

计算机 C/C++ 语言系列丛书

3.1~4.0

# Borland C++

施小龙  
葛玉宝 编写  
邓明辉

## 深入编程

学苑出版社

计算机 C/C++ 语言系列丛书

Borland C++ 3.1-4.0

# 深入编程

施小龙  
葛玉宝 编写  
邓明辉  
万博 审校

学苑出版社  
1994年·北京

(京)新登字 151 号

### 内 容 简 介

本书主要是针对利用 Borland C++ 3.1 进行 C++ 程序设计的人编写的,全面系统地介绍了 Borland C++ 面向对象编程技术。首先,介绍了 C++ 语言的语法,如何定义、存取类,如何重载函数和操作符,如何构造和使用构造函数和析构函数,如何使用 C++ 流和如何使用 C++ 派生类等。然后介绍如何用 Turbo Vision 编写桌面、菜单条、菜单热键、状态行,如何用 Turbo Vision 编写窗口、窗口滚动条,如何用 Turbo Vision 编写对话框处理其元素输入框、静态文本、按钮、列表框、历史列表等,如何用 Turbo Vision 消息的机制对事件进行控制,以及如何用 Turbo Vision 简单地实现功能复杂的应用程序。

欲购本书的用户可直接与北京 8721 信箱联系,电话 2562329, 邮编 100080。

计算机 C/C++ 语言系列丛书

## Borland C++ 3.1-4.0 深入编程

---

编 写:施小龙 葛玉宝 邓明辉  
审 校:万 博  
责任编辑:甄国宪  
排 版:万博图书创作社  
出版发行:学苑出版社 邮政编码:100036  
社 址:北京市海淀区万寿路西街 11 号  
印 刷:北京四季青印刷厂印刷  
开 本:787×1092 1/16  
印 张:52.38 字数:1274 千字  
印 数:1~5000 册  
版 次:1994 年 6 月第 1 版第 1 次  
本册定价:79.00 元  
ISBN7-5077-0875-6/TP·24

---

# 目 录

## 第一部分 C++语言基础

### 第一章 C++语言特性

1.1 引 用 .....	2
1.1.1  引用参数 .....	2
1.2 作用域存取操作符 .....	3
1.3 new 与 delete 操作符 .....	4
1.3.1 关于数组的 new 操作符 .....	4
1.3.2  ::operator new .....	5
1.3.3 带有 new 操作符的初始值 .....	5
1.4 类 .....	5
1.4.1 类 名 .....	5
1.4.2 类类型 .....	5
1.4.3 类名作用域 .....	6
1.4.4 类对象 .....	6
1.4.5 类成员表 .....	6
1.4.6 成员函数 .....	6
1.4.7 关键字 this .....	7
1.4.8 内部函数 .....	7
1.4.9 静态成员 .....	7
1.4.10 成员作用域 .....	9
1.4.11 基类与派生类存取 .....	10
1.5 虚基类 .....	12
1.6 类的友元 .....	12
1.7 构造函数(constructors)与析构函数(destructors)简述 .....	14
1.8 构造函数 .....	15
1.8.1 缺省构造函数 .....	15
1.8.2 拷贝构造函数 .....	16
1.8.3 构造函数的重载 .....	16
1.8.4 构造函数的调用次序 .....	16
1.8.5 类的初始化 .....	17
1.9 析构函数 .....	20
1.9.1 析构函数的调用 .....	20
1.9.2 atexit、#pragma exit 与析构函数 .....	20

1.9.3	exit 与析构函数 .....	20
1.9.4	abort 与析构函数 .....	20
1.9.5	虚析构函数 .....	21
1.10	重载操作符 .....	22
1.11	操作符函数 .....	23
1.11.1	重载操作符和继承 .....	23
1.11.2	重载 new 和 delete .....	23
1.11.3	重载单目操作符 .....	24
1.11.4	重载赋值操作符 = .....	24
1.11.5	重载函数调用操作符() .....	25
1.11.6	重载下标操作符[] .....	25
1.11.7	重载类成员存取操作符-> .....	25
1.12	虚函数 .....	25
1.13	抽象类 .....	27
1.14	C++作用域 .....	28
1.14.1	类作用域 .....	28
1.14.2	隐 藏 .....	28
1.14.3	C++作用域规则小结 .....	28

