

计算机基础教育丛书

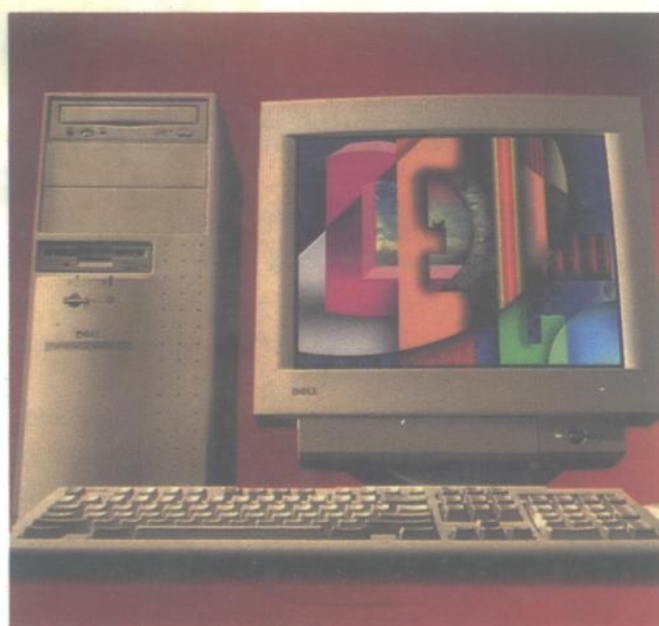
NK COMPUTING

计算机二级教程

(Pascal 程序设计)

刘瑞挺 主编

于长云 刘 璟 李秀萍 编



南开大学出版社

计算机基础教育丛书
NK COMPUTING

计算机二级教程

(Pascal 程序设计)

刘瑞挺 主编

王仁二 刘明 李永法 编

南开大学出版社

内 容 提 要

国家教委考试中心推出的计算机等级考试是一种客观、公正、科学的专门测试非计算机专业人员计算机应用知识与技能的全国范围的等级水平考试,其目的在于推动计算机知识的普及,促进计算机人才的培养。

本书是根据全国计算机等级考试委员会制定的二级考试大纲的 Pascal 语言程序考试要求编写而成的。内容包括 Pascal 及其集成环境、标准数据类型、简单语句及程序、过程和函数、动态数据结构、程序设计等。

在编写上注意了由浅入深、循序渐进、通俗易懂、繁简适当。本书适合高等院校、职大电大的非计算机专业学生,特别是理工类、财经类要求计算机较深的系科作教材使用,也可供参加二级、三级考试的应试者作为程序设计部分的主要复习资料。

计算机二级教程

(Pascal 程序设计)

刘瑞挺 主编

南开大学出版社出版

(天津八里台南开大学校内)

邮编 300071 电话 3508542

新华书店天津发行所发行

天津宝坻第四印刷厂印刷

1996年2月第1版

1996年2月第1次印刷

开本:787×1092 1/16

印张:17.25

字数:442千

印数:1-8000

ISBN 7-310-00895-2

TP·51 定价:18.60元

前言

为了适应改革开放与市场经济对计算机应用人才的迫切需要,我国高等院校越来越重视对非计算机专业的学生进行计算机基础知识的教育。

这项工作的意义很大。它正在成为我国计算机应用人才的重要培养途径。显然,计算机应用人才的宏大队伍,光靠大学里数量有限的计算机专业是远远不够的。必须面向非计算机专业,培养既熟悉自己的专业领域,又能把计算机技术同各领域的专业需要紧密地结合起来的复合人才,才能使计算机在各行各业的现代化中发挥冲锋陷阵的作用。

十多年来,高等院校非计算机专业的计算机教育取得了令人瞩目的成绩。最初仅在少数大学的理工科专业开设计算机启蒙课程。目前则在几乎所有的院校,在理工农医、财经管理、文史政法、音乐美术以及体育等各类专业都或多或少地设置了计算机基础课程。

