

朱 敏 罗 鉴 王 松 陈云章 等 编

Borland C++4.5

使用指南

Borland C++4.5

电子科技大学出版社

2146
73-8742P
C448

图 8 8 清

Borland C++ 4.5 使用指南

朱敏 罗鉴 王松 陈云章 等编

电子科技大学出版社

内 容 提 要

Borland C++ 提供了新的C++编译程序和集成调试程序,并且它支持建立可执行的Windows 和Dynamic Link Library (DLL)模块。另外,Borland C++ 4.5 包含了用于创建和维持Windows 资源(图符、菜单、对话框和其他资源)的Resource Workshop。在本书中包含了所有的这些特色,同时书中还讲述了如何创建基于DOS 的应用程序。本书还阐述了C 和C++语言的区别。

这本书的对象是下列用户:其一是要使用Borland C++编译程序的用户。Borland C++有其独特的程序设计环境和ANSI 标准C 的扩展集合。其二是已经使用了Borland C++的早期版本,而且准备充分利用4.5 版新特性的用户。

使用Borland C++有两个意义,即帮助用户学习C++,以及告诉用户如何使用Borland C++程序设计和开发环境。本书重点是C++语言。比起阅读文本编辑器、调试程序以及窗口信息视窗,用户可能对编写C++程序有更大的兴趣。当然,用户对于Borland C++中的工具和实用程序也会有兴趣,这些工具和实用程序是本书的第二个重点。

附录A 包含了面向对象的quad 类的完整清单。Borland C++重点在于创建基于Windows 的应用程序,在附录B 中提供了创建DOS 应用程序的方法,通过举例,告诉用户在创建DOS 应用程序时要进行的设置。

本书是作为一个学习工具,而不是参考手册来使用。其中的章节条理清晰,循序渐进,易于用户的学习。另外,章节都按一定次序展开,不需要用户事先了解所要覆盖的内容。

参加本书编写的作者还有:黄文波、卢少波、张涛、谭健、廖果、张竞。

Borland C++ 4.5 使用指南

朱敏 罗鉴 王松 陈云章 等编

电子科技大学出版社出版

(成都建设北路二段四号)邮编610054

成都理工学院印刷厂印刷

新华书店经销

*

开本 787×1092 1/16 印张 37.375 字数 950 千字

版次 1996 年6 月第1 版 印次 1996 年6 月第1 次印刷

印数1—5000 册

中国标准书号 ISBN 7—81043—499—3/TP·188

定价:48.00 元

目 录

第一章 Borland C++简介	(1)
§ 1.1 Borland C++的安装及环境配置	(1)
§ 1.1.1 安装Borland C++	(1)
§ 1.1.2 Borland C++的系统配置	(1)
§ 1.2 IDE 介绍	(3)
§ 1.2.1 启动IDE	(4)
§ 1.2.2 使用IDE 的菜单和窗口	(4)
§ 1.2.3 菜单系统	(5)
§ 1.2.4 窗口系统	(5)
§ 1.3 C++程序	(7)
§ 1.3.1 C++程序结构	(7)
§ 1.3.2 函数的说明、定义及调用	(11)
§ 1.3.3 C++程序	(11)
§ 1.3.4 程序的运行	(13)
§ 1.4 库函数简介	(14)
§ 1.4.1 基本输入函数	(14)
§ 1.4.2 基本输出函数	(16)
§ 1.4.3 格式转换函数	(17)
§ 1.5 C语言基础	(19)
§ 1.5.1 基本概念	(19)
§ 1.5.2 基本数据类型	(25)
§ 1.5.3 C表达式和语句	(29)
§ 1.5.4 类型转换	(33)
§ 1.5.5 C的宏定义	(35)
§ 1.6 练习	(39)
§ 1.7 小结	(40)
第二章 C语言的函数	(42)
§ 2.1 main()函数、库函数及用户自定义函数	(42)
§ 2.1.1 编写main()函数	(42)
§ 2.1.2 库函数的使用	(44)
§ 2.1.3 可用的库函数	(44)
§ 2.1.4 在程序中使用库函数	(45)
§ 2.1.5 为用户自定义函数编写模型	(46)
§ 2.1.6 向函数传递变元	(47)
§ 2.1.7 WinExec()函数	(50)
§ 2.2 函数返回值	(50)

