

程 序 设 计

程 序 设 计

程 序 设 计

程 序 设 计

程 序 设 计

刘炳文 编

程 序 设 计

程 序 设 计

程 序 设 计

编程技巧

编程技巧

编程技巧

编程技巧

编程技巧

编程技巧

编程技巧

编程技巧



清华大学出版社

Borland C++ 实用编程技巧

刘炳文 编

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书是作者在总结近年来开发应用软件经验的基础上编写的。它介绍了在西文状态下用 C++ 开发中文用户界面的应用软件所用的一些常用编程方法和技巧。书中还以源代码的形式提供了一个进行图形和界面程序设计的类库。利用这个类库和相应的高点阵汉字库,可以十分方便地在西文状态下开发中文图形界面程序。大量的成员函数可直接用于实际的软件设计,具有极强的实用性。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Borland C++ 实用编程技巧/刘炳文编 - 北京:清华大学出版社,1995.11
ISBN 7-302-01952-5

I. B... II. 刘... III. C语言,程序设计 IV. TP312C

中国版本图书馆 CIP 数据字(95)第 17043 号

出 版 者: 清华大学出版社 (北京清华大学校内,邮编 100084)

责任编辑: 施迎难

印 刷 者: 北京市海淀区清华园印刷厂

发 行 者: 新华书店总店科技发行所

开 本: 787×1092 1/16 **印张:** 24 **字数:** 565 千字

版 次: 1995 年 10 月 第 1 版 1995 年 10 月 第 1 次印刷

书 号: ISBN 7-302-01952-5/TP·898

印 数: 0001—8000

定 价: 22.50 元

前 言

随着计算机应用范围的扩大,人们对界面的“友好”性要求越来越高。因此,目前商品化的软件无一不在界面上大作文章。

图形与界面程序设计有着十分丰富的内容,涉及多方面的知识。良好的界面环境制作工作量大,需要进行反复试验、修改,如果每个软件设计人员都从头做起,则不仅增加了许多重复劳动,而且无法提高编程效率。为此,笔者在总结近年来软件开发经验的基础上,编写了这本《Borland C++实用编程技巧》,希望它能减轻界面程序设计的工作量,在实际应用程序的开发中起到一点作用。

我国是一个以汉语为主要语言的国家,对于大多数用户来说,要读懂软件中的英文信息是比较困难的。因此,在我国,尤其是在计算机日益普及、开始进入千家万户的今天,中文界面的开发具有十分重要的意义。目前许多中文软件都是在汉字操作系统,即中文DOS下开发的,在运行时必须有汉字环境的支持。这不仅使得汉字库占用了大量的内存空间,而且有可能发生地址冲突,造成死机,给用户带来诸多不便。为了克服这一弊端,本书在介绍界面程序设计时,提供了在西文状态下输出汉字的技术。利用这种技术,可以通过调用汉字库直接读取汉字字模的方法来输出汉字,不需要汉字环境的支持。用这种方法开发的中文软件,可以直接在西文环境下运行,不必担心内存容量不够,而且绝对不会发生地址冲突,从而大大提高软件的通用性和可靠性。

本书主要介绍在西文状态下用C++开发中文界面应用软件经常用到的一些方法和技巧,目的在于帮助读者用面向对象的程序设计方法开发中文界面的应用软件。它提供了若干个类,基本上可以满足中文界面程序设计的需要。利用这些类中的成员函数,可以进行菜单、窗口程序设计,并能在程序中实现鼠标操作;不仅可以显示 16×16 点阵的汉字,而且可以输出 24×24 、 40×40 、 48×48 点阵各种字体的汉字。

面向对象的程序设计(OOP)方法越来越为广大软件工作者所接受,因为它可以对问题域进行自然的分割,以更接近人类思维的方式建立问题域模型,以便于对客观的信息实体进行结构模拟和行为模拟,从而使设计出的软件尽可能直接地描述现实世界,构造出模块化的、可重用的、便于维护的程序,并能控制软件的复杂性和降低软件费用。C++是目前较为流行和十分受欢迎的面向对象语言,它既融合了面向对象的能力,又保留了C语言的许多重要特性。

