



应用型本科信息大类专业“十三五”规划教材

C++程序设计 精讲与实训

朱 林 庄 丽 主 编



华中科技大学出版社
<http://www.hustp.com>

应用型本科信息大类专业“十三五”规划教材

C++程序设计精讲与实训

主 编 朱 林 庄 丽
副主编 朱长水 吴 艳 张秀国
江连海 赵凤怡

华中科技大学出版社

中国·武汉

内 容 简 介

《C++程序设计精讲与实训》从初学者的角度全面介绍了C++程序设计语言的主要概念、语法及程序设计技巧等方面的内容,以简单、实用为原则,讲解通俗易懂,行文流畅。在内容安排上由浅入深,让读者循序渐进地掌握C++编程技术。本书有两大特色:一是在“本章知识点精讲”模块明确指出相应的知识点,可以让读者一目了然,帮助读者更好更快地了解章节所涉及的知识;二是在“本章任务实践”模块使用实际开发案例,将本章所涉及的知识点融入实际案例中,可以使读者学以致用,快速掌握相应的知识点,达到更好的学习效果。本书可作为高等院校C++程序设计课程的教材,也可以作为C++语言的培训教材和工程技术人员的自学参考书。

为了方便教学,本书还配有电子课件等教学资源包,任课教师和学生可以登录“我们爱读书”网(www.ibook4us.com)免费注册下载,或者发邮件至 hustpeiit@163.com 免费索取。

图书在版编目(CIP)数据

C++程序设计精讲与实训/朱林,庄丽主编. —武汉:华中科技大学出版社,2016.3

应用型本科信息大类专业“十三五”规划教材

ISBN 978-7-5680-1501-1

I. ①C… II. ①朱… ②庄… III. ①C语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 321881 号

C++程序设计精讲与实训

C++ Chengxu Sheji Jingjiang yu Shixun

朱 林 庄 丽 主 编

策划编辑:康 序

责任编辑:史永霞

封面设计:原色设计

责任监印:张正林

出版发行:华中科技大学出版社(中国·武汉)

武昌喻家山 邮编:430074 电话:(027)81321913

录 排:武汉正风天下文化发展有限公司

印 刷:武汉鑫昶文化有限公司

开 本:787mm×1092mm 1/16

印 张:19.50

字 数:535千字

版 次:2016年3月第1版第1次印刷

定 价:39.00元



本书若有印装质量问题,请向出版社营销中心调换
全国免费服务热线:400-6679-118 竭诚为您服务
版权所有 侵权必究

前言

PREFACE

C++程序设计是一门非常重要的计算机基础课程,通过介绍程序设计的基础知识,使学生掌握高级程序设计语言的基本思想、方法和技术,理解利用计算机编写程序解决实际问题的基本过程和思维规律,从而具备相应的实践能力与创新能力,为未来应用计算机进行科学研究与实际应用奠定坚实的基础。

C++语言由C语言发展而来,它保留了C语言原有的优点,与C语言兼容,C语言写的程序基本上可以不加修改地用于C++语言。同时,C++语言又在C语言的基础上得到发展:一是基于面向过程机制对C语言的功能做了不少扩充;二是增加了面向对象机制,支持面向对象程序设计方法。在当今软件开发中,C++语言有着广泛的应用,也是高等学校最常用的程序设计教学语言之一。

本书是我们多年来进行应用型人才培养教学内容和课程体系改革的综合成果。本书内容以面向工程实践和编程能力训练为主,具有较强的可读性和应用性,为计算机程序设计课程教学内容和课程体系改革构建了一个全新的框架。本书秉着以教学案例为重点、以学生实践为主体、以教师讲授为主导的教学理念,以适应应用型高校计算机实践教育改革需求而编制的。该书的新颖之处在于:首先,每章开头介绍本章简介及本章知识目标,让读者做到心中有数;其次,对本章的知识点进行精讲,辅以经典的案例帮助读者理解相应的知识点。本书最显著的特色是结合对应的知识点讲解相应的任务实践,讲解过程从任务需求说明到技能训练要点的规划再到最终的任务实现,一气呵成,从而实现以案例驱动教学内容、以案例贯穿教与学全过程的教学方法,理论结合实际,有利于读者对相应编程思想和实践的理解与掌握。本书具有以下特色。