§ 2.2.1 定义并使用函数类型	(51)
§ 2.2.2 将函数作为对象使用	(53)
§ 2.3 使用高级程序控制逻辑	(53)
§ 2.3.1 循环控制语句的编写	(53)
§ 2.3.2 goto	(54)
§ 2.3.3 break	(54)
§ 2.3.4 continue	(55)
§ 2.3.5 改变程序的执行流程	(55)
§ 2.3.6 exit()和 abort()	(55)
§ 2.4 存储类	(57)
§ 2.4.1 确定变量作用域	(57)
§ 2.4.2 确定变量的持续时间	(58)
§ 2.4.3 确定变量的连接	(58)
§ 2.5 使用可变的变元表	(59)
§ 2.5.1 设计可变的变元表	(59)
§ 2.5.2 va_...()函数	(60)
§ 2.6 练习	(64)
§ 2.7 小结	(64)
第三章 指针和导出类型	(65)
§ 3.1 标准C 导出类型	(65)
§ 3.1.1 C 的类型模式	(65)
§ 3.1.2 创建新类型	(65)
§ 3.2 C 的指针	(68)
§ 3.2.1 间接寻址	(68)
§ 3.2.2 C 的间接及地址运算符	(73)
§ 3.2.3 指向函数的指针	(75)
§ 3.2.4 指向动态内存的指针	(78)
§ 3.3 数组和串	(85)
§ 3.3.1 数组对象的说明和使用	(85)
§ 3.3.2 C 中的串	(87)
§ 3.4 结构和并集	(90)
§ 3.4.1 从不同类型创建结构	(91)
§ 3.4.2 并集	(93)
§ 3.5 练习	(95)
§ 3.6 小结	(96)
第四章 建立、编译并测试Borland C++ 程序	(97)
§ 4.1 使用多个源文件形成一个程序	(97)
§ 4.1.1 决定源文件中的内容	(97)
§ 4.1.2 建立Borland C++ 项目文件	(100)
§ 4.2 外部引用和头文件	(102)

§ 4.2.1 外部引用	(102)
§ 4.2.2 为外部模块编写头文件	(104)
§ 4.3 使用IDE 编译并运行程序	(107)
§ 4.3.1 编译、运行小型程序	(107)
§ 4.3.2 简单的错误纠正	(108)
§ 4.4 综合调试程序	(108)
§ 4.4.1 Step over 命令	(108)
§ 4.4.2 Trace into 命令	(109)
§ 4.4.3 设置、取消断点	(110)
§ 4.5 练习	(111)
§ 4.6 小结	(111)
第五章 Borland C++ I/O 函数库	(113)
§ 5.1 I/O 概念	(113)
§ 5.1.1 文件和设备	(113)
§ 5.1.2 文件和流	(114)
§ 5.1.3 文本和二进制流	(115)
§ 5.2 标准I/O 流	(116)
§ 5.2.1 格式I/O 函数	(116)
§ 5.2.2 字符I/O 函数	(125)
§ 5.3 直接文件I/O 函数	(128)
§ 5.3.1 直接I/O 概念	(128)
§ 5.3.2 读/写直接文件	(129)
§ 5.4 文件定位函数	(134)
§ 5.4.1 获取当前文件位置	(134)
§ 5.4.2 设置新的文件位置	(136)
§ 5.5 文件控制函数	(137)
§ 5.5.1 打开、关闭、控制文件	(137)
§ 5.5.2 控制文件缓冲区	(140)
§ 5.6 处理文件I/O 错误	(141)
§ 5.6.1 检查文件I/O 错误	(141)
§ 5.6.2 显示、清除文件I/O 错误	(142)
§ 5.7 练习	(142)
§ 5.8 小结	(143)
第六章 C++ 的类与导出类	(144)
§ 6.1 比较C 与C++ 的导出类型	(144)
§ 6.1.1 C++ 的“导出”概念	(144)
§ 6.1.2 C++ 的封装	(145)
§ 6.1.3 使用struct 说明类	(147)
§ 6.2 说明C++ 的类	(149)
§ 6.2.1 class 说明	(151)

