

吴卫华 顾铁成 胡学联 潘金贵 编著
陈世福 萧菜 审校

Borland C ++ 程序开发工具

南京大学出版社

Borland C++程序开发工具

吴卫华 顾铁成 编著
胡学联 潘金贵
陈世福 萧 柔 审校

南京大学出版社

1992·南京

(苏) 新登字第 011 号

内 容 简 介

本书分四个部分，共二十四章和五个附录，系统地介绍了最新版本的面向对象程序设计语言 Borland C++ 中的资源开发工具、源程序的建立和编辑工具、各种实用工具程序的使用方法及参考信息，还介绍了如何利用资源开发工具来编制 Windows 应用程序。

本书叙述通俗易懂，内容循序渐进，示例丰富，便于自学，强调 Borland C++ 面向对象程序设计的特点，注重方法与技巧的训练，有助于培养良好的程序设计风格。

本书及其配套的《Borland C++ 程序设计教程》和《Borland C++ 实用库函数大全》是一套学习和使用 Borland C++ 语言的极好的教科书和教学参考书。既可作为高等院校计算机专业的本科生和研究生学习 C 和 C++ 语言的教材或教学参考书，亦可供有关专业的学生以及从事计算机系统软件及应用软件开发工作的工程技术人员作为 Borland C++ 软件包的使用手册。

欲购买与本书配套的例子盘和原版 Borland C++ & AF (V 2.0) 系统盘的读者可与编者联系。

Borland C++ 程序开发工具

吴卫华 顾铁成 编著
胡学联 潘金贵
陈世福 萧 柔 审校

★
南京大学出版社出版

(南京大学校内)

江苏省新华书店发行 江苏大丰印刷二厂印刷

★
开本：787 × 1092 1/6 印张：24.5 字数：607 千字

1992 年 12 月第 1 版 1992 年 12 月第 1 次印刷

印数：1—5000

ISBN 7-305-01810-4/TP·54

定价：12.50 元

责任编辑：顾其兵

前 言

一、面向对象的程序设计与 C++

面向对象程序设计方法提出了一种全新的程序设计思想，把数据和对数据的操作封装起来。目前，对面向对象的软件设计技术的研究和应用在我国正蓬勃发展，并越来越流行。

C++是一种面向对象的程序设计语言，它是 C 语言的进一步发展。C++语言是可移植的，可以很容易地将一个用 C++语言编写的程序从一个计算机系统移植到另一个计算机系统上。因此，可以毫不夸张地说，任何应用程序或系统程序均可用 C++语言来开发。

二、Borland C++

1. 语言特色

美国 Borland 国际公司在 Turbo C 的基础上，于 1990 年推出面向对象的程序设计语言——Turbo C++。它继承并发展了原来 Turbo C 集成开发环境的优良特性，并引入了面向对象的基本思想和设计方法，曾是国际上最受欢迎的面向对象的程序设计工具。

1990 年 5 月，美国 Microsoft 公司推出了 Windows 3.0，立刻风靡全球，在微机软件界引起了一场轰动。国际上许多软件公司为适应这场变革，纷纷推出了适用于 Windows 3.0 的软件包，或者对原有的流行软件进行改造，使其适用于 Windows 的版本。Borland 国际公司为适应这一潮流，于 1991 年 2 月推出了适用于开发 Windows 应用软件的 C++ 版本——Borland C++ 2.0。

除了与早期版本的 Turbo C、Turbo C++完全兼容外，Borland C++ 2.0 还具有以下新特色：

- 与 AT&T C++和 ANSI C 完全兼容：Borland C++提供了用 C++方法进行编程的全部功能，完整地实现了 AT&T C++ 2.0 版的功能。

- Borland C++也实现了迄今为止最新的 ANSI C 标准。并且完全支持 Kernigham 和 Ritchie 的定义。另外，Borland C++包含有一定的扩展，使得程序员能够用混合语言和混合模型编程来扩充 PC 机的能力。

