

21世纪高等院校计算机专业规划教材

C++语言程序设计

张思民 刘政宇 张光南 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21 世纪高等院校计算机专业规划教材

C++语言程序设计

张思民 刘政宇 张光南 主 编
宋 毅 蔡 明 副主编
杨天明 参 编

内 容 简 介

C++语言是应用最广泛的面向对象的程序设计语言之一。本书注重可读性和实用性，配备了大量经过精心筛选的例题，既能帮助理解知识，又具有启发性。

本书在内容编写上分为 C++语言基础、C++特性和 Windows 操作系统下的应用程序开发基础三个部分，突出 C++的编程思想和编程能力的培养。

本书共分 13 章，分别介绍基本数据类型、程序控制语句、函数、数组、结构体、共用体和枚举类型、指针、类、对象、继承与派生、重载、虚函数、输入输出流、Windows 编程、图形设计及应用、MFC 编程及应用、多媒体程序设计、网络编程和数据库应用等内容。

本书可以作为高等学校程序设计语言的教材，也可供从事软件开发的工程技术人员自学使用。

图书在版编目 (CIP) 数据

C++语言程序设计/张思民, 刘政字, 张光南主编. --

北京: 中国铁道出版社, 2014. 9

21 世纪高等院校计算机专业规划教材

ISBN 978-7-113-18711-8

I. ①C… II. ①张… ②刘… ③张… III. ①C 语言—
程序设计—高等院校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 180863 号

书 名: C++语言程序设计

作 者: 张思民 刘政字 张光南 主 编

策划编辑: 周海燕

读者热线: 400-668-0820

责任编辑: 周海燕

封面设计: 刘 颖

封面制作: 白 雪

责任校对: 汤淑梅

责任印制: 李 佳

出版发行: 中国铁道出版社 (北京市西城区右安门西街 8 号 邮政编码: 100054)

印 刷: 北京市昌平百善印刷厂

版 次: 2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 21.5 字数: 519 千

印 数: 1~2000 册

书 号: ISBN 978-7-113-18711-8

定 价: 42.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 51873659

C++语言是目前应用最广泛的一门面向对象的程序设计语言之一。长期以来，C++一直是学生感到最难学习的一门课程。如何学好这门语言是广大教师一直都在思考的问题。

1. 本书特点

作为一本教材，本书有以下特点。

(1) 简明易懂。本书在叙述方式上尽可能浅显，用日常生活中的例子或图示来加以说明，并用大量短小的例题进行分析解释，使读者学完每一章都可以编写出相应功能的程序。

(2) 对每一个例题均进行了详细分析和解释，既可以帮助读者学习理解知识和概念，大大降低学习难度，又具有启发性，触类旁通。

(3) 本书在内容编写上分为 C++ 语言基础、C++特性和 Windows 操作系统下的应用程序开发基础三个部分，突出 C++的编程思想和编程能力的培养。

2. 学习方法

学习 C++语言，绝不是“为了学语法而学语言”，而应该“为了实际应用而学语言”。培养面向对象程序设计能力。因此，在本书的安排上，C++语言基础部分采用 Windows 平台下的控制台方式（命令行方式）讲解以突出 C++的编程思想和编程能力的培养。在 Windows 操作系统下的应用程序开发部分比较全面地学习标准的 Windows 图形界面编程，提高学生的实际应用能力。

C++的教学应该强调教学实施的过程、知识积累的过程、能力培养的过程，使读者能快乐地学习。使每一个读者喜欢本课程，掌握程序设计的思想和方法，让读者在学习本课程过程中享受到程序设计的乐趣，培养读者在各专业领域中应用计算机解决问题的意识和能力。

对于需要计算机知识较多的专业，程序设计课程应考虑后续课程的需要。尤其是电子信息类专业的学生的后续课程中需要大量的面向过程的程序设计的基础知识，包括汇编语言的编程，单片机、嵌入式系统和 DSP 的 C 语言编程等。面向对象的程序设计其实与面向过程的程序设计是密不可分的。在本课程中，算法的描述实际上是面向过程的，而面向对象是对程序整体而言的，它使程序的整体组织更合理，使用起来更方便。教学中应该合理地将两个方面有机地结合起来，即细节上算法的编程和程序总体上的把握并重。