§ 6.2.2 使用public、private 和protected 关键字	(165)
§ 6.3 编写类的成员函数	(166)
§ 6.3.1 将成员函数与类联系起来	(166)
§ 6.3.2 提供构造及解除函数	(169)
§ 6.4 使用友元函数	(172)
§ 6.4.1 在类中包含友元函数	(172)
§ 6.4.2 确定何时使用友元函数	(173)
§ 6.5 练习	(174)
§ 6.6 小结	(174)
第七章 C++类的创建与访问	(176)
§ 7.1 定义C++对象	(176)
§ 7.1.1 将空间类赋给类对象	(176)
§ 7.1.2 定义任意持续时间的类对象	(177)
§ 7.1.3 定义局部(auto)类对象	(178)
§ 7.1.4 定义全局(static)类对象	(185)
§ 7.2 初始化类对象	(192)
§ 7.2.1 使用构造函数来初始化类对象	(192)
§ 7.2.2 使用初始化表	(194)
§ 7.3 作用域分解运算符	(195)
§ 7.3.1 作用域分解的通常使用	(195)
§ 7.3.2 使用作用域分解进行语法控制	(197)
§ 7.3.3 使用作用域分解控制二义性	(198)
§ 7.4 C++作用域规则	(201)
§ 7.4.1 C++与C 作用域的区别	(201)
§ 7.4.2 C++ 作用域规则	(203)
§ 7.5 与C++类对象进行通信	(205)
§ 7.5.1 向对象传递信息	(205)
§ 7.5.2 this 指针	(223)
§ 7.6 使用引用运算符	(224)
§ 7.6.1 从地址运算符推出引用运算符	(224)
§ 7.6.2 引用运算符	(224)
§ 7.7 将对象作为函数参数	(225)
§ 7.7.1 按值和按引用传递对象	(225)
§ 7.7.2 从成员函数访问其他对象	(226)
§ 7.8 指向对象的指针	(227)
§ 7.8.1 何时需要指针	(227)
§ 7.8.2 说明指针和对象数组	(228)
§ 7.9 练习	(229)
§ 7.10 小结	(230)

第八章 在C++中使用重载函数和运算符	(231)
§ 8.1 重载成员函数	(231)
§ 8.1.1 C++的超载	(231)
§ 8.1.2 说明超载成员函数	(233)
§ 8.1.3 超载类的友元函数	(234)
§ 8.1.4 超载非成员函数	(239)
§ 8.2 安全类型连接	(240)
§ 8.2.1 函数的混乱	(240)
§ 8.2.2 用标准C的Include文件控制连接问题	(242)
§ 8.3 C++的超载运算符	(244)
§ 8.3.1 什么是运算符超载	(244)
§ 8.3.2 说明超载运算符函数	(248)
§ 8.3.3 超载单目和双目运算符	(255)
§ 8.4 超越下标和函数调用运算符	(256)
§ 8.4.1 超越下标运算符的使用	(257)
§ 8.4.2 超载函数调用运算符的使用	(261)
§ 8.5 练习	(264)
§ 8.6 小结	(265)
第九章 C++构造函数和解除函数	(266)
§ 9.1 构造函数与解除函数的调用	(266)
§ 9.1.1 说明构造函数和解除函数	(266)
§ 9.1.2 使用构造函数初启程序	(270)
§ 9.1.3 构造函数的调用	(273)
§ 9.1.4 调用解除函数	(278)
§ 9.2 超载构造函数	(281)
§ 9.2.1 编写缺省构造函数	(281)
§ 9.2.2 编写其他的构造函数	(282)
§ 9.2.3 确定何时需要复制构造函数	(284)
§ 9.3 使用new和delete运算符	(284)
§ 9.3.1 new和delete的一般使用	(284)
§ 9.3.2 类对象的动态创建和删除	(286)
§ 9.4 超载new和delete运算符	(287)
§ 9.4.1 超载全局运算符	(288)
§ 9.4.2 为类超载运算符	(292)
§ 9.5 练习	(295)
§ 9.6 小结	(295)
第十章 使用C++的流(stream)	(296)
§ 10.1 介绍C++流	(296)
§ 10.1.1 C++流和标准流的比较	(296)
§ 10.1.2 使用C++流进行标准I/O操作	(298)
§ 10.2 处理C++流错误	(304)