用户可以设置编译选项，以便在碰到扩充的部分时可以得到提示或警告。也可以设置编译选项，使编译器同等对待 Borland C++中扩充的关键字和常规标识符。

Borland C++中还扩充了 ANSI C 中 #pragma 指令所具有的处理非标准的与具体实

现细节无关的功能。

借助 Borland C++, 用户可以调用 ANSI C 及 C++ 的全部功能。可以单独用 ANSI C 编程, 也可以将 C 程序转换成 C++ 程序。

为了帮助用户入门, Borland C++ 中提供了一些 C++ 类库。这些类库使用类来完成各种功能。并提供了对 C++ 2.0 版流的支持。

· 支持 Microsoft Windows 环境下的应用程序开发: 可以用 Borland C++ 编制 Windows 应用程序。Borland C++ 中添加了许多新的功能, 包括资源编译器和资源开发工具。例子盘中提供了一些用 C++ 和 C 编制的 Windows 应用程序的实例, 有助于读者理解这方面的内容。

· 提供了预编译的头文件, 利用它可以缩短程序的编译时间。

· 提供了实模式和保护模式的编译器。Borland C++ 包括四种编译器, 它们分别为实模式和保护模式下的程序员平台和命令行编译器。每类编译器都可以编译 C 和 C++ 程序。在保护方式下运行 C++ 可以取得非常高的效率, 因为它不存在内外存交换的问题。

· 类库中提供了集合、数组等类型。

· 求助系统中添加了 Windows API (应用程序接口)。

· 崭新的 Borland 程序员平台: 它的新一代的用户接口, 其性能大大优于旧式的集成开发环境, 它提供了对计算机上全范围内程序和工具的访问能力。无论是在实模式还是在保护模式下运行, 它都能支持以下功能:

鼠标器;

多重覆盖窗口;

多重文件编辑;

内部汇编码;

· 嵌入式汇编器;

集成调试器;

对大缓冲区的恢复 (undo) 和重复 (redo) 功能, 等等。

· VROOMM 提供面向对象的实时虚拟存储管理: VROOMM 使程序员能够很简单地覆盖代码。程序员只须为覆盖选择代码段, 其他均由 VROOMM 自动处理, 可使用用户的代码保存在 640k 的空间内。

· 新程序员平台的联机访问。

· 联机的快速文本求助, 用户可用拷贝的模拟程序例子试验每一个函数。

· 有着许多独立的库函数, 包括堆检测函数和完整的复数, 以及 BCD 数学函数集, 能快速完成复杂的算术运算和复数运算。

其他特征包括:

· 对 -S 选择的扩展: C 源码作为注释可加到结果汇编代码上。

· 远程对象和大容量数组

· 候补配置文件: 用户可以创建几个配置文件, 并可在需要时使用其中合适的一个。

· 用于命令行编译的响应文件。

· 快速完成复杂的算术运算和复数运算。

2. 硬件和软件环境

Borland C++运行于 IBM PC 系列计算机上, 包括 XT、AT、PS/2 和兼容机。Borland C++要求机器至少配有 3.0 以上版本的 DOS 和至少 640k 内存。监视器为 80 列即可。最低的要求是一个硬盘和一个软盘驱动器。

Borland C++中包含了利用 8087 协处理器进行浮点运算的例程。如果没有协处理器, 它也可以进行仿真运算。虽然运行 Borland C++时不必有 8087 协处理器, 但 8087 协处理器可以有效地提高程序的运行效率。

Borland C++可支持鼠标器。如果配有鼠标器, 必须与下列要求之一全兼容。

- Microsoft Mouse 6.1 版或更高版本, 或者与此兼容的 mouse 版本。
- Genus Mouse 9.0 版或更高版本。
- IMSI Mouse 6.11 版或更高版本。
- Logitech Mouse 3.4 版或更高版本。
- Mouse Systems/PC Mouse 6.22 版或更高版本。