## 第二章 使用C++的类

2.1	C与C++派生类型的比较 .....	30
2.1.1	在C++中重新定义“派生类型” .....	30
2.1.2	C++的封装概念 .....	31
2.1.3	使用 struct 说明类 .....	32
2.2	说明C++的类 .....	36
2.2.1	类的说明 .....	38
2.2.2	使用类说明的语法规则 .....	39
2.2.3	说明类成员 .....	42
2.2.4	建立后进先出的堆栈类 .....	42
2.2.5	使用 this 指针 .....	49
2.2.6	使用关键字 public, private 以及 protected .....	53
2.3	编写类的成员函数 .....	54
2.3.1	联系成员函数和类 .....	54
2.3.2	提供构造函数和析构函数 .....	58
2.4	使用友元函数 .....	61
2.4.1	在类中包括友元函数 .....	61
2.4.2	决定何时使用友元函数 .....	63

### 第三章 创建 C++ 的对象

3.1 定义 C++ 的对象 .....	64
3.1.1 为类对象指定存储类 .....	64
3.1.2 定义具有动态生存期的类对象 .....	66
3.1.3 定义局部(auto)类对象 .....	80
3.1.4 定义全局(静态)的类对象 .....	88
3.2 初始化类对象 .....	96
3.2.1 使用构造函数来初始化类对象 .....	96
3.2.2 使用初始化表 .....	98

### 第四章 存取 C++ 的对象

4.1 使用作用域限制操作符 .....	100
4.1.1 作用域限制的一般用法 .....	100
4.1.2 使用作用域限制进行语法控制 .....	102
4.1.3 利用作用域限制来控制二义性 .....	104
4.2 理解 C++ 的作用域规则 .....	108
4.2.1 C 和 C++ 作用域的不同 .....	108
4.2.2 C++ 的作用域规则 .....	109
4.3 同 C++ 的对象交流信息 .....	112
4.3.1 向对象传送信息 .....	112
4.3.2 理解 * this .....	136
4.4 使用引用操作符 .....	137
4.4.1 由地址操作符进化而来的引用操作符 .....	137
4.4.2 引用操作符 .....	138
4.5 使用对象作为函数的参数 .....	139
4.5.1 通过值和通过引用来传送对象 .....	139
4.5.2 从成员函数中访问其它对象 .....	140
4.6 使用指向对象的指针 .....	141
4.6.1 理解何时需要指针 .....	141
4.6.2 定义指向对象的指针和数组 .....	142

### 第五章 重载函数和操作符

5.1 重载成员函数 .....	145
5.1.1 C++ 的重载 .....	145
5.1.2 声明重载的成员函数 .....	147
5.2 重载友元函数和非成员函数 .....	148
5.2.1 重载类的友元函数 .....	149
5.2.2 重载非成员函数 .....	153

5.3	理解类型安全的连接 .....	154
5.3.1	理解函数识别 .....	155
5.3.2	利用标准 C 的包含文件来控制连接问题 .....	156
5.4	在 C++ 中重载操作符 .....	158
5.4.1	理解操作符重载 .....	158
5.4.2	声明重载操作符函数 .....	163
5.4.3	重载二目操作符和一目操作符 .....	171
5.5	重载下标操作符和函数调用操作符 .....	173
5.5.1	使用重载的下标操作符 .....	173
5.5.2	使用重载函数调用操作符 .....	177

## 第六章 使用 C++ 的构造函数和析构函数

6.1	构造函数和析构函数调用 .....	182
6.1.1	声明构造函数和析构函数 .....	182
6.1.2	使用构造函数初始化 .....	188
6.1.3	何时调用构造函数 .....	191
6.1.4	何时调用析构函数 .....	198
6.2	重载构造函数 .....	201
6.3	写一个缺省构造函数 .....	201
6.3.1	写其它构造函数 .....	203
6.3.2	决定何时需要一个拷贝构造函数 .....	205
6.4	使用 new() 操作符和 delete() 操作符 .....	206
6.4.1	new 和 delete 的使用 .....	206
6.4.2	动态地创建和删除类对象 .....	208
6.5	重载操作符 new() 和操作符 delete() .....	210
6.5.1	重载全局操作符 .....	210
6.5.2	重载类的操作符 .....	216