应该说，学习任何一种编程语言都有一定的难度。因此，要强调动手实践，多编写、多练习，“熟能生巧”，从学习中体验到程序设计的乐趣和成功的喜悦，增强学习信心。

3. 本书安排

本书从结构上分为三个部分：第 1~5 章为 C++的语法及算法基础部分，这部分是面向过程的，基本涵盖了 C 语言的内容；第 6~7 章为 C++特性部分，这部分是面向对象的基本概念和理论知识部分；第 8~13 章是 Windows 操作系统下的应用程序开发，是 C++语言的设计应用实践方面的内容。

本书虽然采用的是 Visual C++ .NET 编程环境，考虑读者的学习需求及各学校的教学系统安装条件，第 1~5 章内容的所有例题，只需将文件保存为扩展名为“.c”的文件，基本上就可以在 Visual C++ 6.x、Turbo C++、C-Free 或 Linux gcc 环境下编译运行。第 6~7 章内容的所有例题，可以不做任何修改在 Visual C++ 6.x、Turbo C++、C-Free 或 Linux g++ 环境下编译运行。第 1~13 章内容的所有例题，可以不做任何修改在 Visual C++ .NET 环境下编译运行。

4. 本教程建议授课时数

(1) 仅学习 C 语言，即第 1~5 章内容，需要 54 学时，上机练习及课程设计 18 学时。

(2) 有 C 语言基础，需要 54 学时，上机练习及课程设计不少于 18 学时。

(3) 无 C 语言基础，学习全书所有内容，需要 102 学时，上机练习及课程设计 36 学时。

读者与作者联系是十分必要的，读者可以及时从作者处获得勘误（如果有的话），也可以与作者沟通想法。编者目前有以下联系方式：

E-mail: zsm112233@163.com。

编者网站: www.zsm8.com。

本书例题的源程序、课件、视频教学材料及相关系统软件可以在编者网站(www.zsm8.com)上下载。

本书由张思民、刘政宇、张光南任主编，宋毅、蔡明任副主编，杨天明任参编，其中第 1、2、3、5 章由刘政宇完成，第 4 章及附录 A、B、C 由宋毅完成，第 7、8 章由蔡明完成，第 6、12、13 章由张光南完成，第 9、10、11 章由杨天明完成，全书由张思民统稿。为了提高教材的质量，恳请读者多提宝贵意见。

编 者

2014 年 5 月

目 录

CONTENTS >>>

第 1 章 C++语言概述	1
1.1 程序设计语言及 C++的特点	1
1.2 C++程序开发过程	2
1.3 编写简单的 C++程序	3
1.4 应用 VC++编译、调试 C++程序	5
本章小结	7
习题一	7
第 2 章 C++ 语言基础	8
2.1 简单的输入输出语句	8
2.2 数据类型及分类	10
2.2.1 数据类型	10
2.2.2 基本数据类型	11
2.2.3 数据类型转换	12
2.2.4 变量	12
2.2.5 常量	15
2.3 表达式和运算符	18
2.3.1 表达式与运算符	18
2.3.2 算术运算符	18
2.3.3 关系与逻辑运算符	21
2.3.4 位运算符	23
2.3.5 常用的标准函数	26
2.4 程序控制语句	27
2.4.1 语句	27
2.4.2 顺序控制语句	28
2.4.3 选择控制语句	30
2.4.4 switch 语句	34
2.4.5 循环语句	35
2.4.6 转向语句	42
2.5 应用实例	42
本章小结	46
习题二	47
第 3 章 函数	51
3.1 函数定义	51
3.1.1 函数的分类	51