尽管如此,非计算机专业的计算机教育仍然存在许多问题急待解决。学生多、教师少;要求多、学时少;听课多、上机少,这三多三少的现象还普遍存在。不同地区、不同院校、不同专业之间,计算机教学的开展还相当不平衡。教学质量还不好全面评估。

无论在教学对象、教学要求上,还是在教学内容、教学方法上,非计算机专业的计算机教学都与计算机专业的教学有明显的差别。我们不能生搬硬套,把计算机专业的教学计划和教材内容压缩给非计算机专业的学生。

我们知道,计算机系统有不同的层次,计算机知识有不同的台阶,计算机人才有不同的程度,计算机应用有不同的水平。因此,面对占学生总数 95% 以上的非计算机专业的学生,采取分类指导、分层安排、分级教学的方法,乃是推动这项工作深入发展的有效措施。

全国高校计算机基础教育研究会在 1986 年就提出在非计算机专业按四个层次设置计算机课程的建议,得到许多院校的积极响应,形成了计算机课程四年不断线。

国家教委工科计算机基础课程教学指导委员会自 1991 年建立以来,陆续制定了五门基础课程的教学要求和教学大纲,正在有关院校推广施行。

近年来,我国又出现了许多形式的计算机考试,其中,影响较大的有水平考试和等级考试两大类。水平考试主要面向计算机专业人员,全称是“中国计算机应用软件人员水平考试”。这种考试分为三级:程序员级、高级程序员级、系统分析员级。

等级考试则主要面向非计算机专业人员。1992 年以来,上海、北京、天津、江苏、浙江、四川等省市,组织普通高校非计算机专业的学生,开展了计算机应用知识和应用能力的等级考试。这种考试暂分一级、二级、三级,在三级考试中又分偏硬、偏软两类。天津市各高校在天津高等教育局的领导下,经过天津市等级考试委员会和各院校教学行政部门的努力,已经顺利地进行了两次等级考试。天津市计算机等级考试的特点是:笔试与上机并重,两者都通过才算通过,两者都优秀才算优秀。不仅对大量的一、二级考生进行了上机操作考试,对通过三级偏硬和偏软笔试的学生也分别组织了 Z-80、8088 以及实现算法编程与调试的上机测试。

1993年9月国家教委考试中心举行了全国计算机等级考试方案论证会。1994年3月成立了全国计算机等级考试委员会。决定在全国举办计算机等级考试,以推动计算机知识的普及,促进计算机技术的推广应用,适应社会主义经济建设的需要,为用人单位录用和考核工作人员服务。这项考试根据使用计算机的不同要求,暂定为四个等级。

这样一来,就为非计算机专业的计算机教育创造了一个良好的发展环境。这项工作正由自发阶段向自觉的阶段转变。许多学校加强了教学的领导与管理工作、增加了经费、配备了设备、调整了教学计划与教学大纲、选派了有经验的教师上课。无论是在校学生还是进入社会的工作人员,学习计算机的热情都空前高涨。

有鉴于此,南开大学出版社于1994年1月成立了计算机基础教程编辑委员会,规划了陆续出版《计算机一级教程》、《计算机二级教程》……的框架。编委会由全国计算机等级考试委员会委员、国家教委计算机科学教学指导委员会成员、全国高校计算机基础教育研究会副理事长、南开大学计算机系刘瑞挺教授担任主编。由边奠英、朱思俞、杨文太、王家骅等教授为副主编。

我们希望这一套教程能从崭新的角度,按照不同水平的应用需要,结合不同等级的考试要求,根据不同层次的教学内容组织成一系列的基础教程,以便在高校的非计算机专业中,大规模地把计算机基础教学开展起来。

