该目录路径成为用户的当前目录,任何合理的路径名皆可以给出。

8. Os shell: 使用该命令临时返回 DOS 下,可以执行 DOS 的命令,用 Exit 指令返回 TURBO BASIC 体系。

9. Quit: 该选择使用户从 TURBO BASIC 退回到 DOS。屏上会提示用户现在编辑的文件是否已修改过了。若已修改过了,则提示用户现在应存盘,也可以从菜单体系的任意处退出 TURBO BASIC。

三、Options 的菜单——设置编译参数,共八项

Complie to Memory

8087 required OFF

Keyboard break OFF

Bounds OFF

Ovreflow OFF

Stack test OFF

Parameter line

Metastatements

这八项都是为编译服务的环境参数,它们将对编译生成的机器指令代码产生影响。第一项决定编译的结果是在内存中还是在磁盘上;第二至第六项用“开/关”方式(ON/OFF)设置

参数。当颜色棒(高亮度)覆盖住某项时,反复击回车键可以反复选择 ON/OFF。所显示的状态就是已选定的状态。这里只介绍三项:

Complie to Memory 选此项后将显示三种情况供你做进一步选择。

Memory 在内存中。

.EXE file 在磁盘上生成可执行文件。

Chain file 在磁盘上生成可被链接的文件。

Keyboard break OFF 不能在程序运行期间用 Ctrl-Break 操作中断运行。

ON 允许在程序运行期间 Ctrl-Break 操作中断运行,但只在程序进行输入输出工作时有效。

Bounds OFF 不能检查数组的下标超界错误。

ON 检查下标超界错误。

四、Setup 的子菜单——设置环境参数,共五项

Colors

Directories

Miscellaneous

Load options/window/setup

Save options/window/setup

这几项与 TURBO BASIC 的工作环境有关,这里只介绍二项:

Miscellaneous 含有由两项组成的下一级子菜单

Auto save edit ON 运行前自动存储修改后的源程序。

OFF 不自动存储修改后的源程序文件。

Backup source files ON 运行前自动生成源程序文件的备份文件。

OFF 不生成备份文件。

五、Window 调出用于改变各显示窗口的子菜单

六、Debug 含有两项,它决定是否对程序追踪

选择菜单有两种方法:

1. 用菜单条的头一个字母选择。处在菜单系统中的某一级时,输入要选项的头一个字母,就可以选用该项。例如在主菜单中击 E 键就可以选用 Edit 项。

2. 用颜色棒选择。处在菜单系统中时,有一个颜色与众不同的方框条覆盖某一项上,先把该方框条移到要选的项上再击回车键,就可以选用该项。

退回主菜单的方法:按 ESC 键可返回上一级子菜单,多按几次 ESC 键总可以返回主菜单。在菜单系统的任何位置,按住 ALT 键,再击主菜单中某项的头一个字母,可以立即调用该项。

第四节 TURBO BASIC 的编辑器

TURBO BASIC 的编辑程序专用来产生程序源文本。它的编辑程序指令与 Wordstar 编辑指令大致相同;不同的是,TURBO BASIC 有个恢复原状的能力,也就是说,如果没有离开所编辑的行,则可将该行上原来的改变取消,而 Wordstar 则无此能力。

当从 TURBO BASIC 的主菜单中选择 Edit 时,按 E 或将光标移动到 Edit,然后再按 Return 便可以使用编辑程序了。按 ESC 键离开编辑程序回到主菜单,Edit 窗口里的数据仍然保持在屏幕上。

一、编辑窗口顶端的显示

在主菜单中选择 Edit 命令,就调用了 TURBO BASIC 编辑器,可以输入,修改源程序。这时,在 Edit 窗口的顶端显示有下列信息:

被编辑的文件名;光标所在的行(Line)和列(Col);是处在插入状态(Insert);自动按缩排格式(Indent)写程序,Tab 有效。

说明:

1. 如果正在输入新程序,而且从未将其存盘,那么显示文件名为 noname. bas,这是 TURBO BASIC 自动提供的。如果曾将该程序以某个名字存入磁盘,或者编辑的就是从磁盘调入的旧文件,这里显示的就是该文件名。
2. 输入有两种状态供选择,插入状态或者覆盖状态。在插入状态输入的字符将插到光标上面字符的左边,而在覆盖状态下输入的字符将替换掉光标上面的字符。系统默认为插入状态,Ins 键可以在这两种状态之间进行转换。
3. 自动缩排:当输入完一行,并击回车键后,新行中的光标将与上一行的头一个字符对齐,而不一定处在新行的最左边。

