# 面向对象的程序设计

# Borland C++ 4

新

费向东
李炳法
邹贻明
郭新明
审稿
等编

成都科技大学出版社

# Borland C++ 4.0

## ——面向对象的程序设计

费向东　　李炳法　　审稿
邹贻明　　郭新明　　等著

成都科技大学出版社
中国·成都

成都科技大学出版社出版发行
峨影印刷厂·印刷
开本:787×1092 1/16
印张:40 字数:600千字
印数:1—500册
版次:1994年7月第1版
印次:1994年7月第一次印刷
ISBN 7—5616—2867—6/TP·86

定价:39.00元

# 内 容 提 要

本书以 Borland International 公司最新推出的 Borland C++ 4.0 为载体，由浅入深围绕面向对象的主要特征为主干线，循序渐进地介绍 C++ 语言和面向对象编程设计技术。考虑到最近对 Windows 编程兴趣的迅速增加，本书还讨论了如何利用 OWL 编写各种复杂的 Windows 应用程序。为了方便广大读者学习和理解各种面向对象的编程技巧，全书贯穿了大量例程，都经过仔细调试，可以上机编译验证。适用于具有一定 C 语言编程经验的广大软件工作者，可作为大专院校学生和研究生的参考书。

需要购买、批发本书的单位和个人，请直接与四川大学计算机系联系。

地　址：成都市望江路 29 号四川大学计算机系

邮　编：610064

电　话：028—5583875—2955

2516

联系人：郭新明　邹贻明

零售邮寄一律免邮寄费

# 前　言

今天，面向对象的编程(OOP)是一个很热门的话题。对于许多人来说，C++本身就是OOP的代名词，就象许多人认为LISP就是人工智能一样。然而实际上，OOP并非一种具体的语言，把它看作一种特殊方法的结晶也许更适当。我们完全可以把OOP应用于多种程序语言中，如Pascal，Ada，BASIC以及汇编语言等，尽管那会带来相应的困难。

本书强调C++中的OOP，使用特定环境——Borland C++——作为其载体。考虑到最近对Microsoft Windows编程兴趣迅速增加，本书挑选了一些Windows编程的技术做适当讲解。读者在编写Windows程序时，应尽量理解OOP的实质。

本书是一本手册书，书中贯穿了大量编程实例。所有的例程和代码都可以被调入并编译，以验证OOP所具有的许多新特征。对于其功能和效率，我们给了大量注解。要求本书读者熟悉C语言编程。高水平的读者也能在每章后面部分找到自己感兴趣的内容。

在80年代，C语言编程技术风靡全世界。C语言代码可以被高效地移植到各种计算机平台上。软件编写速度提高，以及软件工程的引入使软件平均代码量增加，由此带来的复杂度增加了软件开发时间。今天困扰许多公司的难题便是开发时间和效率的问题。AT&T发展了C++语言作为ANSI C的扩展，试图以此为C语言增加面向对象编程的优点，同时保留C语言本身的优点，如简洁、运行效率高等。

发展C++的目的之一是使编程更容易，要做到这一点，C++肯定要比C复杂。C++中所增加的特性都是为了帮助减少难度。显然，单纯采用C++并不能自动提高软件质量，也不能自动简化软件。要获得C++的优点，你必须采用一种新的编程方法——面向对象的编程，简称OOP。

几年前，计算机科学家们注意到程序员在一定时间内不论采用何种语言均可编写和调试几乎同样行数的代码。工作量相同，但结果并不相同。编写100行C语言代码与100行汇编语言代码难度相差不大，但这100行C语言代码显然能完成较多的任务。在这种思想引导下，研究人员试图发展一种能数倍提高单个程序员能力的高级语言，以降低项目开发的时间和费用。

早在70年代，编程语言研究者们对于"对象"(object)的概念已经比较熟悉了。一个对象便是一个若干代码与数据的集合，以模拟一个物理实体(Entity)或抽象实体。对象是一个高效编程单元，原因有二：它们代表常用单元的直接抽象，同时对其调用者而言其复杂性是不可见的。开发的第一批对象是与计算机紧密相联的对象，如整数、数组和栈。一些语言(如Smalltalk)设计得与其它语言不同，其所有内容都是对象。

面向对象的编程方法着重于对象之间的关系，而不重其完成的细节。关系是联接对象的纽带，通常会发展成一棵关系树，其中一些对象类型由其它对象类型发展而来。隐藏对象的实现使用户更注重对象间的关系而非单个对象的内容。这一点非常重要，代表了它与非面向对象语言(如C语言)的本质区别，后者编程着重于函数和函数调用。