三、内容安排

为了使我国广大的计算机工作人员能够更好地利用 Borland C++进行程序设计工作, 在南京大学出版社的支持下, 我们根据长期从事 C 语言编程和面向对象程序设计的经验, 参阅了国内外的有关软件、资料, 编译这套 Borland C++程序设计丛书, 包括《Borland C++程序设计教程》和《Borland C++程序开发工具》和《Borland C++实用库函数大全》。它们全面、系统地介绍了 Borland C++的基础知识和高级程序设计技术, 是一套引导读者进行 C、C++面向对象程序设计和进行 Windows 应用编程的参考书。

本书由四个部分组成, 分为二十四章和五个附录。

第一部分 (第一到第十一章) 介绍了 Borland C++软件包中的资源开发工具的使用方法, 以及如何利用资源开发工具来编制 Windows 应用程序。章次安排如下:

第一章“资源开发工具简介”介绍如何启动和结束资源开发工具中的某一功能。

第二章“资源和文件”对资源作一概述, 讨论可在资源开发工具中使用的文件, 告诉用户如何使用这些文件去开发 Windows 应用程序的资源。

第三章“资源管理器”介绍如何使用资源开发工具的主窗口, 即资源管理器。

第四章至第九章每章讨论一种资源开发工具编辑器的用法。

第十章“常用键及菜单”讨论如何操纵和编辑键盘加速器、串和菜单编辑器的表格, 以及如何使用在 File、Edit 和 Header 各编辑器中有公共选项的菜单。

第十一章“资源开发工具的故障诊断和错误信息”讨论如何解决在使用资源开发工具过程中所碰到的问题; 告知用户当某些故障意外地中断了编辑过程时如何恢复磁盘空间; 提供了一些内存控制信息。

为了深入了解资源开发工具, 建议读者:

- 首先学习第一章的内容, 学习如何启动和退出资源开发工具。
- 继续阅读第二章的内容, 以初步了解资源的概念, 并学会如何使用资源开发工具。

- 阅读第三章，以了解如何使用资源开发工具的主窗口，即资源管理器。
- 根据需要阅读其他章节。

第二部分（第十二章）为源程序的建立和编辑工具，它详细介绍了如何在集成开发环境中建立 Borland C++的源程序，并在需要时对它进行编辑和修改。

第三部分（第十三章到第二十四章）为 Borland C++软件包中包括的一些辅助程序开发的实用工具程序，利用它们，可以大幅度地提高程序的开发效率，减少错误。

第四部分为附录部分，其中介绍了 C++的 1.2 版流，预编译头文件，容器类库，常见问题解答，系统文件清单。

最后列出了本书的参考文献。

参加本书编著工作的人员有潘金贵、吴卫华、顾铁成、胡学联、虞育新、萧柔、潘东燕、冀惠刚、忻超、李竹华、金子方等，潘金贵同志对全书进行了仔细的修改和统编。

本书承蒙南京大学计算机系陈世福教授主审，萧柔和顾铁成同志协助校阅了有关章节，在此谨向他们深表谢意。

由于水平、时间所限，不妥之处在所难免，欢迎广大读者批评指正。

编著者

一九九二年三月于南京

目 录

前 言	1
一、面向对象的程序设计 with C++	1
二、Borland C++	1
三、内容安排	3

第一部分 Borland C++资源开发工具

第一章 资源开发工具简介	(3)
1.1 什么是资源开发工具	(3)
1.2 资源开发工具的主要成份	(3)
1.2.1 资源管理器	(3)
1.2.2 对话框编辑器	(4)
1.2.3 位图、光标和图标编辑器	(4)
1.2.4 菜单编辑器	(4)
1.2.5 其他编辑器	(4)
1.3 可处理的文件格式	(4)
1.4 运行环境及用户须知	(5)
1.5 资源开发工具的安装与启动	(5)
1.5.1 在 DOS 下启动资源开发工具	(6)
1.5.2 在 Windows 中启动资源开发工具	(6)
1.6 把图标添加到程序管理器中	(6)
1.7 从资源开发工具退出	(7)
第二章 资源和文件	(8)
2.1 什么是资源	(8)
2.2 可编辑资源	(8)
2.2.1 加速器	(8)
2.2.2 位图	(9)
2.2.3 光标	(9)
2.2.4 图标	(9)
2.2.5 对话框	(9)
2.2.6 菜单	(10)
2.2.7 串	(10)
2.3 可编辑或存储的文件	(10)