## 第七章 使用 C++ 流

7.1	C++ 流介绍 .....	219
7.1.1	C++ 流与标准流的比较 .....	220
7.1.2	使用 C++ 流进行标准 I/O .....	222
7.2	处理 C++ 流的错误 .....	229
7.2.1	检测 C++ 流错误状态 .....	230
7.2.2	使用流状态成员函数 .....	232
7.3	用 C++ 流控制数据格式 .....	234
7.3.1	在内部类型中使用插入符和抽取符 .....	234
7.3.2	重载 << 和 >> 操作符 .....	238
7.4	使用 C++ 流操纵符 .....	241

7.4.1	理解 C++ 的操纵符 .....	242
7.4.2	使用操纵符改变状态和属性 .....	242
7.5	使用 C++ 文件 I/O 流 .....	250
7.5.1	读和写 fstream 文件 .....	254
7.5.2	用 C++ 流进行文件定位 .....	255
7.6	使用和转换旧式 C++ 流 .....	257
7.6.1	新旧 C++ 流的区别 .....	257
7.6.2	转化为新的 C++ 流 .....	258

## 第八章 使用 C++ 派生类

8.1	无继承性的代码重用 .....	260
8.1.1	理解代码的重用性 .....	260
8.1.2	通过组合重新使用代码 .....	261
8.2	使用单个基类 .....	264
8.2.1	理解继承 .....	264
8.2.2	声明基本和派生类 .....	265
8.3	一个实践项目:扩展内存支持 .....	271
8.3.1	扩展内存规范(XMS) .....	271
8.3.2	建立 XMS 管理类 .....	273
8.3.3	派生一个交换缓冲类 .....	285
8.4	使用虚拟函数 .....	290
8.4.1	使用迟后联编和虚拟函数 .....	290
8.4.2	使用域限定控制符来控制成员函数的存取权限 .....	293
8.5	使用多基类 .....	294
8.5.1	从多个基类中派生 .....	294
8.5.2	声明和使用虚拟基类 .....	297
8.6	从抽象类中派生类 .....	298
8.6.1	理解纯虚拟函数 .....	299
8.6.2	纯虚拟函数的实现 .....	299
8.7	在继承时使用构造函数和析构函数 .....	300
8.7.1	初始化代码不能继承 .....	301
8.7.2	理解在继承时调用构造函数和析构函数的顺序 .....	301
8.7.3	使用虚拟析构函数 .....	301

## 第九章 对象控制

9.1	用户自定义类型转换 .....	304
9.1.1	使用类的构造函数来进行类型转换 .....	304
9.1.2	类型操作符的重载 .....	306
9.2	使用 generic 类 .....	309



9.2.1	抽象和 generic 类的设计	309
9.2.2	构造 generic 类	312
9.3	控制对象的行为与性能	319
9.3.1	使用友元函数以提高效率	319
9.3.2	使用友元函数来控制对象句法	319
9.3.3	使用静态存贮类来避免重复创建	322
9.3.4	使用可引用量和指针	324
9.3.5	使用直接插入函数来消除函数调用	326
9.4	类对象控制中今后的发展方向	327
9.4.1	类和函数的模板	327
9.4.2	例外处理	328

## 第二部分 Turbo Vision 入门

### 第十章 工作的继承性

10.1	窗口应用程序的框架	332
10.2	应用程序开发的新视觉(Vision)	332
10.3	Turbo Vision 应用程序的组成	333
10.3.1	各部分的名称	333
10.3.2	一种常见的应用程序式样	334
10.4	Turbo Vision 的“Hello, World!”	335
10.4.1	运行 HELLO.CPP 程序	335
10.4.2	下拉一个菜单	336
10.4.3	一个对话框	337
10.4.4	按钮	337
10.4.5	退出对话框	338
10.5	HELLO.CPP 内部	338
10.5.1	应用程序的类	338
10.5.2	对话框的对象	339
10.5.3	执行和调试的过程	340
10.5.4	HELLO 的主程序	340
10.5.5	应用程序的实例化	340
10.5.6	成员函数 run	341
10.5.7	应用程序的析构造函数	341
10.6	建立 Turbo Vision 的应用程序	341
10.6.1	目录	342
10.6.2	编译	342
10.7	小结	343