3.1.2 函数的定义	51
3.2 函数的调用	53
3.2.1 函数调用的语法形式	53
3.2.2 函数调用的过程	54
3.2.3 函数的传值调用	55
3.2.4 函数的引用调用	57
3.2.5 函数的嵌套调用	60
3.2.6 函数的递归调用	61
3.3 函数模板	62
3.3.1 函数模板的定义	62
3.3.2 模板函数	64
3.4 变量的作用域和存储类型	65
3.4.1 变量的作用域	65
3.4.2 变量的存储类型	66
3.4.3 外部变量	69
3.5 随机函数	70
3.6 函数重载	72
本章小结	73
习题三	74
第 4 章 构造数据类型及编译预处理	76
4.1 数组	76
4.1.1 一维数组	76
4.1.2 多维数组	79
4.2 字符数组	82
4.2.1 字符数组与字符串	82
4.2.2 常用字符串处理函数	83
4.3 数组应用实践	85
4.3.1 排序	85
4.3.2 查找	91
4.3.3 统计应用	97
4.3.4 字符处理	98
4.4 构造数据类型	98
4.4.1 结构体类型	98
4.4.2 共用体类型	103
4.4.3 枚举类型	104
4.5 编译预处理	105
4.5.1 “文件包含”预处理	105
4.5.2 宏定义	106

4.5.3 条件编译.....	108
4.6 类型重定义 typedef.....	109
本章小结	111
习题四	111
第 5 章 指针	114
5.1 指针和指针变量	114
5.1.1 指针的概念	114
5.1.2 指针的运算	117
5.2 指针与数组	121
5.2.1 指向数组的指针变量	121
5.2.2 指针与二维数组	124
5.2.3 指针和字符串	126
5.3 指针与函数	129
5.3.1 函数指针	129
5.3.2 指针做函数的参数	131
5.3.3 指针型函数	132
5.3.4 带参数的 main()函数	133
5.4 指向结构体的指针	134
5.4.1 结构体指针	134
5.4.2 指向结构体数组的指针	136
5.5 应用实例	139
本章小结	142
习题五	143
第 6 章 面向对象程序设计	144
6.1 面向对象的基本概念	144
6.1.1 面向对象程序设计的基本特点	144
6.1.2 对象和类的概念	145
6.2 类与对象	147
6.2.1 类的定义	147
6.2.2 类的成员函数	150
6.2.3 对象	151
6.3 构造函数和析构函数	153
6.3.1 构造函数	153
6.3.2 重载构造函数	155
6.3.3 析构函数	157
6.3.4 复制构造函数	159
6.4 对象指针和静态类成员	161
6.4.1 对象指针	161

6.4.2	this 指针	163
6.4.3	静态类成员	164
6.5	动态内存分配	169
6.6	友元	172
6.7	继承与派生	175
6.7.1	继承	175
6.7.2	派生类的继承方式	177
6.7.3	多继承	180
6.7.4	派生类的构造函数和析构函数	183
6.8	运算符重载	184
6.9	虚函数	188
6.9.1	虚函数的定义	188
6.9.2	虚函数的访问	189
6.9.3	虚函数与重载的关系及虚函数的限制	191
6.9.4	纯虚函数与抽象类	191
6.10	应用实例	193
	本章小结	202
	习题六	202
第 7 章	输入输出流类库	207
7.1	流的概念	207
7.1.1	什么是流	207
7.1.2	C++流类库	207
7.1.3	预定义流对象	209
7.2	重载提取和插入运算符	210
7.3	常用操作输入输出流的成员函数	211
7.4	文件	214
7.4.1	文件的概念	214
7.4.2	文件的打开与关闭	216
7.4.3	文本文件的读写操作	219
7.4.4	二进制文件的读写操作	223
7.4.5	随机访问文件	225
7.5	异常处理	227
7.5.1	异常的基本类型	227
7.5.2	异常处理方法	228
7.5.3	多个异常的处理方法	231
7.6	命名空间	232
7.7	应用实例	233
	本章小结	236