1. 本书内容广泛、案例丰富,其中的例题、习题及实践案例都来源于一线教学。
2. 本书按照读者在学习程序设计中遇到的问题来组织内容,随着读者对程序设计的理解和实际动手能力的提高,内容由浅入深地向前推进。
3. 本书每个知识点精讲后都给出了相应的任务实践,给出技能训练要点和任务实现,这些代码不仅能够与理论知识无缝对接,而且短小精炼,方便读者自行尝试完成。
4. 本书以学生信息管理系统项目案例贯穿始终,每章中的知识点则使用独立的例子,并辅以实例输出和任务实现,以阐述该章介绍的知识点,并在知识点

的应用过程中逐步解决相应的案例。

5. 课后的练习题有选择题、填空题、简答题、编程题,部分内容在前后章节中具有一定的延续性。

6. 本书的配套资料包含课件、实例源代码、课后练习答案。书中的源代码可以自由修改、编译,以符合自己的需要。

本书由东南大学成贤学院朱林、庄丽担任主编,由南京理工大学泰州科技学院朱长水、辽宁科技学院吴艳、青岛理工大学琴岛学院张秀国和江连海、武汉传媒学院赵凤怡担任副主编。其中,朱林编写了第1章和第2章,朱长水编写了第3章和第5章,庄丽编写了第4章,张秀国编写了第6章和附录,江连海编写了第7章和第9章,赵凤怡编写了第8章,吴艳编写了第10章。最后由朱林审核并统稿。本书在编写过程中得到了华中科技大学出版社的编辑和同行专家、学者的大力支持和帮助,在此一并表示衷心的感谢。此外,本书的编写参考了部分书籍和报刊,并从互联网上参考了部分有价值的材料,在此向有关的作者、编者、译者和网站表示衷心的感谢。

为了方便教学,本书还配有电子课件等教学资源包,任课教师和学生可以登录“我们爱读书”网(www.ibook4us.com)免费注册下载,或者发邮件至 hustpeiit@163.com 免费索取。

由于编者水平有限,书中难免有不妥之处,敬请读者和专家批评、指正。

朱 林

2015年12月

目
录

CONTENTS

第 1 章 C++概述	(1)
1.1 程序	(3)
1.2 程序结构	(4)
1.3 C++程序的开发步骤	(5)
1.4 程序的调试与运行	(5)
1.5 用 VS 建立 C++控制台程序	(8)
第 2 章 C++程序设计基础	(12)
2.1 数据类型	(12)
2.2 标识符与关键字	(14)
2.3 变量与常量	(15)
2.4 数据的输入与输出	(19)
2.5 运算符与表达式	(22)
2.6 类型转换	(28)
2.7 流程控制	(30)
2.8 数据构造类型	(40)
第 3 章 函数	(66)
3.1 定义函数	(66)
3.2 函数调用	(67)
3.3 函数的形参、实参、返回值及函数的原型说明	(68)
3.4 函数的嵌套与递归调用	(70)
3.5 内联函数	(72)
3.6 函数的重载	(73)
3.7 数组与函数	(74)
3.8 全局变量和局部变量	(77)
3.9 变量的存储属性	(79)
3.10 编译预处理	(80)
第 4 章 指针和引用	(95)
4.1 指针与指针变量	(95)

4.2	指针的运算	(97)
4.3	指针与数组	(99)
4.4	指针与字符串	(102)
4.5	动态分配内存空间	(104)
4.6	指针数组	(107)
4.7	指向一维数组的指针变量	(108)
4.8	指向指针的指针变量	(109)
4.9	引用类型的变量	(110)
4.10	指针与函数	(111)
4.11	C++中的三种参数传递方式	(112)
4.12	指针或数组名作为函数参数	(115)
4.13	链表	(116)
第5章	类和对象	(137)
5.1	类和对象	(139)
5.2	类的声明及成员的访问控制	(140)
5.3	构造函数	(143)
5.4	拷贝构造函数	(145)
5.5	构造函数的重载	(146)
5.6	析构函数	(147)
5.7	成员对象	(150)
5.8	对象数组	(152)
5.9	对象指针	(152)
5.10	静态成员	(155)
5.11	友元函数和友元类	(157)
第6章	继承和派生	(174)
6.1	单一继承	(175)
6.2	多重继承	(180)
6.3	派生类构造函数的构建	(185)
6.4	派生类析构函数的构建	(185)
6.5	多态性	(187)
6.6	抽象类	(190)
第7章	运算符重载	(203)
7.1	运算符重载的概念与规则	(203)
7.2	运算符重载为类的成员函数	(204)
7.3	运算符重载为类的友元函数	(205)
7.4	“++”和“--”的重载	(207)
7.5	重载流插入运算符和流提取运算符	(209)
7.6	转换函数	(210)
第8章	文件与流类库	(220)
8.1	输入输出的含义	(220)
8.2	C++的基本流类体系	(221)

