

Borland C++ 3.1 4.0



Borland C++ 3.1 和 4.0 使用手册

[美] Chris H.Pappas & William H.Murray 著

黄昱 李英 王慧敏 译

周志宇 审校

Borland C++
3.1 和 4.0
使用手册

清华大学出版社



Borland C++ 3.1 和 4.0 使用手册

C. H. 帕帕斯

[美] Chris H. Pappas & William H. Murray

黄 昱 李 英 王慧敏 译

周志宇 审校

清华大学出版社

(京)新登字 158 号

Borland C++ 3.1 和 4.0 使用手册
Borland C++ Handbook, Fourth Edition
Chris H. Pappas & William H. Murray

English edition copyright © 1994 by McGraw-Hill, Inc. All rights reserved.

中文版版权© (1995 年)属于清华大学出版社,保留所有权利。

本书获得 McGraw-Hill 正式授权。在中国大陆内翻译发行,但不得另行授权予他人或在其它地区发行。

未经出版者书面允许,不得用任何手段复制或抄袭本书的内容。

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++ 3.1 和 4.0 使用手册/(美)帕帕斯(Pappas, C. H.), (美)默里(Murray, W. H.)著;黄昱等译. —北京:清华大学出版社,1994

书名原文:Borland C++ Handbook

ISBN 7-302-01725-5

I . B… II . ①帕… ②默… ③黄… III . C 语言-程序设计-手册 IV . TP312C

中国版本图书馆 CIP 数据核字(94)第 15696 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:清华大学印刷厂

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:42.25 字数:998 千字

版 次:1995 年 7 月第 1 版 1995 年 7 月第 1 次印刷

书 号:ISBN 7-302-01725-5/TP·755

印 数:0001—5000

定 价:74.00 元

序 言

本书着重介绍 Borland 公司的最新版的 C/C++ 编译程序软件包。其中一个软件包是为在 DOS 和 Windows 环境下工作的程序开发者提供的;另一个软件包是为开发 Windows NT 应用程序的程序员提供的。这两个软件包中都含有一个 C/C++ 编译程序、汇编程序、调试程序以及各种其它开发工具。由于 Borland 公司的这些编译程序软件包已经非常完整,所以读者不必是使用这些软件包或本书的专业的程序员。读者只要真正对学习 C/C++ 语言感兴趣,对汇编语言和 Windows 编程感兴趣,便能掌握本书的内容。本书将从最基础开始向读者讲授 C/C++ 语言、汇编语言以及 Windows 编程;并且本书是为初学者和专业程序员设计的。如果读者还感兴趣学习更多的关于面向过程或面向对象的编程的知识——那么本书是再合适不过的了。

本书主要是为读者(程序员)而设计的一本多方面的手册。本书将指导读者通过程序开发者可能遇到的“陷阱”和潜在的“陷阱”。读完本书之后,读者将能够有效地使用自己的编译程序软件包,编写和调试复杂的 C/C++、汇编语言和 Windows 代码。与快速开始的书籍不同,本书为读者继续学习打下了坚实的基础。

本书的组织方法

根据读者的背景和编程需要,本书有各种使用方法。本书可被分为几个主要部分:

第 1,2,3,4 章向读者介绍 Borland C/C++ 编译程序软件包中的编程工具。

第 5,6,7,8,9,10,11,12 章涉及 C 和 C++ 语言中的基础和编程概念。这些章节是面向过程的章节,教授传统的编程概念。

第 13 章和第 14 章在前几章介绍的 C 和 C++ 的基础上,向读者教授面向对象编程的概念和定义。在这几章中,读者将学会区别面向过程编程和面向对象编程的方法,学会编写简单的面向对象的程序的方法。第 14 章中开发的面向对象的程序类似于前面几章中的使用面向过程的方法编写的程序。面向对象的概念将被补充到后面的处理 Windows 3.1 和 Windows NT 程序的章节中。

第 15 章和第 16 章向读者示出了利用标准的 C 和 C++ 库的许多特性的方法。读者将从这些章中学习到允许程序员控制计算机硬件和使用 DOS 下的扩展图形例程的功能。

第 17,18,19 章教授汇编语言编程基础。读者将学到使用高速汇编语言代码控制计算机的键盘和屏幕选项的方法。读者还将学到组合 C 代码和汇编语言例程、传递参数以及与外部硬件电路的接口方法。