习题七	237
第 8 章 Windows 程序设计基础	239
8.1 Windows 编程基础	239
8.1.1 Windows 程序设计的基本概念	239
8.1.2 事件驱动和 API 函数	242
8.2 Windows 程序主要函数与设计方法	244
8.2.1 Windows 程序的设计方法	244
8.2.2 Windows 程序的主要函数	245
8.3 利用 API 开发 Windows 程序	246
8.3.1 编写 Windows 程序	246
8.3.2 系统自动生成 Windows 程序框架	254
本章小结	255
习题八	255
第 9 章 图形程序设计	256
9.1 设备描述符和图形设备接口	256
9.2 GDI 基本绘图工具的使用	259
9.2.1 画笔工具	259
9.2.2 画刷工具	260
9.2.3 颜色的设置	261
9.3 GDI 常用绘图函数	262
9.3.1 图形区域的绘图函数	262
9.3.2 填充图形内部区域的绘图函数	265
9.4 应用实例	267
本章小结	269
习题九	269
第 10 章 MFC 编程基础	270
10.1 MFC 类库简介	270
10.2 使用向导开发 MFC 应用程序	273
10.2.1 生成 MFC 应用程序框架	274
10.2.2 MFC 应用程序结构	277
10.3 MFC 图形用户界面程序设计	278
10.3.1 控件	278
10.3.2 编辑框、静态文本框及命令按钮的使用	278
10.3.3 几个常用控件的使用	285
本章小结	288
习题十	288
第 11 章 多媒体应用程序设计	289
11.1 应用音频函数设计声音播放器程序	289

11.1.1	声音文件播放原理.....	289
11.1.2	高级音频函数.....	290
11.1.3	简单音频播放程序设计.....	291
11.2	应用 MCIWnd 设计多媒体播放器程序.....	292
11.3	图像文件显示程序设计.....	295
11.3.1	显示位图图像的相关函数.....	295
11.3.2	图像显示程序设计示例.....	296
11.4	基于.NET 基础类库的图像处理程序设计.....	298
11.4.1	.NET 平台结构的 CLR.....	298
11.4.2	.NET 基础类库的图像处理类.....	298
11.4.3	图像处理示例.....	299
本章小结	302
习题十一	303
第 12 章	网络应用程序设计	304
12.1	网络通信基础.....	304
12.1.1	网络基础知识.....	304
12.1.2	TCP 与 UDP 协议.....	306
12.2	套接字编程.....	307
12.2.1	套接字 Winsocket.....	307
12.2.2	套接字函数.....	308
12.2.3	Winsocket 网络编程示例.....	311
12.3	利用 MFC 编写 Web 浏览器程序.....	314
12.3.1	CHtmlView 类.....	314
12.3.2	CHtmlView 类的应用.....	315
本章小结	318
习题十二	318
第 13 章	数据库应用程序设计	319
13.1	数据库基础知识.....	319
13.1.1	关系型数据库及 Access 数据库.....	319
13.1.2	ODBC 数据源.....	321
13.1.3	应用程序访问数据库.....	322
13.2	MFC 的 ODBC 数据库类.....	322
本章小结	329
习题十三	329
附录 A	ASCII 码表.....	330
附录 B	函数的参数传递.....	331
附录 C	Visual C++ .NET 常用数据类型.....	332

第 1 章 | C++语言概述

C++语言是面向对象的程序设计语言，是一种已经成熟的软件设计语言。本章主要介绍 C++语言的一些基本概念和程序设计方法、简单的 C++语言程序结构和应用 Visual Studio 作为标准 C++程序开发工具的设计过程，为后面章节的学习打下基础。

1.1 程序设计语言及 C++的特点

1. 程序设计语言

编写计算机程序所使用的语言称为程序设计语言，计算机语言是人与计算机之间传递信息的工具。人们要让计算机来处理、完成某些任务，就要给计算机下达指令。这些指令是一些人和计算机都能读懂和理解的符号。计算机在运行时，首先根据事先已经确定的规则，把人用符号表达的内容读懂，再按照这些符号的意思去执行，这一系列的过程，主要是通过事先规定的符号与意义的对应关系进行的。人和计算机用这套关系进行交流。用来表达这种关系的符号系统就是计算机语言。