§ 10.2.1	检查C++流错误状态	(304)
§ 10.2.2	使用流状态成员函数	(306)
§ 10.3	用C++流控制数据格式	(307)
§ 10.3.1	使用<<和>>时的数据类型	(307)
§ 10.3.2	重载 <<和>>运算符	(311)
§ 10.4	使用C++流工具符(manipulator)	(313)
§ 10.4.1	理解C++工具符	(313)
§ 10.4.2	使用工具符改变状态和属性	(314)
§ 10.5	使用C++文件I/O流	(317)
§ 10.5.1	读和写fstream文件	(319)
§ 10.5.2	文件定位	(319)
§ 10.6	练习	(321)
§ 10.7	小结	(321)
第十一章	使用C++派生类	(322)
§ 11.1	再生不含继承的代码	(322)
§ 11.1.1	理解代码的再利用	(322)
§ 11.1.2	通过拼接再利用代码	(322)
§ 11.2	使用单基类	(326)
§ 11.2.1	理解继承	(326)
§ 11.2.2	说明基类和派生类	(326)
§ 11.3	使用虚函数(Virtual Function)	(332)
§ 11.3.1	使用动态联合和虚函数	(332)
§ 11.3.2	使用范围解析符来控制成员函数的访问	(334)
§ 11.4	使用多基类	(334)
§ 11.4.1	从多个基类派生出一个新类	(335)
§ 11.4.2	说明和使用虚基类	(336)
§ 11.5	从抽象类派生出新类	(338)
§ 11.5.1	理解纯粹虚函数	(338)
§ 11.5.2	实现纯粹虚函数	(338)
§ 11.6	使用带继承性的构造函数和析构函数	(339)
§ 11.6.1	初始化代码不能继承	(340)
§ 11.6.2	理解构造函数和析构函数调用的顺序	(340)
§ 11.6.3	使用虚析构函数	(340)
§ 11.7	练习	(342)
§ 11.8	小结	(343)
第十二章	对象控制、效率及其发展	(344)
§ 12.1	用户定义类型转换	(344)
§ 12.1.1	使用构造函数来转换类型	(344)
§ 12.1.2	运算符重载	(345)
§ 12.2	控制对象行为和作用	(348)

§ 12.2.1	使用友元函数提高效率	(348)
§ 12.2.2	使用友元函数来控制对象的句子结构	(349)
§ 12.2.3	使用静态存储类来避免重复说明	(351)
§ 12.2.4	使用关系(reference)和指针	(353)
§ 12.2.5	使用内联函数来减少函数调用次数	(355)
§ 12.3	类对象控制的发展	(356)
§ 12.3.1	类和函数模板	(356)
§ 12.3.2	特例处理	(358)
§ 12.4	练习	(359)
§ 12.5	小结	(359)
第十三章	Windows 程序工具	(361)
§ 13.1	WinSpector 工具	(361)
§ 13.1.1	启动WinSpector 工具	(361)
§ 13.1.2	功能对话框	(361)
§ 13.1.3	对日志文件的说明	(363)
§ 13.2	WinSight 工具	(376)
§ 13.2.1	如何启动WinSight	(376)
§ 13.2.2	选择View	(376)
§ 13.2.3	使用窗口树	(377)
§ 13.2.4	对类的操作	(379)
§ 13.2.5	选择所要跟踪的消息	(379)
§ 13.3	Windows 下的快速调试器(Turbo Debugger)	(381)
§ 13.3.1	调试初步知识	(381)
§ 13.3.2	使用Turbo Debugger	(383)
§ 13.3.3	运用先进的调试功能	(385)
§ 13.4	练习	(386)
§ 13.5	小结	(386)
第十四章	Borland C++下的Windows 编程	(387)
§ 14.1	应用程序开发者的好处	(387)
§ 14.2	理解Windows 编程环境	(388)
§ 14.2.1	Windows 是一个多任务的环境	(388)
§ 14.2.2	Windows 是一个面向对象的环境	(389)
§ 14.3	Windows 资源	(396)
§ 14.3.1	Windows 资源的类型	(396)
§ 14.3.2	资源文件的类型	(397)
§ 14.4	利用资源	(398)
§ 14.5	例程fcwin.c 需要的资源	(398)
§ 14.6	利用Borland 的资源生成器建立FCWIN 的资源	(402)
§ 14.7	练习	(405)
§ 14.8	小结	(405)