第 20 章将向读者介绍适用于 DOS 下的 Windows 3.1 和 Windows NT 的 Microsoft Windows 概念。从第 21,22,23 章中读者将学到使用 C++ 编译程序开发包括 GDI 原型、光标、图标、菜单和对话框的应用程序的方法。这几章是连续的并使读者能很快地开始开发 16 位 Windows 3.1 应用程序,这些应用程序可在 DOS 下的 Windows 3.1 和 Windows NT 环境下运行。

第 24 章和第 25 章向读者示出了把面向对象的特征添加到 16 位的 Windows 应用程序中。读者将学到使用 Borland 公司的 ObjectWindows 库向程序中添加高级功能的方法。这些功能有助于使程序维护轻松地执行。

第 26 章和第 27 章将教授读者在 Windows NT 编程环境下开发真正 32 位应用程序的方法。第 20 章至第 23 章中提到的许多概念在这两章中被补充以包括 Windows NT 的特殊的功能。读者必须将已装入的 Windows NT 作为自己的操作系统并在系统中装入具有 Windows NT 编译能力的 Borland C/C++ 应用程序,才能开发这几章中的应用程序。

本书中所用的约定

本书中一直使用下列约定:

- 若引用的 C 和 C++ 项相同,则所引用的是 C。
- 函数名和其它保留字以黑体字出现。
- 关键字、变量和常量以斜体字出现。
- 程序清单以印刷体风格文字示出。
- 用户从键盘输入的文字以黑体字印出。
- 第一次提到的技术术语以斜体字印出。

目 录

序言	VI
----------	----

第一部分 绪 论

第 1 章 整体是部分的和	3
1.1 BORLAND C++ 软件包的配置	3
1.2 在用户系统中安装 Borland C++ 软件包	6
1.3 本书的组织	6
第 2 章 使用 Borland C++ 编译程序	8
2.1 主窗口	8
2.2 帮助(Help)	9
2.3 用户的第一个程序	10
2.4 管理窗口	18
2.5 管理多个源文件	19
2.6 其它菜单选项	22
第 3 章 使用 Borland 汇编程序	26
3.1 设置汇编程序	26
3.2 汇编过程:第一个例子	27
3.3 汇编程序的选项和开关	30
3.4 连接程序的选项和开关	32
3.5 重要的程序和文件用法	33
3.6 汇编过程:第二个例子	38
3.7 汇编程序的方式:MASM 和 IDEAL	43
3.8 混合方式:第三个和第四个例子	44
3.9 查找汇编语言错误:第五个例子	48
3.10 还有更多的内容	50
第 4 章 使用 Borland 调试程序和剖析程序	51
4.1 调试程序——探查	51
4.2 剖析程序——一种高效的检验工具	51
4.3 使用调试程序	51
4.4 使用剖析程序	63
4.5 准备开发程序	68

第二部分 C 和 C++ 的基础

第 5 章 C 和 C++ 的基础	71
5.1 C 的历史	71
5.2 ANSI C 标准	76
5.3 C++ 的发展和面向对象的编程	77
5.4 C++ 的历史	78

5.5 C程序的基本元素	82
5.6 文件	90
第6章 数据	94
6.1 什么是标识符	94
6.2 关键字	95
6.3 标准C和C++数据类型	97
6.4 访问修饰符	103
6.5 pascal、cdecl、near、far 和 huge 修饰符	105
6.6 数据类型转换	108
6.7 存储类	114
6.8 操作符	114
6.9 对操作符优先级的理解	121
6.10 标准C和C++库	122
第7章 控制	125
7.1 条件语句	125
7.2 循环语句	139
第8章 函数	158
8.1 函数的风格和原型	158
8.2 函数的参数	164
8.3 函数类型	172
8.4 main 函数的参数	178
8.5 C++ 的特殊功能	182
8.6 与作用域有关的编程问题	185
第9章 数组	191
9.1 什么是数组	191
9.2 数组的基本特征	191
9.3 定义数组	192
9.4 初始化数组	192
9.5 使用数组索引	195
9.6 对数组使用 sizeof	197
9.7 数组越界检查	199
9.8 数组和字符串	199
9.9 多维数组	201
9.10 数组与函数	204
9.11 使用数组的字符串函数	211
第10章 指针	216
10.1 什么是指针变量	216
10.2 函数指针	233
10.3 动态内存分配	236
10.4 指针和数组	241
10.5 C++ 引用类型	251
第11章 C和C++中的输入和输出	254