对人们来说，计算机能直接读懂的语言是很晦涩的，不符合人们自然语言的习惯。经过从低级到高级的发展历程，计算机语言从机器语言、汇编语言逐步发展成了高级语言。

高级语言采用接近自然语言的语句进行编程。它具有学习容易、使用方便、通用性强和移植性等特点，便于人们学习和使用。

高级语言按照程序设计思想分为两大类：一类是面向过程的程序设计语言，另一类是面向对象的程序设计语言。C++是一种面向对象的高级语言。

学习程序设计语言，就是学习语言规则及设计方法，即计算机程序的语法及算法。

2. C++语言的特点

C++语言是目前应用广泛的一种计算机程序设计语言。C++是从 C 语言发展而来的，在 1993 年，经过扩充、改进的 C 语言被正式命名为 C++。C++支持面向对象的程序设计方法，特别适合于中型和大型的软件开发项目，从开发时间、费用到软件的重用性、可扩充性、可维护性和可靠性等方面，C++均具有很大的优越性。同时，C++又是 C 语言的一个超集，在基本语法特点方面，C++语言保持与 C 语言兼容，二者没有本质上的差别，这就使得许多 C 程序不经修改就可被 C++编译通过。

C++语言与 C 语言的主要区别是编程思想上的更新，即由面向过程转变为面向对象。在 C++ 语言中引入了类与对象机制，从而引出一系列概念，包括类的定义、继承与派生、多态性等。

数据封装和隐藏是与类的定义紧密相关的现象，也是 C++语言中的一大特点。数据的封装和隐藏使重要的内部数据得到保护。

1.2 C++程序开发过程

1. 一般程序设计过程

一般来说，利用高级语言编程、解决具体问题时，要经过若干步骤，主要有分析具体问题、确定算法、编程、编辑、编译和运行。

程序设计是用计算机语言编制解决问题的方法和步骤的过程。在分析给定问题的基础上，确定所用的算法（即操作步骤）和数据结构（即数据的类型和组织形式），最后用高级语言加以实现。编制的程序必须送入计算机中，以文件的形式存放在磁盘上，这个过程称为编辑。

在编辑方式下建立起来的程序文件称为源程序文件，简称源文件，相应的程序叫做源程序。源程序是用高级语言编写的，它不能直接在机器上运行。因为计算机不能识别源程序，它仅认识规定范围内的一系列二进制代码所组成的指令数据（即指令动作所涉及的对象），并按预定的含义执行一系列动作。通常把这些计算机能识别的二进制代码称为目标代码。为了把源程序变成目标代码，就需要有“翻译”做这种转换工作。在计算机系统中实现这种转换功能的软件是编译程序，如 C++语言编译程序。对应的过程称为编译阶段。

如果在编译过程中发现源程序有语法错误，则系统会给出“错误信息”，提示用户在哪一行中可能有什么样的错误。用户见到这类提示信息后，要重新进入编辑方式，对代码行中的错误进行修改，然后对修改过的源程序重新编译。经编译之后生成的目标程序的文件叫做目标文件。目标程序还不能马上在机器上运行，因为程序中会用到库函数或者其他函数，需要把它们连成一个统一的整体，这一步工作是连接。经过连接就把分离的目标程序连成完整的可执行程序，对应的文件是可执行文件。

2. 程序设计算法及描述

如前所述，程序设计是用计算机语言编制解决问题的方法和步骤的过程。程序设计首先要解决的问题是算法设计。

什么是算法？简单地说，程序设计算法就是用计算机解决问题的方法和步骤。

描述算法的方法有很多，主要有：自然语言、流程图、盒图、伪代码、程序语言等。各种描述方法都有其优点和缺点，实际使用时要根据问题的需要选择采用。本书主要使用流程图来描述算法。

无论是面向对象程序设计语言，还是面向过程的程序设计语言，都是用 3 种基本结构（顺序结构、选择结构和循环结构）来控制算法流程的。使用流程图能比较简洁地表示其算法的逻辑结构。流程图的基本符号如图 1.1 所示。