第十五章 Windows 应用程序设计	(407)
§ 15.1 应用程序fcwin.c 的设计	(407)
§ 15.2 创建Windows 应用程序文件	(435)
§ 15.2.1 熟悉Windows 程序设计环境	(435)
§ 15.2.2 建立模块定义文件	(436)
§ 15.2.3 设计程序的头文件	(437)
§ 15.2.4 建立工程文件	(439)
§ 15.3 练习	(439)
§ 15.4 小结	(440)
第十六章 用Borland C++编写Windows 应用程序	(441)
§ 16.1 怎样设计Windows 接口	(441)
§ 16.2 使用动态链接库	(459)
§ 16.2.1 了解动态链接库	(459)
§ 16.2.2 编写一个DLL 应用程序	(460)
§ 16.3 练习	(464)
§ 16.4 小结	(464)
第十七章 OLE 2.0 的C++程序设计	(465)
§ 17.1 OLE 程序设计基础	(465)
§ 17.1.1 OCF,OWL 和OLE	(465)
§ 17.1.2 OLE 应用程序类型	(465)
§ 17.2 编写第一个OWL 客户应用程序	(467)
§ 17.2.1 OWL 客户应用程序编写准备	(467)
§ 17.2.2 运行OLECONT1 应用程序	(474)
§ 17.2.3 OLECONT1 程序设计	(476)
§ 17.3 OLE 程序的动态菜单设计	(485)
§ 17.3.1 动态菜单编程基础	(485)
§ 17.3.2 一个完整程序:如何使用动态菜单	(488)
§ 17.3.3 MDI 应用程序的动态菜单技术	(491)
§ 17.3.4 建立OLE 客户应用程序的菜单	(499)
§ 17.4 OLE 复合文档的装入与保存	(517)
§ 17.4.1 运行OLECONT3 应用程序	(530)
§ 17.4.2 处理Save 或As 命令	(530)
§ 17.4.3 保存OLE 文件	(533)
§ 17.4.4 处理Open 命令	(534)
§ 17.4.5 处理New 命令	(535)
第十八章 使用ObjectWindows Library	(536)
§ 18.1 ObjectWindows 层次结构	(536)
§ 18.1.1 模块类和应用类	(536)
§ 18.1.2 窗口类	(536)
§ 18.1.3 文档类和视类	(537)

§ 18.1.4 控制类.....	(538)
§ 18.1.5 对话框类.....	(550)
§ 18.1.6 图形类.....	(553)
§ 18.1.7 打印类.....	(554)
§ 18.1.8 其他杂类.....	(554)
§ 18.2 响应Windows 消息	(554)
§ 18.3 最小的OWL 应用程序	(555)
§ 18.4 一个简单的交互式的Windows 应用程序	(558)
§ 18.5 文本编辑应用程序.....	(560)
§ 18.6 练习.....	(573)
§ 18.7 小结.....	(573)
附录A 代数类的程序	(574)
附录B 建立DOS 应用程序	(586)

第一章 Borland C++ 简介

本章中将向用户介绍如何将Borland C++安装到自己的计算机上,以及如何充分利用Borland C++的强大功能。首先,用户要学习如何安装Borland C++,以及根据自己的特定需要对它进行配置。其次,用户要学习新的集成开发环境(简称IDE)有用的特色。最后,在熟悉IDE后,用户将学习编写简单的C程序。

本章另一方面的内容是C语言的成分。用户将会学到程序如何被编辑、执行,如何处理程序中的数据,以及如何编写C语句,如何使用宏功能。

§ 1.1 Borland C++的安装及环境配置

§ 1.1.1 安装Borland C++

Borland有一个完善的安装实用程序,将用户从大型软件包安装的繁琐中解脱出来,因此安装Borland C++是很容易的事。用户要完成的就是告诉INSTALL程序自己所作的选择,然后按提示装入每张磁盘,其余工作都无需用户来做。

此外,Borland C++还可在CD-ROM上使用。从CD-ROM的安装可以大大简化安装操作,用户不必在整个安装过程中逐张插入磁盘。对于CD-ROM的Borland C++,只需在开始安装时确保CD-ROM在驱动器中即可。在本章中的屏幕图示是从CD-ROM安装程序中得到的,它与软盘版本类似,图中注明了二者的差别。

在安装Borland C++之前,用户要检查一下硬驱上的可用空间。Borland C++是个较大的软件包,当所有可选项都安装时需要40M的磁盘空间。

安装过程开始时,将标号为INSTALLATION DISK的磁盘插入软驱(或将CD-ROM盘插入CD-ROM驱动器,然后使用文件管理器(或任何等效的外壳程序),在包含安装磁盘的驱动器中,击中INSTALL.EXE文件。用户也可以在程序管理器的文件菜单中选择Run,然后打入n:install。其中n:是包含INSTALLATION DISK的软驱或包含Borland光盘的光驱。

接下来一系列的对话框将指导用户完成整个安装过程。记住重要的一点,即只选择自己要用的部分。例如,尽管所有对话框都提供了Windows NT、Window 3.1、Win32和DOS的支持,但用户可能并不需要全部的支持。

对于库函数,建议用户将其全部安装,其中包含了书中参考的大量代码。

§ 1.1.2 Borland C++的系统配置

Borland C++功能强大的一个方面是其多面性。使用Borland C++,用户并不局限于软件包所配备的程序设计环境,Borland C++允许用户按自己的需要和爱好来改变程序设计环境。用户可以通过使用Options菜单实现这一功能。