《计算机二级教程》由基础知识(已出书)和程序设计两部分组成;程序设计部分共有五种语言分别成书。读者可任选一种语言与基础知识配套,即可成为一个完整的二级教程。

本书是《计算机二级教程》的Pascal语言程序设计部分。内容包括标准数据类型、常量、变量;简单语句及程序;枚举类型与子界类型;流程控制;过程和函数;数组类型;集合、记录类型;动态数据结构;程序设计等。参加编写工作的有于长云(第1、2、5、6、9、11、12章及附录)、刘璟(第7、8、10章)、李秀萍(第3、4章)。最后由边奠英、刘瑞挺教授审校并定稿。

在本书的编辑出版过程中,得到国家教委考试中心领导和全国计算机等级考试委员会专家的大力支持。天津市高等教育局及天津市计算机等级考试委员会的同志也给予热情帮助。南开大学出版社的领导和编辑都做出很大努力,对此表示衷心的感谢。

由于时间仓促,水平有限,书中必有谬误与不妥之处,敬请各位不吝批评指正。

“计算机基础教程”编委会

主 编	刘瑞挺						
副主编	边奠英	朱思俞	杨文太	王家骅			
编 委	于长云	朱守仁	曲建民	李兰友	赵志武	高福成	
	韩 劼	刘大来	李 信	沈朝辉	余金森	邵秀丽	
	李秀萍	李江卫	裴志明	李正明	何志红	张 蓓	
秘 书	李江卫						

目 录

第1章 绪 论	(1)
1.1 程序和程序语言	(1)
1.2 Pascal 语言的特点	(2)
1.3 Pascal 集成开发环境	(3)
习题一	(8)
第2章 标准数据类型、常量、变量	(9)
2.1 Pascal 程序结构	(9)
2.2 基本语法单位	(11)
2.3 整数类型	(14)
2.4 实数类型	(17)
2.5 布尔类型	(18)
2.6 字符类型	(20)
2.7 常量定义和变量说明	(21)
习题二	(23)
第3章 简单语句及程序	(25)
3.1 表达式与赋值语句	(25)
3.2 输入和输出过程	(31)
3.3 简单程序举例	(37)
3.4 上机操作	(39)
习题三	(40)
第4章 枚举类型与子界类型	(42)
4.1 怎样定义类型	(42)
4.2 枚举类型	(43)
4.3 子界类型	(46)
4.4 类型间的相互关系	(48)
习题四	(49)

第 5 章 流程控制	(51)
5.1 IF 语句	(51)
5.2 复合语句	(57)
5.3 REPEAT 语句	(59)
5.4 WHILE 语句	(62)
5.5 FOR 语句	(65)
5.6 CASE 语句	(69)
5.7 GOTO 语句和标号说明	(74)
5.8 上机操作	(77)
习题五	(79)
第 6 章 过程和函数	(82)
6.1 标准过程和标准函数	(82)
6.2 过程说明和过程语句	(84)
6.3 函数说明和函数调用	(87)
6.4 标识符的作用域	(92)
6.5 递归	(96)
6.6 过程参数和函数参数	(102)
6.7 上机操作	(105)
习题六	(110)
第 7 章 数组类型	(114)
7.1 数组类型	(114)
7.2 数组变量和下标变量	(115)
7.3 一维数组的应用	(119)
7.4 多维数组	(127)
7.5 紧凑数组类型和字符串	(134)
7.6 上机操作	(136)
习题七	(138)
第 8 章 集合类型	(139)
8.1 集合概念	(139)
8.2 集合类型	(140)
8.3 集合变量及其运算	(141)
8.4 集合类型的应用	(145)
8.5 上机操作	(147)
习题八	(148)