对于C++，语言本身几乎没有对象。生成对象的任务完全落到用户身上。Borland C++则集成了一定数量的对象类型，但要真正应用仍需开发相当数量的对象类型。集成的对象类型通常叫做"类系"(class hierarchies)。发展类系是OOP的一项中心任务。

本书基于 Borland C++4.0 讲解发展类系的新方法。在此之前，本书讲解C++的基本特征。C++的主要 OOP 特征在各章节中分别讲解。

本书可分为两部分。第一部分"C++"，对 C++语言作通常意义的讲解，特别注意了 Borland C++这一面向对象语言的特征。本书并不是一本完整的 Borland C++手册，只讲解如何应用其面向对象语言特征。

第二部分"OWL"，从第十章开始，讲解如何使用 OWL，带有大量实例。每一实例均经过测试，可以编译调试。

学习编程可以没有计算机，但通过上机却有益于学习。对于本书的学习，你不但要了解大量的实例代码，还应该修改它，加以完善，并进行编译调试。这需要以下系统环境：

- IBM PC AT 或兼容机
- MS—DOS 3.31 或后继版本
- Microsoft 兼容鼠标
- VGA 或更高级显示适配器
- Borland C++ 4.0
- OWL 2.0(第十章到十二章要求)
- Windows 3.1 或后继版本

第十到十二章的例子程序的工程文件是用 Borland C++ 4.0 建立的，如果使用低版本编译器，可能不能成功。

不需要 Microsoft SDK for Windows。如果你仅有 Turbo C++或 Borland C++ 4.0 以前版本，一到七章大部分代码可以顺利通过，但对第八、九两章的程序需要适当修改。

第十到十二章的 Windows 程序要求有 Borland C++4.0，但如果只有 Borland C++3.1 和 Windows 3.1，对源程序适当修改后也可通过。

参加本书编写工作的有：

张天庆、朱敏、谭胜勇、罗以宁、余建坤、廖果、王炎、刘军、谭多里、邹贻明、郭新明。

# 目　录

第十二章　列表框的变异

# 第一章 基 础

本章讨论在ANSI C基础上扩充而来的C++主要特征，但并不涉及对象和其它面向对象的结构。本章主要是为了解释那些较易理解、不需进一步讨论的C++特点。对于更多的基本内容，我们建议读者查阅Borland C++程序员指南。

与纯的面向对象语言相对，C++被认为是一种不纯的面向对象语言，其原因是它建立在常规的过程语言之上。其它一些语言（例如Eiffel或Smalltalk）是纯的面向对象语言，但并不是说纯的更好。衡量好坏往往要看语言如何应用以及要解决的是什么任务。

C++源代码在运行前必须先通过编译。编程的过程应该是多次重复编辑、编译、联接及运行这几步。尽管重复这一过程相当慢，但代码的生成却非常快，并且效率较高。C++在表达能力，运行速度和代码尺寸之间选择了一个极好的平衡点。进一步而言，Borland C++集成开发环境（IDE）自动地把开发周期的所有步骤集成在一起，极大地加快了开发速度。

因为C++设计为C的扩展改进，它包含了传统ANSI C的全部特性。事实上，任何一种软件在新版本中即使是对一些很基本的内容也作了相应修改。本章中所讲述的各节都是C语言的基本内容。C程序员在本章可能找不到什么新内容，但并不意味着可以跳到第二章，因为其中涉及一些内容的改动。

## §1.1 Borland C++工程文件的结构

就结构而言，Borland C++的工程文件与ANSI C工程文件很类似。你可以在头文件中定义结构（和类）。C++头文件的后缀是*.hpp，用以区别ANSI C头文件的后缀*.h，因为C++头文件中可能包含ANSI C下不能编译的内容。真正的C++源程序可以放在一个或多个文件中，编译后联接在一起，这与ANSI C的编译联接相同。

### §1.1.1 头文件

类标识符必须先定义后使用，通常在类定义时做这件事。要使用类的公有成员的话，不仅要求类标识符在其作用域内，还要求类被完整定义。因为类都有其定义，而且在不同的模块中都可能调用它，所以常把类定义放在头文件中，而在每一个引用到这些类的模块中包含这一头文件。

一 多次引用问题

考虑典型程序中会遇到的有大量类和头文件的情况，可能要在同一个文件中多次引用某个头文件。考虑下列文件：

```
// file HEADER.HPP
class EssentialClass {/*...*/};

// file DESKTOP.HPP
#include "HEADER.HPP"
class DeskTop {/*...*/};
```