一、IDE Options 设置

Borland C++ 允许用户在集成开发环境中改变程序设计环境。在菜单系统的Options 命令下可以发现这一环境改变的便利所在。Options 菜单分为项目、环境、工具、结构表和从一个对话到另一个对话所存储的环境信息。图1.1 显示了Options 菜单。图1.2 显示了Environment Options 对话框,其中包括在Options 菜单选择中Environment 选项的设置。

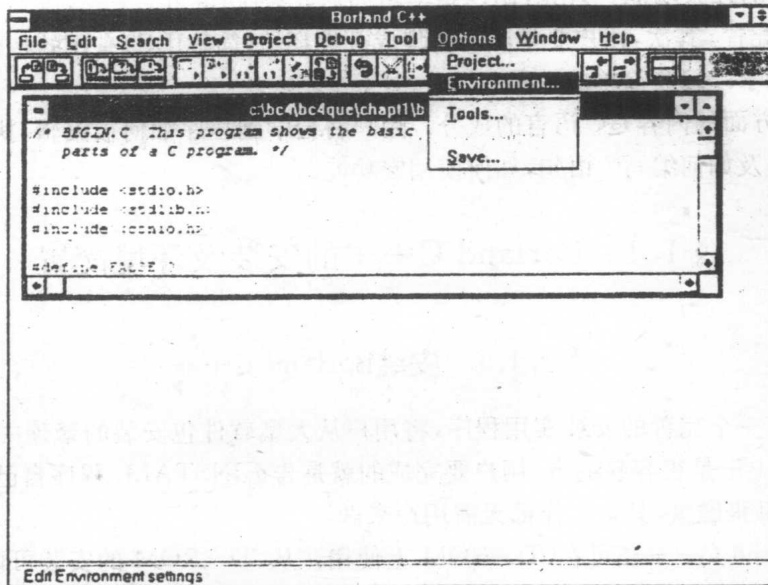


图1.1 Borland C++ Options 菜单

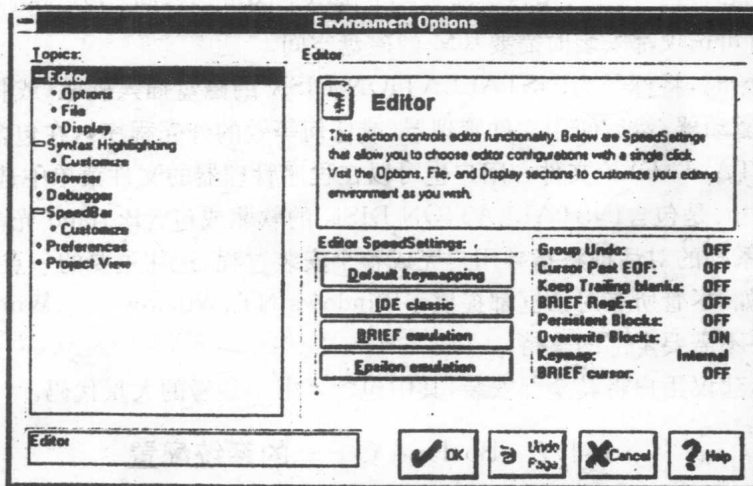


图1.2 Environment Options 对话框

二、Project 设置

Options菜单的第一个选择用来控制程序的创建。Project 选择包括下列条目：

- * 用于包含集、库、头、中间和最终输出文件的目录。
- * 用于处理一般设置和用于产生16位和32位程序的设置的编译器选项。这些设置可以调整编译器的工作方式和它包含的库。
- * 调整C++源代码解释的C++语言选项。这些设置可使用户选择或删除最新的C++语言特性。
- * 最选项，可使用户创建较小、较快，而且通常是优化了的程序。
- * 允许用户选择部分或全部错误信息的编译器信息选项。这些错误信息向用户提示在用户程序中的各种潜在的和具体的错误。
- * 连接程序选项，它可使用户调整连接OBJ目标文件的过程，这些OBJ文件是从其他OBJ目标文件、LIB库文件和RES源文件，由编译程序创建的。
- * 库管理程序选项，它可使用户指定如何将各种OBJ目标文件合并成一个LIB库文件。
- * 源程序选项，它用于确定如何编译特定版本的Windows下的Rc源文件。
- * Make文件选项，它允许用户创建MAK文件，这些MAK文件在使用Borland C++编译器的命令行版本重建程序时起辅助作用。