11.1	C 语言中的输入和输出	254
11.2	C++ 中的输入和输出	274
11.3	C++ 的高级输入和输出	281
第 12 章	结构、联合和杂项	294
12.1	结构	294
12.2	联合	311
12.3	杂项	314
12.4	链表	316
第 13 章	类	321
13.1	基本的类概念	321
13.2	操作符重载	341
13.3	派生类	344
第 14 章	面向对象程序设计导论	350
14.1	C++ 和面向对象程序设计	350
14.2	面向对象程序设计定义和术语	350
14.3	开发一个面向对象的链表程序	351

第三部分 高性能程序设计

第 15 章	高性能程序设计:初涉重要的 C 和 C++ 库	373
15.1	C 和 C++ 头文件	373
15.2	标准库函数	374
15.3	字符函数	381
15.4	内存和串函数	387
15.5	数学函数	394
15.6	时间函数	400
第 16 章	DOS 系统资源和图形	406
16.1	BIOS 头文件	406
16.2	DOS 头文件	409
16.3	图形头文件	415
第 17 章	汇编语言	432
17.1	运算程序	432
17.2	使用查找表	446
17.3	在 16 位 DOS 操作系统上使用 BIOS/DOS 系统中断和地址端口	449
第 18 章	高性能程序设计:宏和过程	460
18.1	宏	460
18.2	过程	470
18.3	目标模块库	478
18.4	宏、过程和库的对比	480
第 19 章	连接 C 和汇编语言代码	482
19.1	使用内联的汇编语言	482
19.2	编写分离的 C 和汇编语言模块	486

19.3	一个简单的 C 和汇编语言连接	489
19.4	使用 C++ 和汇编语言的硬件接口	492
19.5	从 C 向汇编语言传递数组	496

第四部分 Windows 和 Windows NT 程序设计

第 20 章	Windows 和 Windows NT 概念介绍	501
20.1	什么是 Windows	501
20.2	Windows 性能	502
20.3	Windows 特点	504
20.4	Windows 概念与术语	505
20.5	创建一个 Windows 程序的步骤	515
第 21 章	编写 16 位 Borland C++ Windows 3.1 应用程序	516
21.1	为什么使用简化 Windows 平台模板	516
21.2	编译和链接过程	517
21.3	简化 Windows 平台	517
21.4	在每一个 16 位 Windows 3.1 SWP 中使用的重要特性	532
21.5	由此起步	540
第 22 章	使用 Borland Resource Workshop 和资源编译程序	541
22.1	Windows 资源	541
22.2	使用 Borland Resource Workshop (BRW)	542
22.3	从命令行使用资源编译器(RC)	555
22.4	其它资源信息	558
第 23 章	用 Windows 3.1 开发一个 16 位的具有展示品质的棒图	559
23.1	调色板管理程序	559
23.2	在一个应用程序中使用字体	561
23.3	棒图	563
23.4	使用 Windows Debugger	581
第 24 章	ObjectWindows C++ 程序开发的一个模板	582
24.1	ObjectWindows 的三个面向对象的特征	582
24.2	一个 ObjectWindows 对象	584
24.3	一个成功的模板 <i>swpo.cpp</i>	585
24.4	试验 <i>swpo.cpp</i> 模板	591
24.5	更高级的工作	596
第 25 章	开发具有资源的 C++ ObjectWindows 应用程序	597
25.1	<i>draw25</i> , 开发一个定制图符、光标、菜单和一组键盘加速键	597
25.2	<i>pie25</i> , 具有一个定制图符、光标、菜单和两个对话框的展示品质的图形	608
第 26 章	开发 32 位 Windows NT 应用程序	621
26.1	Windows NT 的不同点	621
26.2	一个简单的应用程序 <i>ntswp</i>	623
26.3	展望后续内容	633
第 27 章	一个带有资源的 32 位 NT 应用程序	634

27.1 带有资源的应用程序	634
----------------------	-----

第五部分 附 录

附录 A 扩展 ASCII 表	651
附录 B DOS 10H、21H 和 33H 中断参数	654

第一部分

绪 论

整体是部分的和

读者在一个硬件商店中准备购买一种商品时,是否曾面临必须在两种工具中做出选择?一种工具标有“通用”,另一种工具标有“专业”。根据已有的经验,通用工具使用的机会很多,但从长远的观点看,通用工具不能被扩充使用。另一方面,专业化的工具通常设计得比较好,含有附加的功能,且经得起频繁使用。读者将购买哪种工具呢?通常要根据价格、性能、质量和使用目的来做决定。