8.3 标准的输入输出流	(221)
8.4 文件的输入输出流	(221)
8.5 文件的读写	(223)
第 9 章 模板与异常处理	(244)
9.1 模板的概念	(244)
9.2 函数模板	(244)
9.3 类模板	(247)
9.4 异常处理	(250)
第 10 章 C++ 课程设计综合实践训练	(257)
10.1 课程设计简介	(257)
10.2 课程设计教学内容	(257)
10.3 小型公司人员管理系统的设计与实现	(259)
10.4 饮料自动售卖机模拟系统的设计与实现	(281)
10.5 人机对弈游戏的设计与实现	(292)
附录 A ASCII 表	(302)
附录 B 运算符优先级与结合性表	(304)

第1章

C++概述



C++课程简介

1. C++语言

C++语言是一种应用广泛、支持多种程序设计范型的主流程序设计语言。C++是在C语言的基础之上发展起来的,它既适合于编写面向过程的程序,也适合于编写面向对象的程序。所以,在一定程度上将其称为半面向过程半面向对象的语言。

自从1946年电子数字计算机ENIAC问世以来,计算机的应用领域不断扩大,促进了计算机技术的高速发展,尤其是近年来计算机的硬件和软件日新月异。程序设计语言作为应用计算机的一种工具,得到不断的充实和完善。每年都有新的程序设计语言问世,老的设计语言不断更新换代。

在C语言推出之前,操作系统等系统软件主要是用汇编语言编写的。由于汇编语言依赖于计算机硬件,且写法较为底层,因此程序的可移植性和可读性较差。但汇编语言的好处是能对硬件直接进行操作,速度快,效率高。为了使语言的编写和可读性更接近于人的语言习惯,同时也为了提高高级语言的速度和效率,1973年,贝尔实验室的Thompson和Ritchie开发了C语言,并用它重写了UNIX的大部分代码。C语言具有以下特点。

(1) C语言是一种结构化的程序设计语言。语言本身比较简洁、使用比较灵活方便。

(2) 它具有一般高级语言的特点,又具有汇编语言的特点。除了提供对数据进行算术、逻辑运算外,还提供了二进制整数的位运算。用C语言开发的应用程序,不仅其结构性较好,且程序执行效率高。

(3) 程序的可移植性好。在某一种计算机上用C语言开发的应用程序,其源程序基本上可以不做修改,在其他型号和不同档次的计算机上重新编译连接后,就完成应用程序的移植。

(4) 程序的语法结构不够严密,程序设计的自由度大。精通C语言的程序设计者正是利用这一特点,设计出高质量的通用的应用程序。但对于初学者来说,掌握C语言并不是一件容易的事。往往是源程序编译时容易通过,程序运行时出错,且这种错误不易解决。

随着C语言应用的不断推广,C语言存在的一些不足也开始显露出来。例如:C语言对数据类型检查的机制比较弱,缺少支持代码重用的结构;随着计算机应用面的推广和软件工程规模的扩大,难以适应开发特大型的程序;软件维护困难等。在使用它做较大型项目编程的时候,由于数据的共享和函数的调用错综复杂,所以在维护方面会出现大量的问题,这样就产生了面向对象的技术,1980年,贝尔实验室的Bjarne Stroustrup博士及其同事对C语言进行了改进和扩充。在保持了C语言简洁、高效的前提下,克服了C语言存在的不足,并把Simula 67中类和对象的概念引入到C语言中。1983年由Rick Maseitti提议将改进后的C命名为C++(C Plus Plus)。后来,又把运算符的重载、引用、虚函数等功能加入到C++中,使C++的功能日趋完善,使之可以支持面向对象的程序设计,同时,它既可以支持DOS下的程序设计,也可以用来开发Windows环境下的应用程序。