二、常用的编辑命令

1. 插入:只能在插入状态下输入的新程序行,在插入状态击回车键总要产生新行。

在一行的开始击回车键,可以在此之前插入一空行。

在一行的结尾击回车键,可以在此之后插入一空行。

在一行的中间击回车键,可以把它拆成两行。

在一行的开始击退格键,可以将两行合成一行。

2. 删除:退格键(backspace):删除光标左边的字符。

del 键:删除光标处的字符。

ctrl-y:删除光标所在的行。

3. 移动光标的操作:用光标移动键可以在编辑窗口内移动光标。此外,

home 键:使光标移至一行的开始。

end 键:使光标移到一行的末尾。

pgup 键:向前翻一页。

pgdn 键:向后翻一页。

三、块操作

TURBO BASIC 的编辑工作以“块”为基础。也就是说,可以把一段程序作为一个独立的部分处理,这样的一段程序就叫一个块。可以把块移到光标所在的位置,可以在光标处将块再复制一份,也可以把块单独作为一个文件存入磁盘,也可以打印或删除块。

把一段程序定义成一个块的方法如下:当光标在这段程序开始处时击 F7 键,当光标处在这段程序末尾时击 F8 键。这样定义了块头、块尾之后,该块将以蓝色为背景,十分醒目。

下面是常用的块操作:

F7:标记块头。

F8: 标记块尾。

CTRL—KV: 把块移动到光标前。

CTRL—KC: 把块复制到光标前。

CTRL—KP: 打印块。

CTRL—KY: 删除块。

CTRL—KW: 把块作为一个文件存入磁盘。

CTRL—KR: 从磁盘上把一个文件当作一个块插入至光标所在位置。

同时间只能有一个块, 标记块头块尾次序不能颠倒。

四、查找

CTRL—QF: 可以让计算机在程序中自动地查找某个字符串, 操作步骤如下:

1. 输入 CTRL—QF。
2. 输入要查找的字符串, 并击 ENTER 键。
3. 输入范围参数, 并击回车键。这里只介绍两个范围参数:

G: 对全文进行查找。

B: 从光标处往前查找。

参数 G、B 用于指定查找范围, 如果省略它们, 则从光标处往后查找。

对全文查找时, 光标指示最后找到的一个字符串; 从光标处开始查找时, 光标将指示第一次找到的字符串, 用 CTRL—L 命令可以继续查找。

五、替换

CTRL—QA 命令可以自动地用一个新字符串替换程序中原有的旧字符串, 操作步骤如下:

1. 输入 CTRL—QA 命令。
2. 输入旧字符串, 并回车。
3. 输入新字符串, 并回车。
4. 输入范围参数, 并击回车键(同查找)。
5. 如找到, 光标将在找到的字符串与屏幕顶端之间跳动, 并显示 REPLACE(Y/N), 即请你确认是否替换掉这个字符串。这时, 输入 Y 就进行替换, 输入 N 则不进行替换。

习 题

1. 如何在 DOS 控制下调用 TURBO BASIC? 如何从 TURBO BASIC 中退回 DOS?
2. 你能一看屏幕就知道计算机是在 DOS 控制下, 还是在 TURBO BASIC 的控制下吗?
3. 写出在 TURBO BASIC 中进行编辑、编译和运行的三条命令。
4. 在 TURBO BASIC 中如何从某一窗口退到主菜单中?
5. 写出在 TURBO BASIC 中调试程序的基本过程及屏幕上可能出现的各种现象。

第二章 TURBO BASIC 基础

第一章介绍了 TURBO BASIC 的工作环境,学习了在 TURBO BASIC 中编辑、编译、运行、调试 TURBO BASIC 程序的方法,从本章开始系统学习语言本身,学习 TURBO BASIC 的各种语句、函数以及用它们编程序的方法。

第一节 TURBO BASIC 的程序结构

TURBO BASIC 的程序由若干个程序行组成,其中每一行为以下格式:

[linenumber] statement [,statement]...['comment']

或者

label:

或者

\$ metastatement

式中 linenumber 是 0 到 65535 范围内的一个整数,表示行号,放在一个程序行的最左端。同样的数字不能用来标记不同的两个程序。行号在程序中不需要按从小到大的顺序排列,它们可以任意指定,用于程序中某些地方。其作用是某一程序行的标记,便于由转移性语句控制程序转向有标号的某一语句执行。在 TURBO BASIC 中,行号与行标号作用相同。

statements 是程序的组成部分,即语句。TURBO BASIC 大约有 100 个语句类型。一个程序行上可以有 multiple 语句,也可以一条语句都没有。当有多条语句时,每两条语句间应用冒号“:”分隔。