2.3.1 可执行文件	(11)
2.3.2 资源文件	(12)
2.3.3 动态链接库	(12)
2.3.4 资源与对话框说明文件	(12)
2.3.5 位图、光标和图标文件	(13)
2.3.6 头文件	(13)
第三章 资源管理器	(15)
3.1 资源管理器的组成	(15)
3.2 启动编辑器	(16)
3.3 创建新资源	(16)
3.4 访问现存资源	(16)
3.4.1 选择欲显示的资源种类	(17)
3.4.2 打开已存在的文件	(17)
3.4.3 编辑资源	(18)
3.4.4 删除资源	(18)
3.4.5 拷贝资源	(19)
3.4.6 资源换名	(19)
3.4.7 备份文件	(19)
3.5 创建新文件	(19)
3.6 关闭文件	(20)
第四章 加速器编辑器	(21)
4.1 创建加速器	(21)
4.1.1 游历编辑器	(21)
4.1.2 编辑文本	(21)
4.1.3 加速表	(21)
4.2 头文件	(23)
第五章 位图、光标和图标编辑器	(24)
5.1 文件	(24)
5.1.1 位图文件	(24)
5.1.2 光标和图标文件	(25)
5.1.3 设置分辨率	(25)
5.2 光标和图标编辑器	(26)
5.3 位图编辑器	(26)
5.4 使用图形编辑器	(26)
5.5 使用颜色	(27)
5.5.1 调色板	(28)
5.5.2 为光标和图标图像选择颜色	(28)
5.5.3 调整调色板	(29)
5.6 使用图形工具绘制和编辑图形	(30)

5.7 菜单	(34)
5.7.1 Image 菜单	(34)
5.7.2 Options 菜单	(34)
5.7.3 Tools 菜单	(34)
第六章 对话框编辑器	(35)
6.1 文件和头文件	(35)
6.2 编辑器概述	(35)
6.3 对话框及其控制项	(35)
6.3.1 带菜单的对话框	(36)
6.3.2 创建对话框或控制项	(36)
6.3.3 选择对话框或者控制项	(36)
6.3.4 移动对话框或控制项	(37)
6.3.5 修改对话框或控制项的大小	(37)
6.3.6 限制鼠标移动	(38)
6.3.7 设置对话框或控制项的属性	(38)
6.3.8 修改控制项的制表次序	(40)
6.3.9 定义逻辑控制项组	(40)
6.4 Tools 控制板	(40)
6.4.1 指针工具	(40)
6.4.2 对话框工具	(41)
6.4.3 控制工具	(41)
6.5 对齐控制板及工具	(46)
6.6 菜单	(47)
6.6.1 Dialog 菜单	(47)
6.6.2 Controls 菜单	(48)
6.6.3 Tools 菜单	(49)
6.6.4 Aligment 菜单	(49)
第七章 菜单编辑器	(50)
7.1 文件	(50)
7.2 使用编辑器	(51)
7.3 定义菜单文本	(51)
7.3.1 游历编辑器	(51)
7.3.2 编辑文本	(51)
7.3.3 定义菜单层次	(52)
7.3.4 修改菜单项的位置	(53)
7.3.5 设置激活键	(54)
7.3.6 在菜单文本中插入制表符	(54)
7.4 定义菜单字体和属性	(55)
7.4.1 在弹出式菜单中添加分隔行	(55)