图 1.1 流程图的基本符号

用流程图表示的三种基本控制结构如图 1.2 所示。

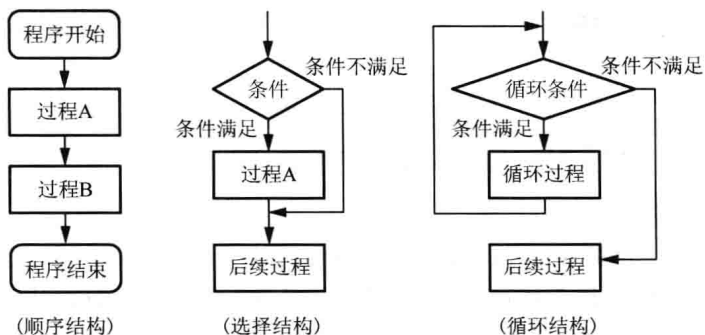


图 1.2 三种基本控制结构

3. C++程序的开发过程

要开发一个 C++ 应用程序，要经过编写源程序、编译、连接程序生成可执行程序、运行程序等步骤。C++ 程序的开发过程如图 1.3 所示。

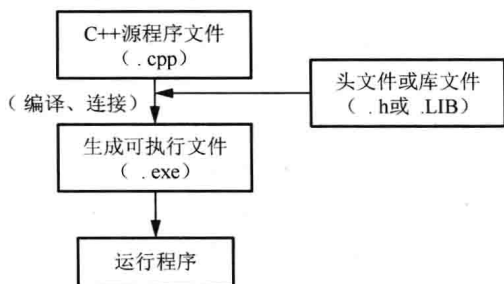


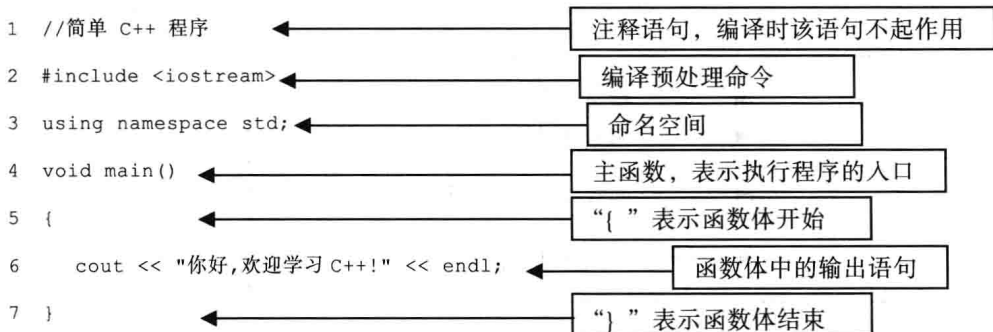
图 1.3 C++程序开发过程

1.3 编写简单的 C++ 程序

程序就是用计算机语言对要完成的任务（即功能）的描述。在设计程序时，首先要按某种规则编写源程序。源程序是一种文本文件，可以用文本编辑器进行编辑。

C++ 源程序文件的扩展名是 .cpp。下面通过一个简单示例来认识 C++ 程序的结构。

【例 1-1】 简单的 C++ 程序示例。



在 C++语言的编辑器编辑上述源程序，并将程序文件命名为 hello.cpp 保存。源程序经过编译、连接后，生成一个可执行的应用程序。运行该应用程序，其运行结果为

```
你好,欢迎学习 C++!
```

程序说明:

通过上面的例子，可以看到，一个 C++源程序包括编译预处理命令、注释和函数体三部分，

1. 编译预处理命令

本程序的第 2 行 `#include <iostream>` 是编译预处理命令。

编译预处理命令的作用是在对源程序编译之前，先对这些命令进行预处理，然后将预处理的结果和源程序一起进行正常的编译处理。C++语言中，编译预处理命令以 `#` 打头，一行只能写一条编译预处理命令。