三、Environment 设置

Environment 设置用以管理Borland C++ IDE的工作方式，它包括以下方面：

- * Editor选项可使用户调整IDE的编辑器如何工作。用户可以选择仿真诸如BRIEF之类的普通编辑器，也可以确定如何管理制表键和文本块。用户还能够指定缺省的文件扩展名、选择光标的形状，以及为源代码选择字体。
- * 语法高亮度选择可使用户在预先设置的语法高亮度格式中进行选择，这可使用户的源代码以彩色出现。若用户对预定义的颜色设置都不满意，IDE可使用户定义自己的语法颜色。
- * 浏览器选择可使用户指定符号（数据类型、常量、变量、结构、类等等），在使用Object-Browser实用程序时进行观察。
- * 调试程序选项可使用户选择软调试方式或者硬调试方式，以及确定跟踪哪个Windows事件。
- * 速度光条选项可使用户调整并定义显示在工具条中的按钮。
- * 优先选项可使用户确定存储桌面的哪些部分和自动存储哪些成分。
- * 项目视窗选项可确定在Project Window中伴随项目文件出现的详细信息类型。

四、Tools 设置

在Options菜单上的Tools选项提供一个程序设计工具的丰富集合，其中有WinSight工具、源编译程序、库管理程序、预处理程序、浏览程序、连接程序以及Turbo汇编程序。当用户能熟练进行C或C++的程序设计，就应该准备学习如何使用这些功能强大的程序设计工具。

§ 1.2 IDE 介绍

用户通过新的IDE集成开发环境，可以在一个非常便利的环境中完成所有的程序设计工具。IDE集成了用户所需的所有工具，这样用户就可以快速有效地进行程序设计。在这一节中将向用户介绍IDE以及帮助用户使用其功能。

§ 1.2.1 启动IDE

打开Borland C++ 4.5 工作群,在Borland C++ 图符上连击。在启动IDE 时没有命令行选项及开关设置。

§ 1.2.2 使用IDE 的菜单和窗口

新IDE 比任何早期的Borland 编译器都更有力、更有用,因为它可以为Windows NT 产生32 位的程序。使用IDE,用户可以在单一的、易于使用的环境中获得全部的程序设计工具。在这里将介绍如何使用IDE 的菜单和窗口。图1.3 显示了IDE 的主要部分。

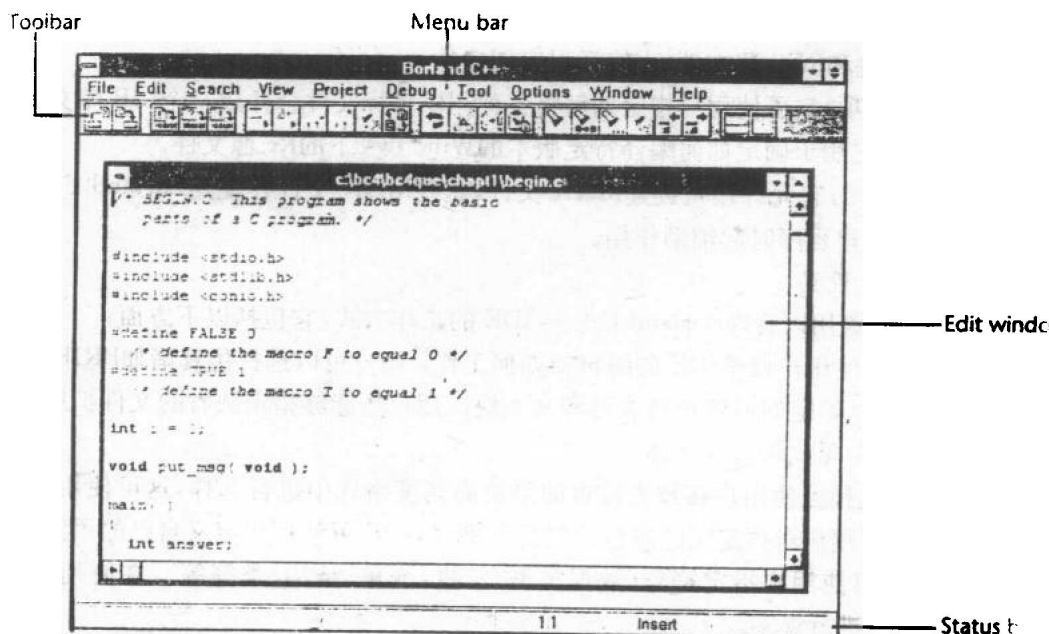


图1.3 集成开发环境(IDE)