第 9 章 记录类型	(149)
9.1 记录类型	(149)
9.2 记录类型说明	(150)
9.3 开域语句	(155)
9.4 分量为记录类型的数组	(159)
9.5 变体记录	(166)
9.6 上机操作	(170)
习题九	(179)
第 10 章 动态数据结构	(182)
10.1 动态变量	(182)
10.2 指针类型	(184)
10.3 指针变量和对象变量	(186)
10.4 标准过程 new 和 dispose	(189)
10.5 动态变量的应用	(191)
10.6 上机操作	(198)
习题十	(200)
第 11 章 文件	(202)
11.1 顺序文件	(202)
11.2 文本文件	(208)
11.3 输入和输出	(212)
11.4 上机操作	(215)
习题十一	(220)
第 12 章 程序设计	(222)
12.1 程序设计方法	(222)
12.2 分析一个算法	(223)
12.3 程序设计举例	(227)
12.4 上机操作	(249)
习题十二	(256)
附录 A Pascal 语法图	(258)
附录 B 保留字及标准标识符	(263)
附录 C Pascal 的数据类型与语句	(264)
附录 D ASCII 字符集	(266)

绪 论

计算机系统是计算机硬件和软件的总称。硬件是系统中的物理设备,软件则泛指机器运行所需的各种程序和文档。计算机中的软件主要包括系统软件和应用软件两大部分。系统软件的代表是操作系统;应用软件则种类繁多,如 CAD、字处理、通信软件等。程序语言也是一种重要软件,它用于开发系统软件和应用程序。目前微型计算机环境已有各种程序语言软件产品,Turbo Pascal 即是其中的重要代表。

1.1 程序和程序语言

程序是一个指令序列,它表达人们控制电脑的意图。任何程序都需要设计人员去编制,也需要 CPU 去执行。程序有以下共同点:

- 指令是顺序地执行,除非遇到跳转指令。
- 处理过程会得到明确的结果。
- 程序是对数据对象进行操作。
- 对数据对象操作之前,均需说明该数据对象的特征。
- 程序本身是一个静态的整体,但执行这些指令的过程是动态的。

程序本质上是其作者与电脑交流的手段。进行交流必须借助语言,正如人类有各种语言一样,人机对话也有许多种计算机语言。但是讲汉语的人不通过学习是无法理解英语的,原因在于两种语言的词语和语法都不一样。计算机语言也面临这样的问题。为了设计出 Pascal 这类高级通用语言,计算机程序语言经历了一番演变。

不论计算机如何设计,每一台机器都有一套称为机器语言的基本指令系统。机器语言是低级语言,由于硬件的限制,它只用 0 和 1 编码。但显然,用机器语言编程序繁琐易错,并且由于没有通用的机器语言,同样的问题在不同的机器上编制的解题程序并不一样。人们研究出所谓高级语言,这些语言在语法和词汇集上有严格的规定,但形式上却类似于自然语言,易理解易修改。高级语言是为了方便开发人员的使用而设计的,电脑本身并不理解这些语言。如果要使高级语言能普及于各种机器,必须设计出一种手段,它能够将高级语言程序翻译成机器本身所能识别的机器语言,这种手段实际就是编译程序。编译程序读入用高级语言写成的源程序,然后判断其中有无拼写或语法等错误,最后输出等价的机器指令程序或指明错误信息。编译程序的不同,决定了它所属的语言软件的不同。在 Pascal 程序语言方面,美国 Borland 公司的 Turbo Pascal 属优秀软件产品,本书中绝大部分例题采用 Turbo Pascal 编制。

1.2 Pascal 语言的特点

Pascal 语言是瑞士苏黎世联邦工业大学(Eidgenossische)的 N. Wirth 教授于 1968 年提出来的,最早发表于 1971 年的“ETH”杂志上。它是以 ALGOL60 语言为基础、按照结构程序设计的原则设计出来的一种描述算法的语言,它以 17 世纪著名的数学家 Blaise Pascal 的姓名命名。

