北京科海培训中心 SAMS

Borland C++3.1 编程指南

[美] Ted Faison 著

蒋维杜 吴志美
张新宇 李景淑 译
李 莉 审校

清华大学出版社

目　录

第二部分 开发 Windows 和 DOS 应用程序

第一部分

用 Borland C++进行面向对象的程序设计

第 1 章　基础知识

本章探讨了 ANSI C 扩展或派生出的 C++语言的主要特征，但没有讲述对象或其它面向对象的构造。笔者花大量笔墨描述了 C++语言的一些很少被人了解的特征，并避免了对细小问题的冗长讨论。想了解更基本的方面，可以阅读《Borland C++程序员指南》，随标准 Borland 文档提供。

C++语言属于混合型面向对象程序设计语言，这是相对于纯面向对象的语言或传统语言而言的，因为它是建立在传统过程语言基础之上的。其他如 Eiffel 或 Smalltalk 等语言，是纯面向对象的语言，但这并没有令它们更具优势。评定一种语言是否最优要看怎样使用它以及它能解决什么任务。

C++源代码在执行前必须先编译。因此程序设计过程必然包括编辑、编译、连接、执行这样一个开发周期。尽管这一周期的反复进行是一个缓慢的过程，产生的代码却快速高效。C++语言体现了表达能力、运行速度、代码大小的最佳搭配。更有用的是，Borland C++的集成开发环境(IDE)自动地将开发周期各个阶段连接起来，使整个周期显著加快。

因此 C++语言是作为 C 语言的提高和扩展而设计的，它具有 ANSI C 的传统特征。然而，正如任何软件的新版本一样，即便一些基本项也会有变化。这些特征在本章讲述 C 语言特征时都重点指出了。C 语言程序员不会发现太多的改变，但这并不鼓励他们跳过这章到第 2 章去，因为描述的变化都是很重要的。

1.1　Borland C++项目文件的结构

在结构上，Borland C++项目文件与 ANSI C 很类似。在头文件中建立结构(和类)以及常量的声明。因为 C++头文件中可能包括 ANSI C 不能编译的构造，所以用 *.hpp 来标识，以区别于通常的 ANSI C 头文件 *.h。正如 ANSI C 一样，C++实际源代码包含在一个或多个源文件中，然后被编译、连接起来。

1.1.1　头文件

类标识符在使用前必须进入作用域，这通常是用类声明来实现的。使用类的公有成员时，类标识符不仅要在活动作用域中，而且必须被完全声明。因为每个类都有自己的声明部分，而且各种模块都可以使用它，所以常把声明置于头文件中，用到该类的每个模块再包含相应的头文件。

1.1.1.1　多次包含的难题

假设一典型的程序是由大量类和头文件组成的，常常会遇到这样一个难题：在同一文件中一个头文件可能被包含多次。例如：

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

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

//file DRAWER.C
#include "HEADER.HPP"
#include "DESKTOP.HPP"        // file HEADER.HPP included
                              // twice !
class Drawer { /* ... */ };
```

为避免这种常见错误，C++头文件用一些预处理指令产生下面的简单而有效的分离构造：

```
// file HEADER.HPP
# ifndef HEADER.HPP
# define HEADER.HPP
class Essentialclass {/* ··· */};
···
#endif
```

所有的头文件都采用相同的机制。通常在#define 语句中使用头文件的文件名或其省略形式。这样试图再次包含同一文件的错误就可以被预处理器发现并处理。

1.1.1.2 预编译头文件

处理C++源文件时，常因调入大量的头文件而花费大量时间，这使C++用户大为抱怨。有时编译器装入、扫描头文件的时间甚至比编译源文件的时间还长。经常同一头文件因被不同模块使用而一次次地装入内存。这似乎是一种时间上的浪费。

Borland 公司的工作人员发现了这一缺点，并用预编译的方法解决了它。当 Borland C++扫描源文件包含的头文件时，在磁盘上保留一个"预编译"版本。下次需包含同一头文件时，编译器读预编译过的版本，这大大加快了整个编译过程。当包含的头文件很大或数量很多时，速度的加快是显著的。最常用的头文件有 windows.h，dos.h，iostream.h 和 graphics.h 等头文件。

使用预编译的头文件，必须通过 Options|Compiler|Source 菜单选中 Precompiled Header 项。当选中该选项时，编译器扫描源文件中包含的头文件，在磁盘上 tcdef.sym 文件中存入其映象。编译器一次只能处理一个预编译的头文件，不过可以不用缺省的文件名以产生不同的预编译头文件。由于这类文件可能会变得很大，这么做必须保证有足够的磁盘空间。如果在制造预处理文件时系统缺少空间，可能会出现一些奇怪的错误信息或结果。

1.1.2 一个完整的样本程序

有关C或C++语言的书大多是从常见的"hello，world!"程序开始的。该程序包括一个小的 main( )函数，在屏幕上打印一行信息，然后终止。因为本书不是介绍性质的，所以直接从一个更复杂的，包括绝大多数基本构造的程序开始。这使读者马上接触到语言的过程部分，而不必等到读过书的四分之三之后。

实例1.1程序接收用户键入的一行字符，计算其中有多少空格，然后打印结果。

实例 1.1 一个完整的程序

```
// Count the number of spaces in a line of text
```