7.4.2 使用选择标志	(55)
7.4.3 定义菜单项的字体	(55)
7.4.4 对齐栏目中的菜单项	(56)
7.4.5 设置求助属性	(57)
7.5 菜单测试	(57)
第八章 串编辑器	(58)
8.1 定义串	(58)
8.1.1 域和行指示器	(58)
8.1.2 串表	(59)
8.1.3 游历编辑器	(60)
8.1.4 编辑文本	(60)
8.2 头文件	(60)
第九章 头文件编辑器	(61)
9.1 头文件	(61)
9.2 启动头文件编辑器	(61)
9.2.1 创建新的头文件	(62)
9.2.2 打开已存在的头文件	(62)
9.3 使用头文件编辑器	(62)
9.3.1 ID 号	(62)
9.3.2 编辑符号	(63)
9.3.3 删除符号	(64)
9.3.4 移动符号	(64)
9.3.5 保存头文件	(64)
第十章 公用键及菜单	(66)
10.1 表编辑	(66)
10.2 表游历	(66)
10.2.1 在编辑域中键入字符	(67)
10.2.2 在选择域中选择选项	(67)
10.3 使用菜单	(67)
10.3.1 File 菜单	(68)
10.3.2 Edit 菜单	(71)
10.3.3 Header 菜单	(73)
第十一章 资源开发工具的故障诊断及错误信息	(74)
11.1 错误信息	(74)
11.2 内存配置	(74)
11.2.1 控制内存分配	(75)
11.2.2 使用 WIN.INI 控制内存分配	(76)
11.2.3 磁盘空间	(76)
11.3 控制板显示	(77)

11.4 切换编辑器	(77)
11.5 恢复磁盘空间	(77)

第二部分 源程序的建立和编辑工具

第十二章 交互式编辑器	(81)
12.1 新增功能和原有功能	(81)
12.2 编辑器命令	(81)
12.2.1 光标移动命令	(82)
12.2.2 插入和删除命令	(82)
12.2.3 块命令	(82)
12.2.4 其他编辑命令	(83)
12.3 编辑器命令的进一步说明	(84)
12.3.1 光标跳移	(84)
12.3.2 块命令	(84)
12.3.3 其他编辑命令	(85)
12.3.4 查找并替换	(86)
12.3.5 配对	(86)

第三部分 实用工具程序

第十三章 创建模块定义文件的实用程序: IMPDEF	(94)
第十四章 生成移入库的工具: IMPLIB	(96)
第十五章 独立使用的程序管理器: MAKE	(98)
15.1 MAKE 的工作原理	(98)
15.2 启动 MAKE	(99)
15.2.1 BUILTINS.MAK 文件	(99)
15.2.2 命令行选项	(100)
15.3 MAKE 的一种简单运用	(100)
15.4 制作 makefile 文件	(102)
15.5 makefile 文件的组成	(102)
15.6 命令表	(103)
15.6.1 显式规则	(106)
15.6.2 隐式规则	(108)
15.6.3 宏	(110)
15.7 指令	(113)
15.7.1 点指令	(114)
15.7.2 文件嵌入指令	(114)
15.7.3 条件执行指令	(115)

15.7.4 报错指令	(117)
15.7.5 取消宏定义指令	(117)
15.8 MAKE 错误信息	(117)
第十六章 Turbo 库管理程序: TLIB	(122)
16.1 为什么要使用目标模块库	(122)
16.2 TLIB 命令行及操作表	(123)
16.3 使用应答文件	(124)
16.4 建立扩展字典: /E 选项	(125)
16.5 设置页大小: /P 选项	(125)
16.6 高级操作: /C 选项	(125)
16.7 例子	(126)
第十七章 Turbo 连接程序: TLINK	(127)
17.1 调用 TLINK	(128)
17.1.1 DOS 下连接的一个例子	(128)
17.1.2 Windows 程序连接的一个例子	(128)
17.1.3 Tlink 命令行中的文件名	(129)
17.1.4 使用应答文件	(129)
17.1.5 TLINK 的配置文件	(130)
17.1.6 使用 TLINK 连接 Borland C++ 模块	(130)
17.1.7 在 BCC 中使用 TLINK	(133)
17.2 TLINK 选项	(133)
17.2.1 TLINK 配置文件	(134)
17.2.2 /3 (80386 32 位码)	(134)
17.2.3 /A (段对齐)	(134)
17.2.4 /c (大小写敏感)	(134)
17.2.5 /C (大小写敏感移出)	(134)
17.2.6 /d (重复出现符号)	(135)
17.2.7 /e (不使用扩展字典)	(135)
17.2.8 /i (未初始化尾段)	(135)
17.2.9 /l (行号)	(135)
17.2.10 /L (库查找路径)	(136)
17.2.11 /m、/s 和 /x (映像选项)	(136)
17.2.12 /n (忽略默认库)	(137)
17.2.13 /o (覆盖)	(137)
17.2.14 /P (组合代码段)	(138)
17.2.15 /t (极小模式.COM 文件)	(138)
17.2.16 /Td 和 /Tw (目标选项)	(139)
17.2.17 /v (调试信息)	(139)
17.2.18 /y (扩充或扩展内存)	(139)