Pascal 语言比 FORTRAN 语言、ALGOL 语言、BASIC 语言等问世较晚,它的出现与程序设计发展的历史密切相关,自从 1946 年第一台计算机问世以来,计算机硬件技术迅猛发展,运算速度不断提高,存储器容量逐步加大,外设种类增多。程序设计语言也从低级向高级发展,陆续出现了机器语言、汇编语言、高级程序设计语言。60 年代,计算机应用渗透到各学科和技术领域,高级语言也随之发展为不同风格、适应于不同领域的多种语言。但此时,程序设计方法没有大的改进,程序的编制仍是一项技术性很强的工作。60 年代末至 70 年代初,开始出现大型软件系统,如操作系统、数据库等。研制这样大型的程序或程序系统,仅凭以往的经验和技术技巧是不够的,必须对程序设计方法、可靠性以及程序设计技术应如何规范化、工程化进行研究。著名的荷兰计算机科学家 E. W. 戴克斯特拉(Dijkstra)于 1968 年提出结构化程序设计概念。用结构化程序设计编写的程序,结构清晰,便于阅读,易于修改,易于验证,从而提高了程序的可靠性,并使程序设计发展为一门科学。按照这种程序设计的理论和方法,设计出的程序设计语言称为“结构程序设计语言”。利用结构程序设计语言可以有效地进行结构化程序设计。

Pascal 语言就是为适应结构化程序设计的需要而设计出来的。

设计 Pascal 主要有两个目的:

- (1)提供一种融合多种概念的教学语言。
- (2)定义一个容易在任何机器上实现的真正的标准语言。

从某种意义上讲,Pascal 是一种程序设计的混合语言或公用语言。

Pascal 语言具有数据类型丰富,程序清晰可靠,并且风格优美,编译紧凑高效,书写格式自由等特点。它特别适用于教学,也可用于编写应用软件及科学计算。

FORTRAN 等语言之所以局限于数值计算,主要在于它们的数据类型少并且构造数据结构的方法简单。而 Pascal 语言提供了丰富的数据类型和构造数据结构方法。除了整型、实型、布尔型和数组类型外,还增加了字符型、子界型、集合型、记录型、文件型和指针型等数据类型。

Pascal 语言的另一特点是它的程序清晰可靠,易读易懂,便于修改。Pascal 是一种结构程序设计语言,其方块结构由外至内按层次构成,采用逐层缩进的形式来书写,程序层次分明,静态结构非常接近于动态结构。

Pascal 的特征,正如 N. Wirth 教授指出的:

1. 变量说明是强制性的。
2. 某些保留字(例如 BEGIN,END,FOR 等)不能作标识符。
3. 分号(;)作为语句分隔符,而不是语句终止符(如同 PL/1 一样)。
4. 标准数据类型是整数、实数、字符和布尔类型。基本的数据构造工具是数组、记录、集合和文件。这些结构可以组合和嵌套以形成集合的数组、记录的文件等。数据还可以动态地分配并由指针访问。

5. 集合 SET 数据结构提供了类似 PL/1 语言里“位串”的能力。
6. 数组可以有任意的维数与任意的界,数组的界是常量,即无动态数组。
7. 如同 FORTRAN,ALGOL 和 PL/1 一样,存在转移语句。标号为无符号整数,必须加以说明。
8. 复合语句如同 ALGOL 语言,相应于 PL/1 语言中的 DO 群。
9. ALGOL 语言中的开关和 FORTRAN 语言中的计算转移语句功能在 Pascal 中由分情况语句实现。
10. 循环语句相应于 FORTRAN 语言中的 DO 循环,它的步长只能是 1(TO)或-1(DOWNT0),并且仅当控制变量的值在界内时执行。因此,被控制的语句可能完全没有执行。
11. 没有条件表达式和多重赋值。
12. 过程和函数可以递归调用。
13. 对变量不存在“OWN”属性。
14. 参数有值调用和引用调用,没有名字调用。
15. 不存在匿名分程序。这里的“分程序结构”不同于 ALGOL 和 PL/1 语言的分程序结构,即每个分程序都给一个名字,并编写成一个过程。
16. 常量、变量等所有对象必须在它们被引用之前加以说明。但允许两个例外:
 - (1)指针类型定义里的类型标识符;
 - (2)当存在向前引用时过程和函数调用。