当购买 Borland C++ 软件包时,读者很可能正在寻找一种编程工具,或一组工具,这些编程工具应具有经得起频繁使用的灵活性和性能。这一软件包不会因为读者编程技巧的增加而过时。选择 Borland C++ 软件包便是最佳的选择。使用 Borland C++ 软件包并不要求读者是一个专业的程序员。

本书将试图做三项基本工作。首先,本书将帮助读者研究和理解 Borland C++ 软件包中个别组成的定义和用法。其次,本书是一本完整的 C、C++ 和汇编语言的教科书。它将教给读者编写 C、C++ 和汇编语言程序的规则和技巧。读完本书后,读者所学的技巧将使读者从初级变为这些语言的中级水平的程序员。第三,本书将向读者示出集成该编译软件包的组成的方法以便使读者以最佳方法混合编程 C、C++ 和汇编语言。

本章将向读者介绍 Borland C++ 软件包及其所有组成部分,且读者将学会如何在计算机上装入和运行该软件包。这样,不管读者的程序是在具有 Borland C++ 3.1 编译程序的 DOS 环境下,还是在具有 Borland C++ 4.0 编译程序的 DOS 或 Windows NT 环境下,一个充满编程乐趣的新世界都将在等着读者。

1.1 BORLAND C++ 软件包的配置

当用户从包装盒中取出 Borland C++ 软件包时,可能会面对一大堆东西。但是,不要被这么多东西吓倒。先将说明书、磁盘和文献分开摆放。把磁盘和文献放在一边,先检查一下说明书。用户可将这一堆说明书分成两类:用户指南和参考指南。用户指南中含有软件包的每个组成部分的安装指导(本章后面将讲述安装)。其中还含有关于每个组成部分的特征的详细信息。另一方面,参考指南中含有每种语言的参考信息。其中包括函数调用及其它助记符的详细信息。切勿丢失这些说明书。

这样,便可将说明书分成三类:C++、汇编语言和调试程序/剖析程序。下面几节将解释这三种产品是什么,能干什么用,以及如何使用它们。

1.1.1 C 和 C++ 编译程序

用户可使用 C++ 软件包为 Windows 3.1 或 Windows NT 编写标准的且是独立的 C 或 C++ 程序或者应用程序。C 语言可提供结构化、模块化且是可编译的通用的编程环

境。C 正在很快成为系统程序员和初学者所需的语言。用户装入 C++ 软件之后, 便可选择是在集成开发环境下工作还是命令行方式下使用编译程序。

C++ 编译程序将程序代码转换为目标代码。当用户编写 C 或 C++ 程序时, 源代码名应以 .c 或 .cpp 扩展名结束。下面是一些源代码名的例子:

```
myfirst.c
mysecond.cpp
counter.c
multiplier.cpp
```

编译成功后, 磁盘上将含有目标文件, 例如:

```
myfirst.c
mysecond.cpp
counter.c
multiplier.cpp
myfirst.obj
mysecond.obj
counter.obj
multiplier.obj
```

目标文件中含有原始源代码翻译过来的计算机可直接读取的代码。要获得最终的可执行文件, 应使用连接程序连接目标代码。连接程序设置程序执行时可执行代码所处的内存地址。连接程序通常产生带有 .exe 扩展名的文件。目标文件被连接后, 磁盘中将含有下列文件:

```
myfirst.c
mysecond.cpp
counter.c
multiplier.cpp
myfirst.obj
mysecond.obj
counter.obj
multiplier.obj
myfirst.exe
mysecond.exe
counter.exe
multiplier.exe
```

用户在命令行提示符下键入可执行文件的名称便可执行该文件。读者从第 2 章“使用 Borland C++ 编译程序”中可学到输入和执行简单的 C 和 C++ 代码的方法。

1.1.2 汇编程序

用户可使用汇编程序软件包用汇编语言编写独立的程序。汇编语言是机器特定的

(微处理器),因此,它不如 C 代码的移植性好。Borland 汇编程序允许用户为 Intel 系列的处理器芯片开发代码。这些处理器芯片当前包括 8086, 8088, 80186, 80286, 80386 和 80486。

汇编语言是一种意义深远的编程语言,它允许用户编写与计算机的实际机器码联系最紧密的代码。尽管许多程序员认为汇编语言很难学且很难编程,但是汇编语言可向熟练的程序员提供速度上的优势和完全的硬件控制。用户可使用 Borland 编辑器编写独立的汇编语言程序,然后汇编和连接编写好的程序。

汇编程序将程序代码转换为目标代码。当用户编写汇编语言程序时,源代码的名称通常以 *.asm* 扩展名结束。下面是一些汇编语言源代码的名称:

```
srcrclear.asm  
timer.asm
```