C++除了继承C语言的一些特点之外,还具有以下特点。

(1)C++是C语言的一个超集,它基本上具备了C语言的所有功能。因此,用C语言开发的源程序代码可以不做修改或略作修改后,就可在C++的集成环境下编译、调试或运行。这对于推广或进一步开发目前仍有使用价值的软件是极为重要的,可节省人力和物力。

(2)C++是一种面向对象的程序设计语言。面向对象的程序设计可大大增强程序的可读性和可理解性,使得各个模块的独立性更强、更好,程序代码的结构性更加合理。这对于设计和调试大的应用软件是非常重要的。

(3)用C++语言开发的应用程序,其扩充性强,可维护性好。首先,在应用软件的开发过程中,对要解决的实际问题有一个认识、理解,再进一步认识和理解,直至客观地弄清楚问题的本质。这种认识和理解的过程,往往伴随着可能需要改变程序的结构或功能,这就要求开发工具具有较强的可扩充性。其次,对于任何一个已开发的应用软件,随着时间的推移和应用的深入,常要求增加或扩充新的功能、改进某些功能或发现程序故障。这均要求所设计的程序具有可扩充性和可维护性的特点。对于设计大的应用程序,这一特点是非常重要的。

2. C++课程地位

C++语言程序设计课程是计算机及电类、工程类各专业学生必修的专业基础课程之一,在整个教学体系中占有非常重要的地位。经过了多年的发展,C++语言的开发能力越来越强,应用领域也越来越广阔,在程序设计领域中占有不可或缺的地位。加之近年来多家公司对C++语言在可视化及通用化方面的扩充,使得C++语言的应用更加宽广而又不乏青春活力。从最底层的基于C++语言的机器人控制,到日常的Visual C++编写的一个个管理系统,再到现在流行的用ASP、NET制作的高性能网页,无处不显现着C++语言的光芒。所以,无论是考虑做一名程序员、一个网页制作者、一个系统开发人员、一个游戏开发者,或者工科的学生想通过一个简单的程序来解决自己实际专业中遇到的问题与困惑,都可以借助C++语言这样一个工具与平台达到最终的目的。本课程的学习,使学生能较系统地掌握程序设计的基本语法、程序设计基本思想的基本知识、原理和方法,初步具备分析问题、解决问题的能力,能够阅读和编写较复杂的程序,为后续课程打下基础。

3. C++课程目标

本课程教学要求学生熟练掌握高级程序设计语言程序的构成;基本语法成分;数据定义和相关运算;熟练掌握程序的三种基本控制流程的概念和实现;函数的定义和调用;熟练掌握指针的基本概念,能够应用于数据组织和函数调用;熟练掌握面向对象程序设计的核心概念——类和对象,以及类的特性(封装、继承、多态);熟练掌握用C++定义类和操作对象的方法;有关派生类的构造机制;多态的实现技术等问题。

要求学生能够建立基本的计算机程序设计概念体系和基本的程序设计方法。理解和掌握C++语言的基本语法和语义,初步理解面向对象的思想,初步掌握面向对象程序设计的方法。逐步提高学生的编程能力和调试程序的能力。能够使用结构化思想和面向对象的思维方法设计实用性较强的小型应用程序,并能够在集成环境下(例如Visual C++或VS的各版本)调试运行通过。

用C++语言解决实际问题时,对所设计的程序是有质量要求的。通过本课程的学习,所设计的程序应达到以下几个方面的基本要求。

(1)程序的正确性。首先要求程序正确无误,包括语法和语义正确,以及算法描述正确。这是对程序的最基本的要求。

(2) 程序的可读性和可理解性好。设计的程序被他人阅读时,要易于读懂,容易理解程序的设计思想和设计方法。为了保证程序的可读性和可理解性,第一,要求程序的结构性好,应采用软件工程的程序设计方法来设计程序;第二,在程序中增加足够的注解,说明程序设计的思想和方法;第三,程序的书写格式必须规范。