通常选用 Pascal 语言作为学生的第一程序设计语言,旨在使学生从一开始就树立严谨规范的设计风格,为其它程序设计语言的学习打下坚实的基础。

根据大多数初学者的特点,本书首先简明地介绍 Pascal 语言的概貌知识,然后逐步展开,书中较多地采用例题,以培养学习者开发 Pascal 程序的能力及设计方法。

1.3 Pascal 集成开发环境

Turbo Pascal 是极受欢迎的集成开发环境(IDE)。在该环境下可以编辑、编译、运行和调试程序,而不必返回 DOS。这一节简单介绍有关 Turbo Pascal 5.5 的用户接口及各项功能。其主菜单如图 1-1 所示。

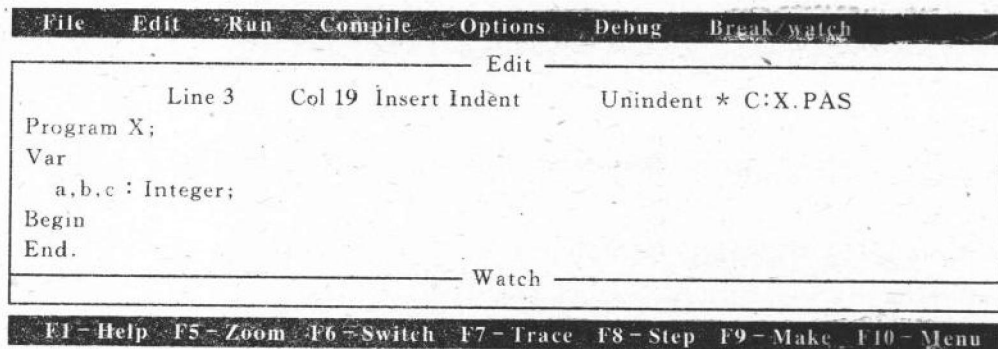


图 1-1 Turbo Pascal 5.5 主菜单

1.3.1 Turbo Pascal 启动

设 Turbo Pascal 安装在当前目录下,在 DOS 提示符下键入 Turbo Pascal,就进入 Turbo Pascal 的集成开发环境,显示图 1-1 所示菜单。屏幕顶部是主菜单,有 File,Edit,Run,Compile,Options,Debug 和 Break/Watch 等 7 个任选项。进行选择时按 F10 和箭头键,选中项成为高亮区,再按回车。也可以用首字母选择某一项(如 F 选择 File)。

1.3.2 File 菜单

File 菜单是主菜单的第一项,其功能是装一个文件到 Turbo Pascal 编辑器,改变当前磁盘驱动器及路径,激活 DOS Shell 等等。

文件子菜单如下:

Load	F3
Pick	Alt-F3
Save	F2
Write to	
Directory	
Change dir	
OS Shell	
Quit	Alt-X

1. load

Load 的功能是从磁盘上读一个文件,并装入 Turbo 编辑器。可以用 F3 调用 Load 功能。选择 Load 时,Turbo Pascal 询问被装入的文件名,此时可输入文件名,或使用 DOS 的通配符(*,?)确定一组文件。若使用通配符,Turbo Pascal 将在窗口显示所有匹配文件,再用箭头选择,选中后按回车。

2. Pick

编辑不同文件时,Turbo Pascal 可以记住已处理过的八个文件,并把文件名存于 Pick 表中。选择 Pick 功能,可以显示最近工作的八个文件名(也可用 Alt-F3 调用 Pick 功能)。使用箭头选择文件,按回车可把选中的文件装入编辑器。使用 Pick 功能可在进行编辑时在几个不同的文件间切换。