本书所讲述的内容是C++实际应用的一个重要方面,是笔者近年来开发应用程序的经验总结。它以源代码的形式提供了一个进行图形和界面程序设计的类库,系统介绍了C++的图形与界面开发技术。利用这个类库和相应的高点阵汉字库,可以很容易地西文状态下进行中文图形界面程序设计。所提供的类函数可直接用于实际的软件开发中,具有极强的实用性,能大大减轻编程的工作量。

本书中的程序全部在Borland C++ 3.1环境下调试通过,不用作任何修改,即可在

Turbo C++ 3.0 环境下编译、连接。如果在 Turbo C 2.0 中使用,则应去掉与类有关的内容(书中相当数量的程序可直接在 Turbo C2.0 环境下编译、连接)。

全书共九章。其中前四章分别介绍了 Borland C++ 3.1 的编程环境、图形函数、表示图形以及 C++ 的基本概念。第五章介绍了西文状态下各种点阵汉字的显示输出。第六、七、八章分别讲述了鼠标、窗口、菜单程序设计技术。第九章介绍界面程序设计的一些技巧。

本书注重实用,以讲清“怎样做”为主,不过多地涉及理论概念。为了能灵活地使用和改进书中所提供的类函数,最好能有一定的 C++ 程序开发实践。

为了方便读者使用,清华大学出版社软件部同时出版本书中的程序软盘(包括类及例子),需要者可直接与软件部(电话:2594891)联系。书中部分程序由田渭涛老师提供,在此谨致谢意。

衷心希望本书在图形与界面程序设计方面能对读者有所帮助。由于面向对象技术的应用尚处于发展和不断完善阶段,笔者经验不多,本书不足之处在所难免,敬请读者批评指正。

作 者

1995 年 4 月于国防大学

目 录

第一章 Borland C++ 3.1 集成开发环境	1
1.1 Borland C++ 3.1 的安装	1
1.2 IDE 的启动与退出	3
1.2.1 IDE 的启动	4
1.2.2 IDE 的退出	5
1.3 IDE 的结构	5
1.3.1 菜单	6
1.3.2 热键	8
1.3.3 窗口	10
1.3.4 状态行与对话框	13
1.4 文本编辑	16
1.4.1 文件的打开(建立)与存盘	16
1.4.2 简单编辑操作	17
1.4.3 块操作	19
1.4.4 查找与替换	21
1.5 工程文件	23
1.6 程序调试	28
1.6.1 Borland C++ 调试器	28
1.6.2 程序调试举例	35
第二章 Borland C++ 图形程序设计基础	40
2.1 IBM PC 显示屏幕	40
2.1.1 IBM PC 图形显示器	40
2.1.2 文本方式与字符坐标系	41
2.1.3 图形方式与点坐标系	42
2.2 文本方式编程	43
2.2.1 显示方式与窗口	43
2.2.2 输入输出与屏幕操作	46
2.2.3 属性控制	51
2.3 BGI 与图形方式	55
2.3.1 图形方式的初始化	55
2.3.2 错误检测与图形方式的关闭	58
2.4 像素与颜色	60

2.4.1	像素	60
2.4.2	颜色	63
2.5	BGI 绘图函数	68
2.5.1	直线	68
2.5.2	矩形和多边形	73
2.5.3	弧、圆及椭圆	76
2.6	BGI 文本与字型	80
2.6.1	文本输出函数	80
2.6.2	字体、字型和输出方式的设置	82
2.7	填充	90
2.7.1	填充图形函数	90
2.7.2	填充模式和填充颜色的设置	93
2.7.3	漫延填充	98
2.8	图形窗口与屏幕页	100
2.8.1	图形窗口	100
2.8.2	屏幕页	101
2.9	图形的存取	103
第三章	C++ 程序设计简介	108
3.1	C++ 对 C 的扩充	108
3.1.1	注释与续行	108
3.1.2	C++ 输入输出	109
3.1.3	动态存储分配	113
3.1.4	函数重载	116
3.2	类和对象	119
3.2.1	类的定义	119
3.2.2	数据与方法	122
3.3	继承与多态	125
3.3.1	基类与派生类	125
3.3.2	访问控制	127
3.3.3	多态性	131
3.4	C++ 程序设计举例	132
3.5	Borland C++ 图形程序设计工作环境	136
第四章	表示图形	139
4.1	表示图形的有关术语和程序构成	139
4.1.1	表示图形的术语	139
4.1.2	表示图形的程序结构	141