(3) 程序的可维护性好。这要求程序易于修改,易于增加新的功能。

(4) 程序的结构好,而且执行速度快。

必须指出,要想设计出高质量的程序,仅学习本课程的知识是不够的,还要掌握数据结构、算法设计与分析、软件工程及程序设计方法学等知识。本课程特别强调实践,只有把理论学习与大量的上机实践相结合,才能学好本课程,才可能设计出高质量的程序。



课程贯穿项目介绍

本书选取读者熟悉的学生管理系统作为本书的贯穿案例,对应每一章的具体知识点都有相应的项目案例实践,用来实现系统的模块功能。本书重点讲解管理系统中增加、删除、修改、查询功能的实现,辅以排序和查找的功能。对于系统中数据存储的讲解也是一个循序渐进的过程,在本书中先用数组来存放管理系统中的数据,然后分析数组中存放数据的弊端,进而使用链表来存放系统中的数据,最后再扩展到使用文件来存储管理系统中的数据,真正有了数据仓库的模型。系统的实现跟随各个章节知识点的讲解逐步介绍给读者,使读者能够知道所学的理论知识在实践中应该如何运用,达到学以致用目的。



本章知识目标

本章需要读者了解 C++ 的基本概念和特点;学会编写简单的 C++ 应用程序;了解 C++ 应用程序的基本结构及编程中需要注意的问题,学会使用 C++ 的开发环境,会在开发环境下对应用程序进行编译、构建、运行等操作,学会 C++ 语言的输入输出的写法,会使用输入输出语句做简单的输出测试及实现应用程序的界面。



本章知识点精讲

1.1 程序

在日常生活中,我们其实在不断地编写程序并执行,只不过人们并没有明确地意识到而已。举个例子,我们现在要用全自动洗衣机洗衣服,应该怎么做呢?尽管简单,我们还是按照一般人的习惯来描述一下吧。

第一步,就是要把脏衣服扔进洗衣机。

第二步,打开上水的水龙头并安装好电源插头。

第三步,放入洗衣粉。

第四步,按下洗衣机的开始按钮。

第五步,等待洗衣机洗完衣服(当然,不妨去干点什么别的事情)。在洗衣机提示洗完的蜂鸣声响了以后,就可以从洗衣机中拿出干净衣服去晾晒了。

上面所描述的五个步骤,就是人们洗衣服的“程序”。也许不同的人使用的步骤并不完

全一样,例如将第一步和第二步互换一下,也同样能将衣服洗干净,所以干一件事的“程序”可以不唯一,这也是计算机程序的一个特点。

对于计算机来说,程序就是由计算机指令构成的序列。计算机按照程序中的指令逐条执行,就可以完成相应的操作。实际上计算机自己不会做任何工作,它所做的工作都是由人们事先编好的程序来控制的。程序需要人来编写,使用的工具就是程序设计语言。

1.2 程序结构

程序结构是程序的组织结构,指该程序语言特定的语句结构、语法规则和表达方式,其内容包括代码的组织结构和文件的组织结构两部分。只有严格遵守这种规则,才能编写出高效、易读的程序。否则,写出的代码将晦涩难懂,甚至不能被正确编译运行。

本章通过一个简单程序向读者讲解 C++ 程序的基本结构,同时也说明 C++ 程序中输入输出操作的方法,以便后续章节中的讲解。

例 1-1 一个简单的 C++ 程序。

```
/*第一部分*/
//这是一个演示程序,它从命令行读入一个整数,然后加 1 再输出
/*第二部分*/
#include <iostream.h>
/*第三部分*/
void main( )
{
    cout<<"This is a c++ program";
}
```

C++ 程序通常会包括例 1-1 中所示的 3 部分。

1. 第 1 部分——注释部分

第 1 部分是整个文件的注释,注释内容是为了增加程序的可读性,系统不编译注释内容,自动忽略从“/*”到“*/”之间的内容。Visual C++ 6.0 中以“//”开头直到本行结束的部分也是注释。与“/*……*/”的区别在于“//”只能注释一行,不能跨行,这种注释也称为行注释,而“/*……*/”注释可以跨行,称为块注释。