## 第十一章 编写 Turbo Vision 应用程序

11.1	开始编写第一个 Turbo Vision 应用程序 .....	344
11.2	桌面、菜单条和状态行 .....	345
11.2.1	桌面 .....	347
11.2.2	状态行 .....	347
11.2.3	菜单条 .....	350
11.2.4	关于结构 .....	352
11.3	设计窗口 .....	352
11.3.1	窗口结构 .....	354
11.3.2	窗口的行为 .....	355
11.3.3	看透任何窗口 .....	356
11.3.4	注意事项 .....	357
11.3.5	一个简单的文件视图器 .....	358
11.3.6	带有缓冲的显示 .....	359
11.3.7	视图的上下滚动 .....	361
11.3.8	一个窗口中的多个视图 .....	364
11.3.9	功能程序块置于何处 .....	366
11.4	建立一个对话框 .....	366
11.4.1	执行一个模态对话框 .....	368
11.4.2	控制对话框 .....	369
11.4.3	选择标准控制 .....	371
11.4.4	为控制按钮标号 .....	372
11.4.5	输入行类 .....	372
11.4.6	数据的设置和获取 .....	373
11.4.7	热键及其引起的一些冲突 .....	376
11.5	对话框的其它控制项 .....	377
11.5.1	静态文本 .....	377
11.5.2	列表视图器 .....	377
11.5.3	列表框 .....	377
11.5.4	历史 .....	377
11.6	标准对话框 .....	377

## 第三部分 使用 Turbo Vision

### 第十二章 类体系

12.1	整体体系 .....	380
12.1.1	关于体系的建立 .....	380
12.2	类的类型 .....	381

12.2.1	种子类	381
12.2.2	空成员函数	382
12.3	类的初始化和派生	383
12.3.1	初始化	383
12.3.2	派    生	383
12.4	成员函数	383
12.4.1	空成员函数	384
12.4.2	缺省成员函数	384
12.4.3	虚拟成员函数	384
12.4.4	非虚拟成员函数	384
12.4.5	静态成员	384
12.5	Turbo Vision 的数据成员	384
12.6	原始类	385
12.6.1	TPoint	386
12.6.2	TRect	386
12.6.3	TObject	386
12.7	视    窗	386
12.7.1	概    述	386
12.7.2	族	387
12.7.3	终端视图	388
12.7.4	流	390
12.7.5	集    合	390
12.7.6	资源(resources)	391
12.7.7	字符串列表	391

### 第十三章 视    图

13.1	视图在 Turbo Vision 应用程序中的重要性	392
13.2	简单的视图对象	392
13.2.1	设置视图大小	393
13.2.2	建立视图的面貌	394
13.2.3	使视图实现最佳的行为	395
13.3	复杂的视图	395
13.3.1	族和子视图	395
13.3.2	族的初步介绍	396
13.3.3	族的进一步介绍	397
13.3.4	视图之间的相互关系	398
13.3.5	子视图和视图树	399
13.4	被选视图和焦点视图	401
13.4.1	如何找到焦点视图	402

13.4.2	如何设定焦点视图	402
13.4.3	焦点链	403
13.5	模态视图	403
13.6	缺省行为的修改	403
13.6.1	任选标志	404
13.6.2	growMode 标志	406
13.6.3	dragMode 标志	406
13.6.4	状态标志和 setState	407
13.7	视图颜色的设置	408
13.7.1	调色板	408
13.7.2	调色板内部情况	409
13.7.3	getColor 成员函数	410
13.7.4	缺省颜色的重新设定	410
13.7.5	增加新的颜色	411