4.2	饼图	142
4.2.1	饼片的绘制	142
4.2.2	为饼片加标注	144
4.2.3	饼片的区分与图示说明	145
4.2.4	饼片的强调	146
4.3	直方图	148
4.4	饼图和直方图类	152
4.5	用成员函数绘制饼图和直方图	160
4.5.1	画饼图	160
4.5.2	画直方图	161
4.5.3	使画图通用化	164
第五章	西文操作系统下各种点阵汉字的显示输出	167
5.1	汉字代码体系	167
5.2	中文状态下编写含有汉字输入输出的程序	169
5.3	西文状态下汉字的显示输出	174
5.3.1	16×16 点阵汉字的显示输出	174
5.3.2	24×24 点阵汉字的显示输出	179
5.3.3	40×40 及 48×48 点阵汉字的显示输出	182
5.4	汉字的旋转与放大显示	186
5.5	chinese 类	193
5.6	应用举例	207
第六章	鼠标程序设计	215
6.1	鼠标概述	215
6.1.1	鼠标器发展简史	215
6.1.2	鼠标器的应用	216
6.1.3	鼠标器程序设计	217
6.2	中断调用与鼠标驱动程序	217
6.2.1	中断调用	218
6.2.2	通过中断 33H 访问鼠标驱动程序	221
6.3	鼠标的初始化	224
6.4	鼠标光标与鼠标按钮	225
6.4.1	鼠标光标控制	225
6.4.2	鼠标按钮	226
6.5	mouseobj 类	228
6.6	鼠标的键盘模拟	233
6.6.1	键盘对象的初始化	233

6.6.2 模拟鼠标光标和按钮	234
6.7 类 kbdmouseobj	238
6.8 应用举例	247
第七章 窗口程序设计	253
7.1 窗口概述	253
7.1.1 弹出式窗口	253
7.1.2 弹出式窗口基本原理	254
7.2 栈与弹出式窗口	255
7.2.1 栈的一般概念	255
7.2.2 栈在窗口操作中的应用	256
7.3 弹出式窗口类	258
7.4 弹出式窗口的建立和删除	264
7.4.1 弹出式窗口的建立	264
7.4.2 弹出式窗口的删除	266
7.5 应用举例	267
第八章 菜单程序设计	271
8.1 概述	271
8.2 文本方式下拉式菜单的程序设计	271
8.3 西文状态下中文弹出式菜单程序设计	278
8.4 菜单中的鼠标操作	288
8.5 西文状态下中文下拉式菜单程序设计	296
8.6 类 menuobj	308
8.7 应用举例	323
第九章 界面编程技巧	327
9.1 小汉字库的建立	327
9.2 利用小汉字库显示汉字	334
9.3 类 small_clib	344
9.4 清屏	356
9.5 显示空心汉字	364
9.6 屏幕“淡出”技术	368
附 录 《Borland C++ 实用编程技巧》配套软盘的使用	372

第一章 Borland C++ 3.1 集成开发环境

Borland C++ 3.1 是一个内容丰富的 C++ 程序开发软件包, 它提供了命令行编辑程序、集成开发环境软件、调试软件、性能测试软件及汇编程序。对于 C++ 程序员来说, 最重要的, 也是 Borland C++ 3.1 中功能最强的软件, 就是集成开发环境软件。集成开发环境的英文名字是 Integrated Development environment, 简称 IDE, 它是集编辑、编译、连接、调试于一体的全屏幕、多窗口、具有菜单结构的用户应用程序开发环境。IDE 在语言上完全兼容命令行方式对 C++ 的支持, 但使用起来比命令行方式容易得多, 效率高得多, Borland C++ 的用户基本上都在这个环境下开发软件。本章将介绍 IDE 的用法。

1.1 Borland C++ 3.1 的安装

Borland C++ 3.1 有两种版本, 一种是 Borland C++ 3.1, 一种是 Borland C++ 3.1 & Application frameworks, 这两种版本都可以在 IBM PC 及其全兼容计算机上运行。相对来说, Borland C++ 3.1 & Application frameworks 的功能要强得多, 本章将介绍这种版本的安装和使用。在下面的叙述中, Borland C++ 3.1 实际上指的是 Borland C++ 3.1 & Application frameworks。它要求有如下的最低配置:

- 具有 80 列显示器的 PC 机或其兼容机
- MS-DOS 或 PC-DOS 3.31 或更高版本
- 一个 5 吋高密软盘驱动器或一个 3 吋软盘驱动器(1.44MB)
- 一个硬盘, 可用空间不少于 55MB
- 640KB 的 RAM, 1MB 扩展内存

以上是最低配置要求, 在有些情况下, 可能需要增加配置。例如, 为了提高浮点运算速度, 可以安装 80X87 数学协处理器。此外, Borland C++ 3.1 提供了 Turbo C++ for Windows 的 IDE, 如果使用 Windows, 则应有 Windows 3.1, 至少 2MB 的扩展内存和与 Windows 兼容的显示器。

Borland C++ 3.1 必须安装在硬盘上才能使用(具体方法见后), 全部软件需要近 50MB 的硬盘空间, 加上工作空间, 硬盘的可用空间应不少于 55MB。

在 IDE 环境中可以使用鼠标器。如果只使用 C++, 鼠标器并不是必要的; 但如果需要在 IDE 中开发 Windows 程序, 则应安装鼠标器。

Borland C++ 3.1 包括两种不同的工作方式: 一种是集成环境方式(IDE), 一种是命令行工作方式。它们各有一套编译程序, 根据用户源程序和数据量的不同, 编译存储模式也不一样。此外, Borland C++ 3.1 还含有运行于 Windows 下的 Turbo C++ for Windows。不管使用哪种工作方式, 必须先安装在硬盘上安装 Borland C++ 3.1 软件系统。

Borland C++ 3.1 软件包存放在 18 张 5 吋高密软盘或 14 张 3 吋 1.44MB 软盘上。

由于使用了软件压缩技术,其系统软件均已被压缩,因此,不能用 COPY 命令把 Borland C++ 3.1 系统文件简单地拷贝到硬盘上,而必须用系统本身提供的安装程序 INSTALL 把商品盘上的文件“解压”后拷贝到硬盘上,这一过程叫做“安装”。INSTALL 检查所使用的硬件环境,适当地配置 Borland C++ 3.1,并能根据需要建立各级子目录,然后把所有系统文件从软盘拷贝到硬盘上。整个安装操作是自解释性的,每一步都有明确的提示信息,对这些提示作出适当的回答,系统就能自动完成安装。

Borland C++ 3.1 的安装步骤如下(假定使用 5 吋盘):

1. 将 Borland C++ 3.1 的第一张盘片(安装盘)插入驱动器 A,然后在 DOS 下键入下述命令:

A>INSTALL <CR>

按回键后,屏幕上显示初始说明信息。从这个说明中可以看出,如果安装 Borland C++ 3.1 & Application Frameworks,则硬盘空间至少需要 49MB,并外加 5MB 的工作空间。如果系统硬盘确有如此大的可用空间,则按回车键继续;否则按 Esc 键回到 DOS,删除无用的文件,准备足够的空间。

2. 如果硬盘空间够用,则按回车键,屏幕上显示图 1.1 的内容:

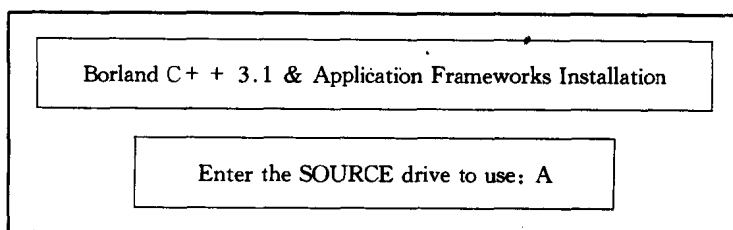


图 1.1

让用户指定安装的源驱动器名。在一般情况下,如使用 5 吋原装软盘,则源驱动器应为 A;如果使用 3 吋原装软盘,则为 B。

屏幕下方的提示区给出适当的信息,告诉用户当前应做的工作。

3. 输入源驱动器名并按回车键后,屏幕显示如图 1.2 所示。

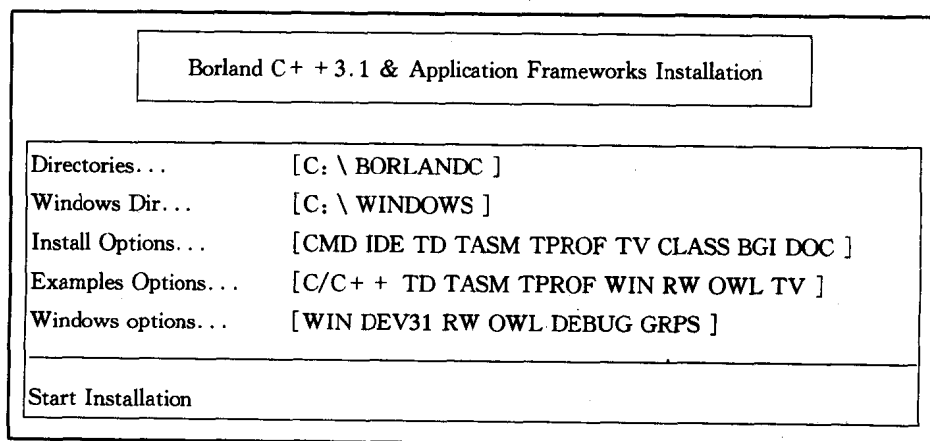


图 1.2

这是整个安装过程中最重要的部分,必须耐心而仔细地来完成,这样 Borland C++ 3.1 才会有最佳的执行情况。在这一部分中,将确定 Borland C++ 3.1 安装的驱动器和目录及其他选择项。如果想修改某一项,则只要把条形光标移到该项,然后按回车键,即可根据提示和自己的需要输入适当的信息。对于大多数选择项,可以使用系统提供的缺省值,如果不使用缺省值,则应输入自己的选择。

4. 设置好安装目录等选择项后,把条形光标移到“Start Installation”并按回车键(或者按 F9),即开始安装。安装程序依次把各个软盘上的文件拷贝到指定的硬盘和目录中,拷贝完一张后,提示插入下一张,直到所有软盘上的文件都拷贝完为止。注意,系统软盘要按编号顺序插入。

在整个安装过程中,每一步都在屏幕下部给出适当的提示,告诉用户当前可以进行什么操作。如果想中止安装过程,可随时按 Esc 键回到 DOS 下。安装结束后,INSTALL 显示 README 文件的内容,让用户了解 Borland C++ 3.1 的最新信息。

在退出 README 文件之后,如果使用 Turbo C++ for windows,则应启动 windows,并进行相应的操作。本书只介绍 Borland C++ 3.1 本身的程序设计,不涉及 windows 的内容。

为了能在任何驱动器或任何目录下使用 IDE,还需要完成以下两步操作:

1. 在 CONFIG.SYS 文件中增加以下信息(如果文件中已有该行,则此步可以省略):

```
FILES=20
```

2. 将下面的内容加到自动批处理文件 AUTOEXEC.BAT 中(假定 Borland C++ 3.1 安装在 D 盘的 BORLANDC 目录下):

```
PATH=D:\BORLANDC\BIN
```

如果启动盘根目录下没有 AUTOEXEC.BAT 文件,可以建立该文件,并把上面一行放到文件中;如果启动盘根目录下已有 AUTOEXEC.BAT 文件,且文件中有路径设置,则应把上面一行附加到原有的路径设置之后。例如假定文件中已有下面的路径设置:

```
PATH=C:\DOS;C:\;C:\QC25
```

则应把前面一行加到该行后面,变为:

```
PATH=C:\DOS;C:\;C:\QC25;D:\BORLANDC\BIN
```

当然,也可以建立专门用于运行 Borland C++ 3.1 的批处理文件。例如,可以建立一个名为 RUNBC.BAT 的批文件,内容如下:

```
PATH=D:\BORLANDC\BIN
```

在运行 Borland C++ 3.1 之前,执行上面的批命令,之后就可以在任何驱动器、任何目录下运行 Borland C++ 3.1

1.2 IDE 的启动与退出

安装好 Borland C++ 3.1,并按前面讲的方法设置路径后,就可以在 DOS 下启动 IDE 了。本书只介绍与 C++ 有关的 IDE,不涉及 Turbo C++ for windows 方面的内容。

1.2.1 IDE 的启动

启动 IDE, 实际上是执行 BORLANDC\BIN 目录下的 BC.EXE 文件。BC 命令可以有若干选择项, 这些选择项对提高 BC 命令的性能有一定的作用, 但并不是必要的。启动 IDE 的一般格式如下:

BC [选择项[选择项...]] [文件名] <CR>

其中“文件名”是要建立或编辑的 C++ 程序的名字。可以是 C++ 源程序文件(.CPP), 也可以是工程文件(.PRJ, 见 1.5 节)。如果输入时不带扩展名, 则系统自动加上扩展名.CPP。这就是说, 启动 IDE 时, 缺省的扩展名是.CPP, 如果是其他扩展名, 则必须给出; 如果要建立或编辑的文件不需要带扩展名, 则必须在文件后加上一个句点“.”。

BC 命令的选择项可以有一个, 也可以有多个, 每个选择项都必须以斜杠(/)开头, 不区分大小写字母。BC 命令的选择项包括:

1. /B: 自动装入源文件, 并编译、连接生成可执行程序, 然后自动退出 IDE, 返回 DOS。编译、连接过程中的信息, 包括错误信息和警告信息, 都被送到标准设备输出。例如:

C>BC /B MYPROG.PRJ <CR>

该命令编译、连接工程文件 MYPROG.PRJ, 生成可执行文件 MYPROG.EXE, 然后返回 DOS。再如:

C>BC /B MYPROG <CR>

将编译、连接 C++ 源程序文件 MYPROG.CPP, 生成可执行文件 MYPROG.EXE, 然后返回 DOS。

2. /M: 与/B 类似, 不同之处是, 只编译、连接修改过的源程序文件。

3. /D: 当系统配有两个显示卡(例如一个单色显示卡和一个彩色显示卡)时, /D 选择将使 Borland C++ 以双显示器的方式工作; 如果系统只配有一个显示卡, 则选择项/D 将被忽略。

4. /E 和/X: 在一般情况下, 分配内存空间时, Borland C++ 要和硬盘进行数据交换。当系统配有扩充(expanded)或扩展(extended)内存时, 使用/E 和/X 选择项可以使数据交换在内存和扩充内存及扩展内存之间进行。选择项/E 的格式如下:

BC /E[=n]

这里 n 是由 IDE 指定的用作数据交换的扩充内存页面数, 每个页面为 16K。

选择项/X 的格式如下:

BC /X[=[r][,n]]

这里 n 是 IDE 用于数据交换的扩展内存的容量(K 字节), r 是给其他程序保留的扩展内存的容量(K 字节)。

5. /RX: 如果为计算机系统设置了虚拟盘, 并准备把虚拟盘用于 IDE 的数据交换, 可以使用/RX 选择项, 其中 X 为虚拟盘的盘号。例如, 为了用虚拟盘 F 进行数据交换, 可键入:

BC /RF <CR>

6. /L:当系统配有液晶显示器(LC)时,应使用该选择项。

7. /P:如果用户程序要修改 EGA 调色板,则应使用/P 选择项,否则 IDE 屏幕和程序输出屏幕无法正常切换。在每次切换时,/P 选择项将保留或恢复相应的 EGA 调色板值。

8. /H:使用该选择项可以得到一个所有可用命令行选择项的列表,同时显示各选择项的缺省值。

9. /S:使用该选择项时,编译程序将把大部分可用内存分配给内部表。

1.2.2 IDE 的退出

利用 File 菜单中的命令,可以从 IDE 退出,回到 DOS 下。通常可分为两种情况:一种是暂时返回 DOS,在完成所需要的操作(如运行其他程序或执行 DOS 命令)后再回到 IDE;一种是真正从 IDE 返回 DOS。前者用 File 菜单中的 Dos Shell 命令来实现,后者用该菜单中的 Exit 命令来实现。具体操作如下:

1. 暂时返回 DOS(执行 Dos Shell 命令)

按下 Alt 键(不要松开),再按 F 键,然后松开,屏幕上显示 File 菜单的下拉菜单项,接着按 D 键,即可回到 DOS 命令级。此时 IDE 仍驻留在内存中,执行完 DOS 下的操作后,键入:

```
exit <CR>
```

即可回到 IDE。

2. 完全退出 IDE(执行 Exit 命令)

按 Alt-F, X(方法同上),执行 File 菜单中的 Exit 命令,将从内存中清除 IDE,回到 DOS 状态。当从一个未存盘的文件退出时,IDE 将显示一个窗口,此时可有以下三种选择:

(1) 按 Y 键,以当前状态保存该文件,退出 IDE。

(2) 按 N 键,文件不存盘,退出 IDE。

(3) 按 Esc 键,仍回到 IDE(即不退出)。

1.3 IDE 的结构

进入 IDE 后,屏幕分为三个可见的部分,即顶端的菜单条,中间的窗口区和底部的状态行,如图 1.3 所示。IDE 主要由下列成分构成:菜单,热键,对话框,窗口和状态行。

菜 单 条
窗 口 区
状 态 行

图 1.3

1.3.1 菜单

菜单操作是 IDE 的核心。在 IDE 中,几乎每种操作都要通过选择菜单及其相应的命令来实现。因此,为了能顺利地在 IDE 中进行程序设计,首先应掌握菜单及其命令的使用方法。

启动 IDE 后,在屏幕顶部显示出主菜单各个菜单的名字。IDE 的主菜单共有 11 个菜单项,如图 1.4 所示。

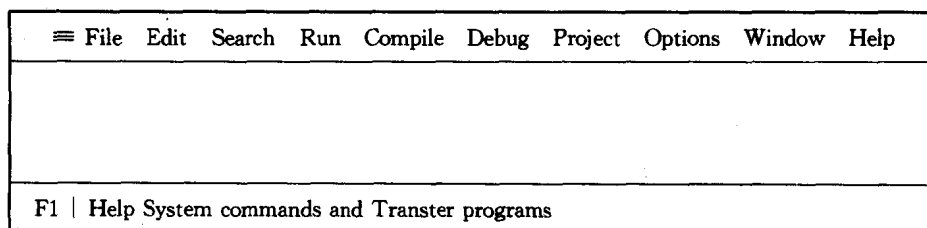


图 1.4

其名称和作用分别为:

1. $\equiv$: 系统级命令和外部程序调用。
2. File: 处理文件(包括装入、存盘、建立等),目录操作(列表、改变工作目录),退出 IDE 及调用 DOS。
3. Edit: 在编辑窗口中对文本进行剪裁、拷贝、粘贴等编辑操作。
4. Search: 在程序文件中查找文本、函数声明及出错位置。
5. Run: 运行和调试程序。
6. Compile: 用于工程文件的编译、连接。
7. Debug: 用于程序调试。
8. Project: 建立工程文件。
9. Options: 显示、改变 Borland C++ 3.1 的各种缺省选择项设置(包括存储模式、编译、连接选择项等),指定包含文件(include)和库文件(lib)的路径,保存所设置的选择项等。
10. Window: 窗口管理,包括窗口的放大、缩小、移动、关闭、“瓷砖”式排列等。
11. Help: 提供各种帮助信息。

上述各个菜单都可以显示含有多个选择项或子菜单的菜单项。为了实现所需要的某种操作,必须打开相应的菜单。例如,为了调试程序,必须打开调试(Debug)菜单;而为了运行程序,必须打开运行(Run)菜单,等等。一个菜单被打开后,将在屏幕上显示该菜单所含有的操作,这些操作叫做“菜单命令”,利用这些命令,可以实现各种功能。

IDE 使用下拉式菜单,每个菜单都可以通过“下拉”打开。只有在打开菜单后,才能选择所需要的命令,执行相应的操作。Borland C++ 提供了多种打开菜单和选择菜单命令的方法,用户可以根据自己的兴趣或习惯选用其中的一种。这些方法包括:

第一种方法,步骤如下:

1. 按 F10 键,激活菜单条。
2. 按下菜单名中变亮(或颜色不同)的字母键,即可打开相应的菜单。例如按 R 键打开 Run 菜单,按 D 键打开 Debug 菜单等等。菜单被打开后,各菜单命令都有一个变亮的字母。
3. 按下菜单命令中变亮的字母键,即可执行相应的操作。例如打开 File 菜单后,按 S 键即执行存盘操作。

第二种方法,步骤如下:

1. 按 F10 键,激活菜单条。
2. 用←或→键把条形光标(滚动条)移到所需要的菜单处,按回车键。
3. 菜单打开后,滚动条盖在第一个或上一次执行过的菜单命令上。用↑或↓键把滚动条移到所需要的菜单处,按回车键即可执行滚动条所在位置的菜单命令。

第三种方法,步骤如下:

1. 按下 Alt 键,不要松开,接着按需要打开的菜单名的第一个字母键,然后松开(Alt 键接着松开),该菜单即被打开。
2. 按菜单命令中变亮的字母键,选择指定的菜单命令。