2. 第 2 部分——预处理部分

第 2 部分是预处理部分,它是在编译前要处理的工作。这里是以 #include 说明的头文件包含代码 #include <iostream.h>,它指示编译器在预处理时,将文件 iostream.h 中的代码嵌入到该代码指示的地方。其中 #include 是编译指令。头文件 iostream.h 中声明了程序需要的输入输出操作的信息,在 C++ 中,用标准输入设备(键盘)和标准输出设备(显示器)进行输入输出时,使用输入输出流中的对象 cin(输入)和 cout(输出)来完成,它们的定义就属于头文件 iostream.h,所以不使用 #include <iostream.h>,就不能使用上述的输入输出对象。需要注意的是,在 Visual Studio 中,有时还会看到 #include <iostream> 的引入方式,这也是 Visual Studio 中鼓励使用的方式。但是采用这种方式时,还需要用“using namespace std;”引入 std 命名空间。

在编译源程序时先调用编译预处理程序对源程序中的编译预处理指令进行加工处理后,形成一个临时文件,并将该临时文件交给 C++ 编译器进行编译。由于编译预处理指令不属于 C++ 的语法范畴,为了把编译预处理指令与 C++ 语句区分开来,每一条编译预处理

理指令单独占一行,均用符号#开头。根据编译预处理指令的功能,将其分为三种:文件包含、宏和条件编译。

文件包含是在一个源程序文件中的任一位置可以将另一个源程序文件的全部内容包含进来。include 编译预处理指令可以实现这一功能。该编译预处理指令的格式为:

```
#include <文件名>
```

include 指令的作用是:要求编译预处理程序将指定“文件名”的文件内容替代该 include 指令行。这种文件约定的扩展名为“h”,h 是 head 的缩写。所以,将这种文件名称为“头文件”。

编译预处理部分的内容将在后面章节中详细介绍。

3. 第3部分——主要部分

第3部分是代码的主要部分,它实现了一个函数,结构如下:

```
void main()
```

```
{
```

```
...
```

```
}
```

程序中定义的 main() 函数又称为主函数,其中 main 是函数名,void 表示该函数的返回值类型,在 C++ 中,void 表示返回类型为“空”。任何程序必须有一个且只能有一个主函数,且程序的执行总是从主函数开始,其他函数只能被主函数调用或通过主函数调用的函数所调用。关于函数定义及调用的内容会在后面章节做详细介绍。

程序中的代码 cout<<"This is a c++ program;" 在执行后屏幕上会出现 This is a c++ program 这句话,即 C++ 输出语句中双引号中的内容会被原样输出。

1.3 C++ 程序的开发步骤

针对一个实际问题,用 C++ 语言设计一个实用程序时,通常要经过如下五个开发步骤。

(1) 用户需求分析。根据要解决的实际问题,分析用户的所有需求,并用合适的方法、工具进行详细描述。

(2) 根据用户需求,设计 C++ 源程序。利用 C++ 的集成环境或某一种文本编辑器将设计好的源程序输入到计算机中的一个文件中。文件的扩展名为 .cpp。

(3) 编译源程序,并产生目标程序。在编译源程序文件时,若发生语法或语义错误,要修改源程序文件,直到没有编译错误为止。编译后,为源程序产生了目标文件。在计算机上,目标程序文件的扩展名为 .obj。

(4) 将目标文件连接成可执行文件。将一个或多个目标程序与库函数进行连接后,产生一个可执行文件。在计算机上,可执行文件的扩展名为 .exe。

(5) 调试程序。运行可执行程序文件,输入测试数据,并分析运行结果。若运行结果不正确,则要修改源程序,并重复以上的过程,直到得到正确的结果为止。

1.4 程序的调试与运行

Visual C++ 6.0 是 Microsoft 公司在 1998 年推出的基于 Windows 9x 和 Windows NT 的优秀集成开发环境。该开发环境为用户提供了良好的可视化编程环境,程序员可以利用该开发环境轻松地使用 C++ 源代码编辑器、资源编辑器和使用内部调试器,并且可以创建项目文件。Visual C++ 6.0 不仅包含编译器,而且包含许多有用的组件,通过这些组件的协同工作,可以在 Visual C++ 6.0 集成环境中轻松地完成创建源文件、编辑资源,以及对程序的编译、连接和调试等各项工作。