程序编译预处理命令语句 `#include <iostream>` 中 `iostream` 是头文件的文件名。在标准 C++ 中，头文件一般以 `.h` 或 `.hpp` 为扩展名，主要提供数据类型声明、类的定义、函数的说明等信息。一般来说，C++语言系统提供的头文件名用 `<>` 括起来，如 `<iostream>`，而用户自己定义的头文件则用双引号 `"` 把头文件括起来。在本程序中，`iostream` 是 Visual C++.NET 系统提供的头文件，该头文件声明了程序所需要的输入输出操作的有关信息。`cin`、`cout`、`>>`和`<<`等操作信息都在该文件中有声明。

特别提示：不同的编译器对 C++头文件预处理的写法有不同的规定，如果在 Turbo C++环境中编译程序，编译预处理命令中要包含头文件的扩展名：`#include <iostream.h>`，且不需要使用命名空间语句 `using namespace std`。在本书中使用的 C++是 Visual Studio 环境，编译预处理命令中所包含的头文件不需要带扩展名，以后的例题不再作此说明。

2. 注释

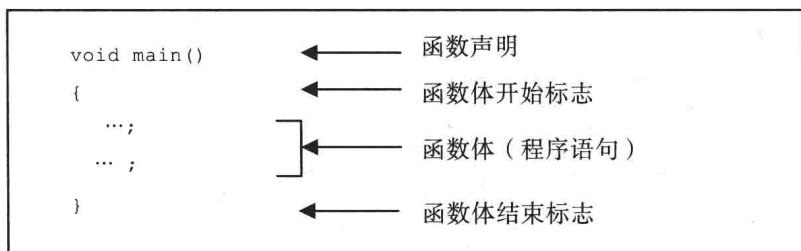
程序中标有 `//` 号的语句为注释语句。注释是程序中的说明性文字，是程序的非执行部分。它的作用是为程序添加说明，增加程序的可读性。C++语言使用两种方式对程序进行注释。

`//` 符号：表示从 `//` 符号开始到此行的末尾位置都作为注释。

`/* ... */` 符号：表示从 `/*` 开始到 `*/` 结束的部分都作为注释部分，可以是多行注释。

3. 函数

C++语言的源程序是一系列函数的集合。函数是用特定格式描述、具有特定功能的程序模块。函数由函数声明和函数体两部分组成。每个程序有且只有一个主函数，其函数名为 `main`，其结构如下：



主函数 main() 可以位于程序中的任何位置，程序的执行总是从 main() 函数开始的。

4. 语句

语句是组成程序的基本单元。函数是由若干条语句组成的。C++ 语言的语句以分号表示语句结束。

本程序的第 6 行 cout 是 C++ 语言系统定义的输出流对象，可以输出常量、变量、表达式、函数的返回值以及字符串等。

另外，C++ 语言对字母的大小写敏感，所以在书写程序语句时要注意字母的大小写。

1.4 应用 VC++ 编译、调试 C++ 程序

学习和使用 C++，需要有一个程序开发工具，本书使用的开发工具是 Microsoft Visual Studio。Visual Studio C++ 又称为 Visual C++ 或简称为 VC，它是一个集 C++ 程序编辑、编译、调试、运行等功能于一体的软件集成开发环境，是 Windows 应用程序设计领域中的主流开发工具。

下面简要介绍在 Visual Studio 开发环境下创建标准 C++ 程序的方法。

1. 启动 Visual Studio C++

Microsoft Visual Studio 正确安装完毕后，可以启动其集成开发环境。Microsoft Visual Studio 中文版的集成开发环境（Integrated Develop Environment，IDE）界面如图 1.4 所示。

2. 新建项目

选择“新建项目”选项，弹出“新建项目”对话框，如图 1.5 所示。在左边的“项目类型”栏中的 Visual C++ 目录下选择 Win32 选项，在右边的“模板”栏中选择“Win32 控制台应用程序”选项。然后确定项目名称和项目保存位置，项目名称为 t1_1。