下面是 TURBO BASIC 的合法语句行:

start;	'一行中只有标号
10	'行号
\$ INCLUDE "CONST.TBS"	'语句
20 a=a+1	'行号和语句
a=a+1;b=b+1	'两个语句
30 a=a+1;b=b+1;c=a+b	'行号和三个语句

TURBO BASIC 规定一行中最多只能容纳 249 个字符。某些编程者认为,将许多语句放在一行上是不好的,除非特殊的句法结构需要这样做。例如 IF/THEN/ELSE。

TURBO BASIC 不能解释 BASIC,它对于程序中多余的空格和程序注释不会增加运行时间,即编译程序忽略空格、注释和空行。

若一行超过 80 个字符,则不能使用户立刻在屏幕上看到一行的全部字符。若句法需要一行超过 80 个字符,例如 FIELD 语句,则需在该行的结尾放上一个短划线_ (非减号)。这样,TURBO BASIC 视下一行为上一行的继续行。例如:

```

FIELD#1,30 AS NAME $ ,30 AS address1 $ ,30 ___
AS address2 $ ,15 AS city $ ,2 AS state $ , ___
5 AS ZIP $ ,30 AS comments $

```

编译程序认为这一行是从 FIELD 开始,结束于 comments \$,其中不包括短划线(____),它只是续行的标记。

comment 是注释。注释是程序设计者对语句的说明,它不是程序本身的组成部分。在程序行中,可以在注释字符串的左侧加用单引号(')。单引号的作用是把注释与程序行的语句部分分开。单引号能够用来代替 REM。在 DATA 语句结尾处不使用单引号注释程序,因为 DATA 语句将认为单引号注释是串的一部分。

TURBO BASIC 编译程序对下面两行视为相同的行:

```

area=radius^2*3.14159      '计算面积
area=radius^2*3.14159      :REM 计算面积

```

Label 是标号,它必须单独出现在一行上,后面可以有注释。标号必须用一个字母开始,它可以含有许多字母,大写字母和小写字母没有区别。标号后面必须有一个冒号(:)。但与标号有关的语句,例如 GOSUB 则不应包括冒号(:)。例如:

```

SORTSOBRoutine:          '标号
EXITPOINT:a=a+1          '不合理,标号必须单独占一行
sortinvoices:           '标号
GOSUB sortinvoices       '引用一标号

```

\$metastatements 是与标准语句不同层次下操作的语句,它们被称之为编译元指令,总是以一个美元符号(\$)开始。标准语句在运行时间控制计算机,而元语句(metadlatements)就像选择菜单中的选择项那样,在编译时间控制编译程序。

例如,元指令 \$INCCUDE 可使编译程序在现在文件中所指示的地方插入另外文件的内容。TURBO BASIC 元指令不出现在注释内,每个行仅能有一个元指令语句。

第二节 TURBO BASIC 的字符和保留字

一、字符

TURBO BASIC 语言中允许出现的字符包括字母、数字以及一些专用字符。

1. 字母:26 个英文字母(A,B,C,……,Z),在语句、函数和变量中不区分字母的大小写,一律认为是大写字母,但在字符串中大小写字母是区分的。

2. 数字:0,1,2,……,9 共 10 个数字,加上符号“+”和“-”以及字 E,e,D,d,都用来形成数值常数。

3. 专用字符。表 3-1 列出 TURBO BASIC 专用字符及专门意义。

二、保留字

在 TURBO BASIC 中,字是由字符组成的字串,其中一些字,(即单词或缩略写法的单词)对 TURBO BASIC 有专门的意义。这些有专门意义的字就称为保留字。保留字是语言本身固有的。

保留字不能用于作标号、变量、命名的常数、过程或函数名,但保留字可以被包含在变量

名或其它名字之中。例如,END 不能作变量名,因为它是保留字,但 endhere 和 friend 中虽包

表 3-1 TURBO BASIC 专用字符

字符	说明	字符	说明
=	等于(赋值号,关系运算符)	!	感叹号(单精度说明符)
+	加号(加法运算符,并置运算符)	#	数字符号(双精度)
-	减号(减法运算符,负号)	\$	美元符(字符型后缀)
*	星号(乘法运算符)	()	括号
/	斜线(除法运算符)	[]	方括号
\	反斜线(整除运算符)		空格(间隔符)
^	插入记号(幂运算符)	,	逗号(通用空界符)
%	百分号(整数类型说明符)	.	句号(十进制小数点)
&	长整数类型说明符	'	单引号(注释定界符)
;	分号(通用定界符)	>	大于号(关系运算)
:	冒号(语句定界符)	"	双引号(串定界符)
?	问号(替代 PRINT)	_	字下划线(续行标记)
<	小于号(关系运算符)		