IDE 由四个主要部分组成:

1. 菜单系统。菜单系统可使用户访问并使用Borland C++ 功能强大的实用程序及功能。用户通过从菜单条中选择一个选项来使用这些实用程序和功能。用户从图1.3 中可以看到,菜单在IDE 窗口的顶部。菜单条上的命令访问了实用程序 and 功能的每个主体工作群。
2. 窗口系统。它是IDE 最易见和使用的部分,提供了各种各样的窗口。窗口可用于编辑程序、调试程序、观察程序的运行。其他窗口可使用户为实用程序选择选项,或用户显示重要信息。图1.3 显示了一个编辑窗口,用户可以从其中创建或修改程序。在这个图中,该窗口包括了BEGIN.C 文件。
3. 状态条。在屏幕底部,状态条通常在一行上显示菜单选择帮助、当前编辑光标的位置、文件更新标记和插入/覆盖的方式。
4. 工具条。在菜单下面是一组小按钮,它提供了简捷的操作。用户在工具条按钮上移动鼠标时,IDE 会自动在状态行显示简要的描述,说明该按钮所完成的操作。工具条按钮是与上下

文有关的,IDE 根据活动窗口的类型改变它们。工具条又称快速光条。

§ 1.2.3 菜单系统

菜单系统中包含了有效地管理程序设计的任务所必需的实用程序和功能,其中包括处理文件、编辑程序,以及编译、调试、运行用户程序等功能。用户可使用三种方法进入菜单系统,其中前面两种使用键盘,第三种使用鼠标。

1. 移到菜单条,使某一命令高亮度显示,并选择它。用户可以通过按F10 键,在IDE 中从任何地方移到菜单条。当光标在菜单条上时,使用方向键使某一特定的命令以高亮度显示,然后按回车,下拉该菜单。用户可使用方向键选择某一命令,也可以打入命令名中带下划线的字母。选中后按回车,开始执行。图1.4 显示出了Project 命令的下拉菜单。

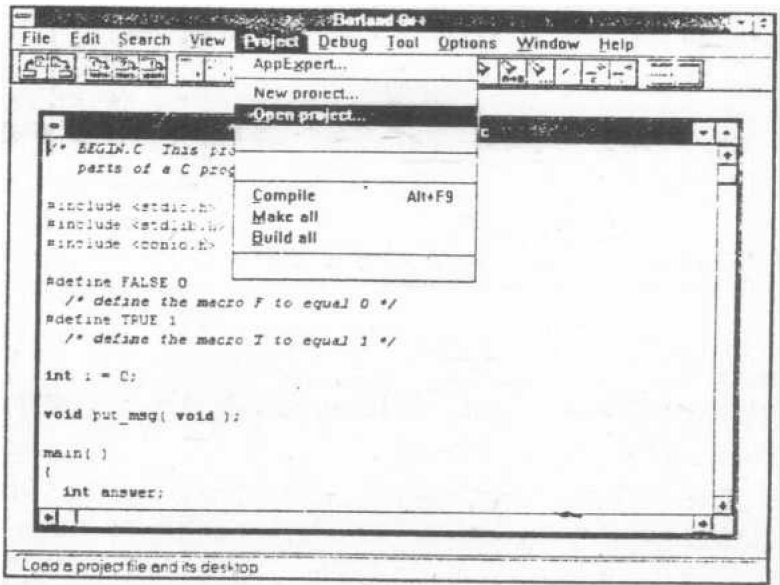


图1.4 IDE 中的一个下拉菜单

2. 使用Alt 组合键。要选择某一命令,可以先按下Alt 键,并按命令的下划线字母。当出现下拉菜单时,就使用方向键将光标移到所需的命令上,然后按回车即可。选择命令会有三种可能的结果:

- (1) 命令被执行。
- (2) 弹出一个子菜单。

(3) 出现一个对话框,其中包含一个菜单,用以提供所选命令的其他可行性选择。在命令执行之前,用户必须在对话框中给出必要的信息。

3. 在菜单条内的一个命令上敲击鼠标。若选择的命令弹出一个对话框,则仍需给出一些信息。但是,用户可以依然使用鼠标在对话框中进行选择。

§ 1.2.4 窗口系统

用户可以使用Borland C++在窗口内完成大多数操作。在IDE 中有几种可用窗口。窗口