图 1.4 Microsoft Visual Studio 2008 中文版的集成开发环境



图 1.5 “新建项目”对话框

3. 建立一个空项目

在“Win32 应用程序向导”对话框的“应用程序设置”界面中，选择“控制台应用程序”和“空项目”选项，如图 1.6 所示。

这时，建立了一个应用程序空项目框架，可以在这个空项目中建立自己开发的 C++ 程序。

4. 在当前项目中建立程序文件

在“解决方案资源”框中，右击“源文件”选项，弹出选项菜单，选择“添加”→“新

建项”命令，如图 1.7 所示。

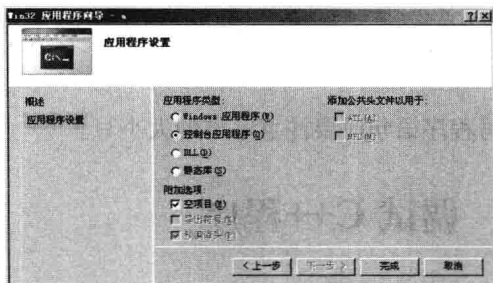


图 1.6 选择“控制台应用程序”和“空项目”选项

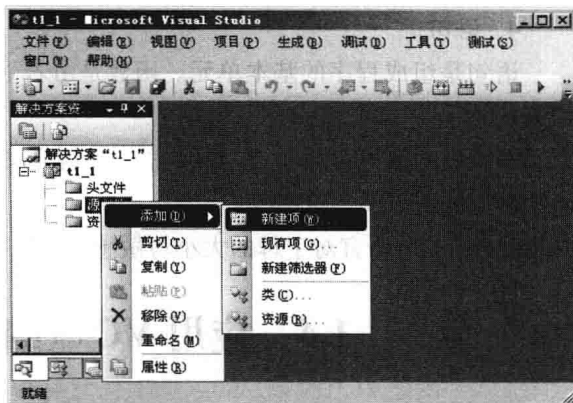




图 1.7 选择“新建项”命令

在“添加新项”对话框中，选择“C++文件(.cpp)”选项，并设置文件名为 t1_1.cpp，如图 1.8 所示。

5. 编写源程序并编译程序

在编辑窗口中，编写并保存源程序。然后单击工具栏中的“编译”按钮，编译程序，单击工具栏中的“编译、链接”按钮，将文件链接为执行程序，如图 1.9 所示。

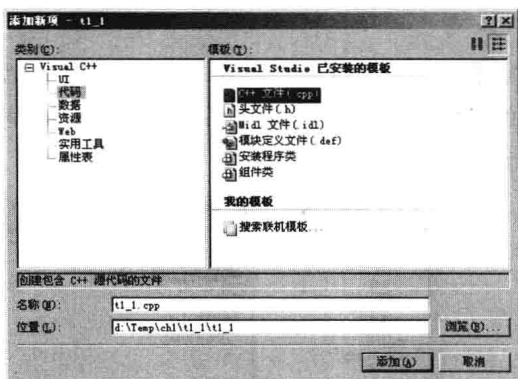


图 1.8 选择“C++文件(.cpp)”选项，并设置 C++文件名

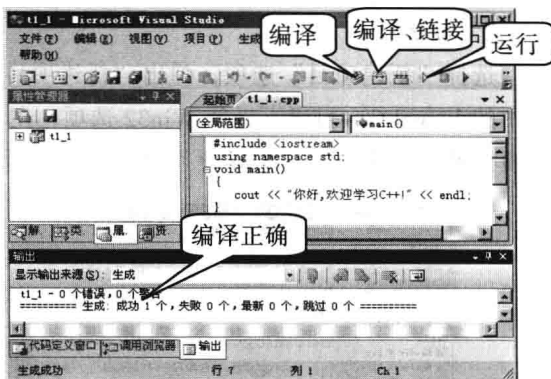



图 1.9 编译程序

6. 运行程序

单击工具栏中的“运行”按钮，则可以看到程序运行结果，如图 1.10 所示。

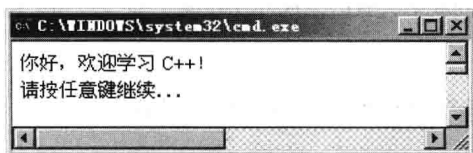


图 1.10 运行程序