含 end,都可以作为变量名等使用。用 FN 作为标记物的开始两个字符,也是无效的,因为它容易与用户自定义函数相冲突。

用保留字作为标记物会产生编译时间语法错误。

保留字一般都是语句定义符及标准函数名等。在此不一一列举。

第三节 TURBO BASIC 的常量和变量

TURBO BASIC 程序处理两种数据:常量和变量。变量的值在程序运行时可变。常量的值在编译时是固定不变的,在程序运行时也不能改变,TURBO BASIC 支持三种常量:数字常量、命名常量和串常量。

一、常量

1. 数字常量

TURBO BASIC 存储和处理数字有四种不同的形式,也就是说它支持四种数字:整形、长整形、单精度实形和双精度实形。

(1) 整数

在 TURBO BASIC 中对整数的处理最快、最简单,整数范围是-32768 到 32767,整数仅需用 2 个字节(16 位二进制数)存储。

有时候需使用其它进制表示整数,TURBO BASIC 提供了十进制、十六进制、八进制和二进制表示整数:一个十六进制最多可以有 4 位数字,在十六进制数的前面有 &H 作为十六进制数字的标记,八进制数最多可有 6 位数字,在八进制数的前面有 &O、&Q 或 & 作为八进制数字的标记,二进制最多可有 16 位数字,在二进制数的前面有 &B 作为标记。

例如十进制数 256 用其它进制数表示如下:

256	(十进制)
&H100	(十六进制)
&O400	(八进制)
&400	(八进制)

&B10000000 (二进制)

各种进制的对应关系在此不详述。

(2)长整数

长整数从-2147483648到2147483648,范围比整数进行计算需要的时间更长,同整数一样,长整数不能有小数点,长整数需用4字节存储,长整数只能用十进制表示。

(3)单精度浮点数

在TURBO BASIC中单精度浮点数是最常用的一类数,单精度浮点数可以含有十进制小数点,范围为 10^{-38} 到 10^{38} ,即可以是正的,也可以是负的,虽然单精度浮点数计算速度不如整数计算速度那么快,但也不太慢,每个单精度浮点数要4字节存储,单精度浮点数精度不超过小数点后6位。

(4)双精度浮点数

双精度浮点数提供了比单精度浮点数高的精度,每个数在内存中用8个字节存入,因此计算时间比单精度浮点数长,双精度数表示的范围从 10^{-308} 到 10^{308} ,其精度可到小数点后16位。

TURBO BASIC对浮点数运算使用IEEE标准,它没有解释BASIC使用的MICROSOFT格式,因此用户必须使用特殊的MICROSOFT/IEEE翻译函数CVMS、CVMD、MKMS\$和MKMD\$读和写解释BASIC产生的随机文件中的浮点数据。

2. 命名常数

命名常数是TURBO BASIC扩展的常数类型,为了命名一个常数可以一个百分符号放到它的标识符前面赋以一个整数值就行了,命名常数不同于整型变量,整型变量的标识符在百分符号左边。例如:

```
%DEBUG = -1          '一个命名常数值-1
DEBUG% = 12409       '一个整型变量
PRINT %DEBUG,DEBUG%  %DEBUG和DEBUG%不同
```

本命名只适合于整数。

3. 串常数

串常数是用双引号括起来的一组字符。例如:

“THIS IS A STRING”

“3.14159”

“TIM JOHES, ATTORNY AT LAW”

若一个串常数放在一行的末了,则闭合双引号可以除掉,如:

PRINT“THIS IS SLOPPG BUT LOGAL”。

二、变量

在TURBO BASIC中,变量是表示一个数值或串值的标识符,在程序执行期间,变量的值可以改变。变量名要求以字母开头,后面可有字母或数字,以对变量名的长度几乎没有限制,TURBO BASIC的长变量名不会多用运行时间内存,如变量名END OF MORTHTOTALS和EMT两种变量都需要4字节的运行时内存。不能使用BASIC的保留字作为变量名。

TURBO BASIC支持五种变量名:串、整数、长整数、单精度浮点数和双精度浮点数。一