17.2.19 限制	(140)
17.3 模块定义文件	(140)
17.3.1 模块定义文件的缺省值	(140)
17.3.2 一个例子	(141)
17.4 模块定义参考	(142)
17.4.1 CODE	(142)
17.4.2 DATA	(142)
17.4.3 DESCRIPTION	(143)
17.4.4 EXETYPE	(143)
17.4.5 EXPORTS	(143)
17.4.6 HEAPSIZE	(144)
17.4.7 IMPORTS	(144)
17.4.8 LIBRARY	(144)
17.4.9 NAME	(145)
17.4.10 SEGMENTS	(145)
17.4.11 STACKSIZE	(145)
17.4.12 STUB	(146)
17.5 TLINK 出错信息	(146)
第十八章 图形驱动程序转换工具: BGIOBJ	(156)
18.1 向 GRAPHICS.LIB 添加新的.OBJ 文件	(156)
18.2 注册驱动程序和字模	(156)
18.3 例子	(157)
18.4 /F 选项	(158)
18.5 高级特性	(158)
第十九章 C 语言程序预处理器: CPP	(162)
19.1 将 CPP 用作宏预处理器	(162)
19.2 例子	(162)
第二十章 文本查找工具: GREP	(164)
20.1 GREP 的选项及优先级	(164)
20.2 查找串及正则表达式中的操作符	(166)
20.3 文件名	(167)
20.4 例子	(167)
第二十一章 目标模块交叉引用工具: OBJXREF	(171)
21.1 OBJXREF 命令行及选项	(171)
21.2 应答文件	(173)
21.2.1 自由形式的应答文件	(173)
21.2.2 工程文件	(173)
21.2.3 连接器应答文件	(174)
21.2.4 /D 命令	(174)

21.2.5 /O 命令	(174)
21.2.6 /N 命令	(174)
21.3 OBJXREF 报告样本	(175)
21.3.1 按公用名报告 (/RP)	(176)
21.3.2 按模块报告 (/RM)	(176)
21.3.3 按引用报告(/RR) (缺省方式)	(176)
21.3.4 按外部引用报告 (/RX)	(177)
21.3.5 按模块长度报告 (/RS)	(177)
21.3.6 按类报告 (/RC)	(177)
21.3.7 按未引用符号名报告 (/RU)	(178)
21.3.8 冗长报告 (/RV)	(178)
21.4 使用 OBJXREF 的例子	(178)
21.5 OBJXREF 错误信息和警告	(179)
21.5.1 错误信息	(179)
21.5.2 警告信息	(179)
第二十二章 联机帮助工具: THELP	(181)
22.1 安装 THELP	(181)
22.2 装入和激活 THELP	(181)
22.3 THELP 的命令行	(181)
22.4 THELP 命令行选项摘要	(181)
22.4.1 /C#xx 选项	(182)
22.4.2 /F 名选项	(183)
22.4.3 /H, /? 和? 选项	(183)
22.4.4 /Kxyy 选项	(184)
22.4.5 /S+选项	(184)
22.4.6 /S-选项	(184)
22.4.7 /U 选项	(185)
22.4.8 /W 选项	(185)
第二十三章 编辑器宏编译器: TEMC	(186)
23.1 TEMC 命令行	(186)
23.2 语法	(186)
23.3 手稿文件例子	(188)
23.3.1 MakeFuncText	(190)
23.3.2 键码	(191)
23.3.3 命名键	(191)
23.4 预定义的编辑器命令	(192)
23.5 错误信息	(196)
第二十四章 其他实用工具程序	(199)
24.1 工程文件转换工具: PRJ2MAK	(199)