## 第十四章 事件驱动的程序设计

14.1	用活 Turbo Vision	413
14.1.1	读入程序使用者的输入	413
14.2	事件的本质	414
14.2.1	事件的类型	414
14.2.2	事件和命令	416
14.3	事件的路线	416
14.3.1	事件从哪里来	416
14.3.2	事件到哪里去	416
14.3.3	标志事件	418
14.3.4	事件行踪的阶段	418
14.4	命令	420
14.4.1	命令的定义	420
14.4.2	命令的赋值	420
14.4.3	命令的生效和失效	421
14.5	事件的处理	421
14.6	事件记录	422
14.6.1	事件的清除	424
14.6.2	废弃事件	424
14.7	对事件处理机制的修改	424
14.7.1	集中化的事件收集	424
14.7.2	getEvent 的重设	425
14.7.3	空闲时间的利用	425
14.8	视图之间的通讯	425

14.8.1	中间视图	425
14.8.2	视图之间的消息	426
14.8.3	广播事件是谁处理过的	427
14.8.4	handleEvent 的调用	428
14.8.5	帮助上下文	428

## 第十五章 编写安全的程序

15.1	编程纵览	429
15.1.1	安全池	429
15.1.2	新旧两种删除方法的比较	431
15.1.3	非内存的出错	431
15.1.4	主要消费视图	432

## 第十六章 集 合

16.1	TCollection 类	434
16.1.1	集合大小的动态确定	435
16.1.2	集合中的混合项类型	435
16.2	集合的建立	435
16.3	迭代符成员函数	437
16.3.1	forEach 迭代符函数	437
16.3.2	firstThat 和 lastThat 迭代符函数	438
16.4	有序集合	439
16.5	字符串集合	441
16.5.1	重复算子的再次使用	442
16.6	多态集合	442
16.7	集合和内存管理	445
16.7.1	堆的有效性	445

## 第十七章 可流对象

17.1	永远滚动的流	446
17.2	"<<"和">>":两个重载操作符	447
17.3	可流类和 TStreamable	448
17.4	流管理器	448
17.5	可流类的构造函数	450
17.6	可流类的名称	451
17.7	流管理器的使用	451
17.7.1	流管理器程序的链入	451
17.7.2	流对象的建立和使用	451
17.8	基于流的集合	452

17.9	桌面的存储和恢复	453
<b>第十八章 资源</b>		
18.1	为什么要使用资源	456
18.2	资源由什么组成	456
18.3	资源文件的建立	457
18.4	资源的读取	459
18.5	字符串列表	460
18.5.1	字符串列表的建立	461
<b>第十九章 一个典型的示例</b>		
19.1	头文件 tvedit.h	466
19.2	程序 tvedit1.cpp	467
19.3	程序 tvedit2.cpp	470
19.4	程序 tvedit3.cpp	473
19.5	MAKE 文件	477
<b>第二十章 调试及其它</b>		
20.1	Turbo Vision 应用程序的调试	480
20.1.1	某部分程序不被执行的问题	480
20.1.2	“答非所问”的问题	481
20.1.3	程序挂线的问题	481
20.2	把应用程序移植至 Turbo Vision 中	481
20.2.1	净化原来的程序代码	481
20.2.2	重新考虑原来程序的结构	482
20.3	位映射域的使用	482
20.3.1	标志值	482
20.3.2	位屏蔽	483
20.3.3	位操作	483
20.3.4	小结	484
<b>第四部分 Turbo Vision 参考资料</b>		
<b>第二十一章 头文件</b>		
21.1	APP 头文件	488
21.2	BUFFERS 头文件	488
21.3	CONFIG 头文件	489
21.4	DIALOGS 头文件	489
21.5	DRAWBUF 头文件	489

21.6	MENUS 头文件	489
21.7	MSGBOX 头文件	410
21.8	RESOURCE 头文件	499
21.9	SYSTEM 头文件	490
21.10	TEXTVIEW 头文件	491
21.11	KEYS 头文件	491
21.12	TOBFSTRM 头文件	493
21.13	TYPES 头文件	493
21.14	TV 头文件	493
21.15	TVOBJS 头文件	494
21.16	VIEWS 头文件	494
21.17	类体系图	497