汇编成功后,磁盘中将含有目标文件,例如:

```
srcrclear.asm  
timer.asm  
srcrclear.obj  
timer.obj
```

由汇编语言汇编得来的目标文件中也含有原始源代码翻译过来的计算机可直接读取的代码。要获得最终的可执行文件,应使用连接程序连接目标代码。汇编语言程序汇编后形成的目标文件被连接后,磁盘中将含有下列文件:

```
srcrclear.asm  
timer.asm  
srcrclear.obj  
timer.obj  
srcrclear.exe  
timer.exe
```

用户在命令行提示符下键入可执行文件的名称便可执行该文件。读者从第 3 章“使用 Borland 汇编程序”中可学到输入和执行简单的汇编语言代码的方法。

1.1.3 调试程序和剖析程序

当用户编写 C, C++ 或汇编语言程序时,总会出现错误。用户可能的错误可被分为两大类:句法错误和编程错误。当用户编译或汇编程序代码时,句法错误被报告给用户。若错误足够严重,则编译程序或汇编程序将退出编译或汇编过程且不生成任何可执行文件。要纠正这类错误,用户应进入编辑程序并找到代码中标明出错的行。Borland 参考指南没对这类错误作说明。句法错误通常是最容易确定的错误(编辑和纠正代码将在第 2 章中详细讨论)。

用户程序可能被正确地编译或汇编——不报告任何错误——但程序却不按用户的意图工作。若出现这种情况,则错误通常是编程错误——即用户的编程逻辑有错。编程错

误比句法错误难跟踪。Borland 公司的调试程序是为用户尽可能快地定位编程错误而设计的。要使用调试程序,用户必须拥有一个可执行程序,即使该可执行程序不能正确执行。读者将在第 4 章“使用 Borland 调试程序和剖析程序”中学到编写集成到调试程序环境中的程序的方法。

最后说一下,剖析程序是一种工具,它将帮助用户编写更有效的代码。通过使用剖析程序,用户将知道自己的程序在执行期间在何处最耗费时间。这便可帮助用户编写效率更高的 C 和 C++ 代码并可提供一个汇编语言例程来加速这种操作。读者将在第 4 章中学到使用剖析程序的方法。

1.2 在用户系统中安装 Borland C++ 软件包

用户的 Borland C++ 编译软件包的成功与否和使用的容易程度依赖于所装入的计算机系统和用户所选择安装选项。下面几节提供优化用户编程环境的方法。

1.2.1 用户计算机

Borland C++ 软件包可在 IBM 和 IBM 兼容计算机的一个很宽的范围内运行。Borland C++ 的使用说明书中将讨论最低的系统配置,但本书建议使用下列系统剖面:

80486 计算机

DOS 6.0 或 Windows NT 操作系统

对 DOS,需 2M RAM 内存;对 Windows NT 需 16M RAM 内存

彩色监视器(图形)

300MB 硬盘

鼠标

若用户系统由前面建议的剖面构成,则用户便可快速并有效地交换使用 Borland C++ 软件包的各种组成。

1.2.2 设置用户系统

将第一张盘插入磁盘驱动器并键入 install 便可开始安装。在安装的各步骤中,若用户不知道所需的另一个选项,则应接受建议的缺省值。若用户需要,则可修改这些缺省值,其中包括路径名。当用户安装 C++ 编译程序时,应确定需在用户环境中包括哪种内存模式。本书中的程序中仅使用小(“Small”)内存模式,若用户仅安装小内存模式,便可保留大量磁盘空间。若用户需要的话,可在以后添加其它内存模式。

一旦用户安装了每一个软件,都将返回到含有 C++ 编译程序的子目录中。

1.3 本书的组织

序言中已对本书的安排作了完整的描述。若读者的编程目的具有某种选择性,则下

面是本书建议读者读的章节：

快速使用每种软件：第 2 至第 4 章

只讲述汇编语言的章节：第 3、4、17 至 19 章

只讲述 C 和 C++ 的章节：第 2 章、第 4 至第 14 章

Microsoft Windows 3.1 开发：第 20 至第 25 章

Microsoft Windows NT 开发：第 26 至第 27 章

若读者希望尽可能快地开发 C 和 C++ 程序，则应去阅读第 2 章。若读者对汇编语言感兴趣，则应去阅读第 3 章。读者一旦读完了这些章节，便可立刻去阅读书中自己感兴趣的章节，继续提高自己的编程技巧。