3. New

New 选择项使 Turbo Pascal 清编辑器,并置文件名为 NONAME.PAS。存储该文件时,则询问新的文件名。

4. Save

Save 功能是把编辑器的内容存到磁盘上。F2 键可直接调用 Save 功能。

5. Write to

Write to 选择项可把编辑器当前的内容写到一个用户指定的文件中。

6. Directory

Directory 功能同 Load。

7. Change dir

Change dir 功能改变装载驱动器或目录。选择此功能可以编辑当前目录及路径,也可以键

入一个新目录。

8. OS Shell

OS Shell 功能把 Turbo pascal 暂时挂起,清屏,显示 DOS 提示符。此时可运行其它程序或 DOS 命令。但此时 Turbo Pascal 仍驻留内存,可用空间比正常的少。在 DOS 提示符下键入 Exit 命令可返回 Turbo Pascal 环境。

9. Quit

选择 Quit 或 Alt-X 退出 Turbo Pascal 集成环境,返回 DOS。

1.3.3 Edit 选择

选择 Edit 进入 Edit 窗口,并激活 Turbo pascal 编辑器,菜单上高亮显示消失,在编辑窗口出现光标。在编辑窗口的顶部是一个状态行,其各项如下:

Line n 光标所在位置行数。

Col n 光标所在位置列数。

Insert 显示这一标志时,所有输入字符插入当前编辑正文。标志消失时,输入将覆盖当前编辑的正文。可按 Ctrl-V 组合进行切换。

Indent 这一标志说明自动缩排生效,可用 Ctrl-OI 组合键进行切换。

Tab 指示 Tab 模式生效,此时按 Tab 键可产生一个 Tab 符(^I),并按固定空格数右移光标。可用 Ctrl-OT 进行切换。

Fill 这一属性只在 Tab 模式下工作,此时 Turbo Pascal 每一行以一个 Tab 符或若干空格开始,且使用最少字符。这一属性用 Ctrl-OF 切换。

Unindent Unindent 功能生效时,按 Backspace 键将移动光标与当前行以上第一个缩进行对齐。

按 F10 可以由编辑返回主菜单。

1.3.4 Run 菜单

Run 菜单可以执行一个程序,并可以一次一行地单步执行。可以用 Ctrl-F9 选择此功能。

Run 子菜单包括:

Run Ctrl-F9

Program reset Ctrl-F2

Go to Cursor F4

Trace into F7

Step over F8

User Screen Alt-F5

其中选择 Run(或 Ctrl-F9)时执行当前编辑器中的程序(或指定当前的主文件)。

选择 Program reset(或按 Ctrl-F2),关闭所有打开的文件,把程序指针指到程序的第一条语句。

Go to Cursor 是 Turbo Pascal 5.5 最有用的新功能。使用此功能时,可移动光标到程序中的任意行。选择 Go to Cursor,此时正常执行程序直到光标所在行,然后返回编辑器。利用此功能可以调试或重置程序。

使用 Trace into(或按 F7),Turbo Pascal 以每次一行的方式执行程序。由此可以一行一行

了解程序的运行。

Step over 选择与 Trace into 类似,只是遇到执行过程或函数时一步完成,不进入子程序源代码。

User Screen 功能(可按 Alt-F5),可以观察程序在屏幕上如何显示。

1.3.5 Compile 菜单

Compile 菜单包含生成可连接的目标代码文件或完整的可执行程序。可以指定一个主文件(Primary File),得到有关的编译程序信息。

Compile 子菜单包括:

Compile	Alt-F9
Make	F9
Build	
Destination	Disk
Find error	
Primary file:	
Get into	

1. Compile

选择 Compile 将编译当前装入编辑器的源代码文件。若源代码文件只是一个单元,则产生一个以 .TPU 为扩展名的文件。若源代码是程序,则将其编译成可执行代码,放在内存或磁盘中的 .EXE 文件里。如果源文件用到其它单元,则其它单元必须事先编辑。使用 Alt-F9 可以启动 Compile。

2. Make

Make 性能提供一个相当复杂的编译处理。若没有定义主文件,Make 功能将编译当前装入编辑器的源代码文件。若已定义了主文件,Make 将先编译主文件,而不理睬编辑器中的源代码文件。例如:PROG1.PAS 在当前编辑器中,而 PROG.PAS 是主文件,Make 将编译 PROG.PAS,而 PROG1.PAS 仍留在编辑器中。