24.2	配置文件与工程文件互换实用程序: PRJCFG	(200)
24.3	Turbo C 与 Borland C++的工程文件转换工具: PRJCNVT	(200)
24.4	源模块交叉引用实用程序: TCREF	(200)
24.4.1	应答文件	(201)
24.4.2	与 TLINK 的兼容性	(201)
24.4.3	报告	(201)
24.5	文件日期和时间改变工具: TOUCH	(202)
24.6	工程间拷贝传送项的实用程序: TRANCOPY	(202)
24.7	字符转换工具: TRIGRAPH	(203)

第四部分 附 录

附录 A	C++ 1.2 版流	(207)
A.1	概要介绍	(207)
A.2	输出	(207)
A.2.1	内部类型	(207)
A.2.2	Put 函数	(208)
A.2.3	内部格式化函数	(208)
A.2.4	用户定义类型	(210)
A.3	输入	(211)
A.3.1	内部类型	(211)
A.3.2	空白控制	(211)
A.3.3	用户定义类型	(213)
A.3.4	初始化流	(213)
A.3.5	输出流	(213)
A.3.6	输入流	(214)
A.4	文件 I/O	(215)
A.5	流缓冲	(217)
A.6	流 I/O	(218)
A.7	流状态	(219)
附录 B	预编译头文件	(220)
B.1	预编译头文件的工作原理	(220)
B.2	使用预编译头文件	(220)
B.2.1	指定文件名	(221)
B.2.2	验证一致性	(221)
B.2.3	优化预编译头文件	(221)
附录 C	容器类库	(223)
C.1	概述及重要概念	(223)
C.1.1	基于对象的层次中的抽象类	(224)
C.1.2	容器类	(224)

C.1.3 Collection 层次	(225)
C.1.4 ContainerIterator 层次	(225)
C.2 类库中所使用的约定	(226)
C.2.1 文件命名	(226)
C.2.2 源命名	(226)
C.2.3 注释风格	(227)
C.3 类库目录	(229)
C.3.1 INCLUDE 目录	(229)
C.3.2 SOURCE 目录	(229)
C.3.3 LIB 目录	(229)
C.3.4 EXAMPLES 目录	(229)
C.4 类参考	(230)
C.4.1 AbstractArray	(230)
C.4.2 Array	(231)
C.4.3 ArrayIterator	(233)
C.4.4 Association	(233)
C.4.5 Bag	(235)
C.4.6 BaseDate	(235)
C.4.7 BaseTime	(236)
C.4.8 Collection	(237)
C.4.9 Container	(238)
C.4.10 ContainerIterator	(239)
C.4.11 Date	(240)
C.4.12 Dictionary	(240)
C.4.13 Deque	(241)
C.4.14 DoubleList	(243)
C.4.15 DoubleListElement	(244)
C.4.16 DoubleListIterator	(244)
C.4.17 Error	(245)
C.4.18 HashTable	(246)
C.4.19 HashTableIterator	(246)
C.4.20 List	(247)
C.4.21 ListElement	(248)
C.4.22 ListIterator	(248)
C.4.23 Object	(248)
C.4.24 Queue	(250)
C.4.25 Set	(252)
C.4.26 Sortable	(252)
C.4.27 SortedArray	(253)
C.4.28 Stack	(253)
C.4.29 String	(255)