## 第二十二章 类参考资料

22.1	使用概述	499
22.2	TSample 类 header.h	499
22.2.1	数据成员	499
22.2.2	成员函数	500
22.3	CharScan Type SYSTEM.H	500
22.4	fpbase TOBJSTRM.H	500
22.4.1	成员函数	500
22.5	fpstream TOBJSTRM.H	501
22.5.1	成员函数	501
22.6	ifpstream TOBJSTRM.H	502
22.6.1	成员函数	502
22.7	iopstream TOBJSTRM.H	502
22.7.1	成员函数	503
22.8	ipstream TOBJSTRM.H	503
22.8.1	成员函数	503
22.8.2	友元类	504
22.9	KeyDownEvent	505
22.10	MessageEvent	505
22.11	ofpstream TOBJSTRM.H	506
22.11.1	成员函数	506
22.12	算子	506
22.13	opstream TOBJSTRM.H	507
22.13.1	成员函数	507
22.13.2	友元类	508
22.14	pstream TOBJSTRM.H	509

22.14.1	数据成员	509
22.14.2	成员函数	509
22.14.3	友元类	510
22.15	TApplication APP.H	511
22.15.1	成员函数	511
22.16	TBackground APP.H	512
22.16.1	数据成员	512
22.16.2	成员函数	512
22.16.3	相关函数	512
22.16.4	cpBackground 调色板	513
22.17	TBufListEntry BUFFERS.H	513
22.18	TButton DIALOGS.H	513
22.18.1	数据成员	514
22.18.2	成员函数	514
22.18.3	相关的函数	516
22.18.4	调色板	516
22.19	TCheckBoxes DIALOGS.H	516
22.19.1	数据成员	516
22.19.2	成员函数	516
22.19.3	相关的函数	517
22.19.4	调色板	517
22.20	TCluster DIALOGS.H	518
22.20.1	数据成员	518
22.20.2	成员函数	518
22.20.3	相关的函数	520
22.20.4	调色板	520
22.21	TCollection OBJECTS.H	520
22.21.1	数据成员	521
22.21.2	成员函数	521
22.21.3	相关的函数	522
22.22	TColorDialog COLORSEL.H	522
22.22.1	数据成员	522
22.22.2	成员函数	523
22.22.3	相关的函数	524
22.23	TColorDisplay COLORSEL.H	524
22.23.1	数据成员	524
22.23.2	成员函数	524
22.23.3	相关的函数	525
22.24	TColorGroup COLORSEL.H	525



22.24.1	数据成员	525
22.24.2	成员函数	526
22.25	TColorGroupList COLORSEL.H	526
22.25.1	数据成员	526
22.25.2	成员函数	526
22.25.3	相关的函数	527
22.26	TColorItem COLORSEL.H	527
22.26.1	数据成员	527
22.26.2	成员函数	527
22.27	TColorItemList COLORSEL.H	527
22.27.1	数据成员	528
22.27.2	成员函数	528
22.27.3	相关的函数	528
22.28	TColorSelector COLORSEL.H	529
22.28.1	数据成员	529
22.28.2	成员函数	529
22.28.3	相关的函数	530
22.29	TCommandSet VIEWS.H	530
22.29.1	成员函数	530
22.29.2	友元函数	531
22.30	TDeskInit APP.H	531
22.30.1	成员函数	531
22.31	TDeskTop APP.H	532
22.31.1	数据成员	532
22.31.2	成员函数	532
22.31.3	相关的函数	533
22.32	TDialog DIALOGS.H	533
22.32.1	成员函数	534
22.32.2	相关的函数	535
22.32.3	调色板	535
22.33	TDisplay SYSTEM.H	535
22.33.1	成员函数	536
22.34	TDrawBuffer DRAWBUF.H	537
22.34.1	成员函数	537
22.34.2	友元类和友元函数	538
22.35	TEvent SYSTEM.H	538
22.36	TEventQueue SYSTEM.H	538
22.36.1	成员函数	538
22.36.2	友元类和友元函数	539