Make 不仅编译主文件,还编译主文件用到的所有单元。

3. Build

Build 功能与 Make 相似,不同的是所有单元不论是否过时都重新编译。

4. Destination

编译(Compiling)、制作(Making)、或重建(Building)一个程序时,可以把获得的可执行代码放到磁盘或内存中。编译到内存比较快,但需要较多内存,并且不产生可执行代码的永久磁盘备份。

5. Find error

在 DOS 提示符下运行一个程序,可能会产生错误,如:

Runtime error 200 at 0000:0099

代码 200 表示由除数为 0 引起的运行错误。0000:0099 是错误产生地址(段地址:偏移量)。为了定位错误,在 Compile 菜单下选择 Find error,并且输入地址。Turbo Pascal 在源代码中将程序定位并在屏幕上显示。

6. Primary file

该选项用于指定主文件。指定主文件后, Make, Build 功能将先处理这个文件, 不管编辑器中是什么。对于由多个源代码文件组成的程序, 这一功能尤为必要。

7. Get into

编译完一个程序后, 可以选择 Get into 打开一个信息窗口得到有关编译信息。信息窗口指出程序所需内存空间, 执行程序的可用空间, 程序已编译的行数及其它内容。

1.3.6 Options 菜单

使用 Options 菜单, 可以对编译及编程环境等许多特性进行控制和选择。这样 Turbo Pascal 编译器、连接器和环境可按用户需求的方式进行工作。Options 菜单包括:

- Compiler
- Linker
- Enviroment
- Directories
- Parameters
- Save options
- Retriere option

1.3.7 Debug 菜单

Debug 选择可以使用 Turbo Pascal 内部集成调试器, 或独立调试器所需的各种特性。

Debug 菜单包括:

Evaluate	Ctrl-F4
Call stack	Ctrl-F3
Find procedure	
Integrated debugging	On
Standalone debugging	On
Display Swapping	Smart
Refresh display	

其中 Call stack 和 Find procedure 只用于完成编译之后的程序。

调试有集成和独立两种工作方式。当选择 Intergrated debugging 方式时, 生成只能用于在集成开发环境中调试的信息。选择 Standalone debugging, 将调试信息附加在 .EXE 程序文件上, 在开发环境之外调试。

1.3.8 Break/Watch 菜单

调试程序的最简单方法就是观察程序中关键变量的值, 在程序各点之间如何变化。利用 Break/Watch 菜单中的各选择项:

Add Watch	Ctrl-F7
Delete Watch	
Edit Watch	
Remove all Watches	
Toggle Breakpoint	Ctrl-F8
Clear all break-points	

View next breakpoint

可以监视一个或多个变量,还可以在程序中设置断点,全方位控制程序的运行。

1.3.9 主菜单热键(Hot Keys)

图 1-1 给出的集成开发环境屏幕底部显示 7 个热键,现叙述如下:

- F1—Help 打开求助窗口,提供一系列关于集成开发环境信息。
- F5—Zoom Turbo Pascal 通常显示编辑和监视两个窗口。按 F5,当前活动窗口将占据整个屏幕。再按 F5 恢复。
- F6—Switch F6 键可以在编辑窗口与监视窗口之间相互切换。
- F7—Trace 按 F7 可实现程序跟踪。即编译当前程序,并以一次一行的方式开始执行。
- F8—Step 与 Trace 类似。只是在使用 Step 时 Turbo Pascal 将一步完成过程和函数调用,它可以实现程序步进。
- F9—Make 按 F9 编译当前程序。如果一个单元的 .TPU 文件的日期和时间比其所依赖的文件的日期和时间近,则该单元需要编译更新。
- F10—Menu 按 F10 则激活主菜单。

习 题 一

1. Pascal 语言有哪些基本特点?
2. Turbo Pascal 集成环境包括哪些内容? 主菜单中每个可选项的功能是什么?