例如,为了执行 File 菜单中的 Save 命令,可这样操作:按住 Alt 键,不要松开,接着按 F 键,先后松开 F 键和 Alt 键后再按 S 键。我们把上述过程记做:Alt-F, S。

当系统配有鼠标器时,打开菜单和选择菜单命令的操作就简单多了。只要把鼠标光标移到所需要的菜单项上,按一下鼠标左按钮,就能打开这个菜单。菜单被打开后,把鼠标光标移到需要执行的菜单命令,按鼠标左按钮,即可执行相应的操作。

每个被打开的菜单都有若干菜单项。这些菜单项分为以下几种情况:

1. 立即执行。选择这类菜单项后,立即执行相应的操作。立即执行的菜单项没有任何其他标记,但有的菜单项的右侧有键组合提示,表示该键组合与选择这个菜单项等效,这种键组合称为“热键”(见 1.3.2 中内容)。

2. 打开对话框。这类菜单项的后面带有省略号“...”,选择该菜单命令后,屏幕上弹出一个“对话框”,通过与对话框“对话”执行所需要的操作。

3. 打开下一级菜单。这类菜单项的右端有一个“└”,表示选择该菜单项时,将打开下一级菜单。

4. 无效菜单项。有的菜单项在某些情况下是无效的,这类菜单项以浅灰色显示,且没有高亮字母。选择无效菜单时不执行任何操作。

如果打开了不适当或不需要的菜单,可以按 Esc 键关闭。如果打开了多级菜单,也要用 Esc 键按与打开时相反的顺序逐级关闭。如果配有鼠标器,则只要把鼠标光标移到打开的菜单之外,然后按鼠标左按钮,即可将打开的菜单关闭。

在缺省情况下,IDE 中的鼠标一律用左按钮执行。如果习惯于用右按钮,则可选择 Options 菜单中的 Environment/Mouse 命令重新设置。

在下面的叙述中,我们用斜杠(/)来分隔菜单级别及其命令。例如,为了执行 File 菜单中的 Save 命令,记为 File/Save;而 Options/Compiler/Names 则表示执行主菜单 Options(一级菜单)下 Compiler(二级菜单)中的 Name 命令。

IDE 有三个菜单命令集,一个是 CUR(Common User Access)命令集,一个是 Alternate 命令集,一个是 Native 命令集。其中 CUA 是 windows 程序使用的标准,而 Alternate 兼容 Borland 公司以前的产品,Native 使 alternate 命令集成为 Borland C++ 和宿主 DOS 的 IDE 的缺省设置。可以通过菜单命令 Options/Environment/Preferences 来设置命令集。本书使用的是 Alternate 命令集。

1.3.2 热键

打开菜单或执行菜单命令可以使用更简单、快速的方法,即所谓“热键”或“直接键”(Shortcut key)方式。前面介绍的 Alt-F 实际上是打开 File 菜单的热键方式。使用热键方式,可以不必打开菜单,直接执行菜单命令。在对话框中,同样也可以使用热键方式(见后)。

在 Borland C++ 中,大多数菜单项都有相应的热键,这些热键是单个键或两个键的组合,列在相应的菜单命令之后,按下热键立即执行该菜单命令。例如,Run 菜单中的命令如下:

- Run Ctrl-F9
- Program reset Ctrl-F2
- Goto Cursor F4
- Trace into F7
- Step over F8
- Argument...

这里的 Ctrl-F9、Ctrl-F2、F4、F7、F8 就是相应菜单命令的热键,它们的作用与相应的菜单命令相同。Borland C++ 中常用的热键如表 1.1 所示。

表 1.1 Borland C++ 热键

1. 一般热键		
键	菜 单 命 令	功 能
F1	Help/Contents	显示帮助信息
F2	File/Save	文件存盘
F3	File/Open	弹出对话框,打开文件
F4	Run/Go to cursor	执行程序到光标所在行
F5	Window/Zoom	放大或复原活动窗口
F6	Window/Next	在打开的各窗口间循环切换
F7	Run/Trace Into	跟踪调试,进入函数内部
F8	Run/Step over	单步跟踪,不进入函数内部
F9	Compile/Make EXE file	调用工程管理程序,生成.EXE 文件
F10	(无)	激活菜单条