1. 启动 Visual C++ 6.0

成功地安装了 Visual C++ 6.0 以后,可以在“开始”菜单中的“程序”选项中选择“Microsoft Visual Studio 6.0”级联菜单下的“Microsoft Visual C++ 6.0”命令,启动 Visual C++ 6.0,进入 Visual C++ 6.0 的集成环境,如图 1-1 所示。

2. 创建项目

若开始一个新程序的开发,必须先用 AppWizard(应用程序向导)建立新工程项目。

1) 建立新工程项目

在“文件”菜单下,选择“新建”命令,弹出“新建”对话框的“工程”标签,如图 1-2 所示。Visual C++ 可为用户创建用于多种目的的项目,如创建 DOS 平台及 Windows 平台下的项目文件;创建数据库项目、动态链接文件等,如在“工程”标签下选择“Win32 Console Application”项,可创建一个基于 DOS 平台的项目文件,在“位置”编辑栏中选择该工程项目所存放的位置,在“工程”编辑栏中输入该项目名。单击“确定”按钮,弹出创建 Win32 Console Application 项目步骤一对话框,如图 1-3 所示。

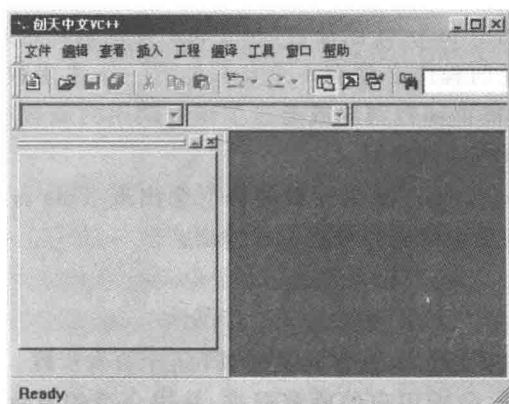


图 1-1 Visual C++ 界面



图 1-2 “新建”对话框中的“工程”标签

该对话框提供了四种项目的类型,选择不同的选项,意味着系统自动生成一些程序代码,为项目增加相应的功能。如选择“An empty project.”选项,则生成一个空白的项目,单击“完成”按钮,完成创建新项目,并生成一个工作区文件,扩展名为 .dsw。

2) 打开已有的项目

在“文件”菜单下,选择“打开”命令,或使用工具栏中的“打开”按钮,弹出“打开”对话框,该对话框可用于打开任何类型的文件,如 C++ 头文件、资源文件等。通过打开工作文件(扩展名为 .dsw),可打开相应项目。选择相应文件后,单击“打开”按钮,即可打开已有的工作项目,可编辑项目中的各个文件内容。

打开已有的项目文件,也可直接选择“文件”菜单下的“打开工作区”命令,弹出“打开”对话框,选择相应项目的工作区文件,单击“打开”按钮即可打开项目文件,如图 1-4 所示。

3. 编辑源代码文件

1) 建立新源代码文件

创建的空白项目中没有任何文件,这时可在“文件”菜单下选择“新建”命令,弹出“新建”对话框,选择“文件”标签,在该标签中选择新建的文件类型,如果是新建 C++ 源程序文件,则选择“C++ Source File”选项。在“File”编辑栏中输入新建文件名,单击“确定”按钮便激活文件编辑窗口,在此窗口中就可以输入源代码文件的内容了。

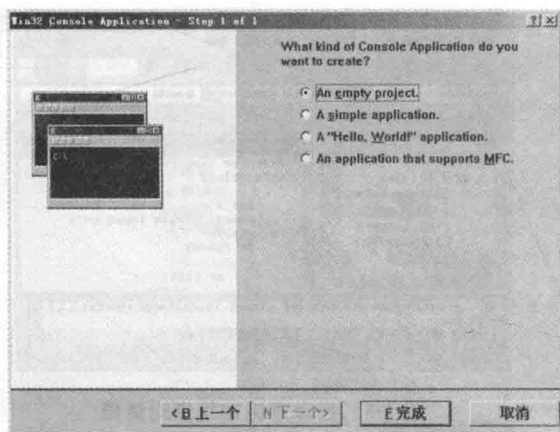


图 1-3 创建 Win32 Console Application 项目步骤一对话框

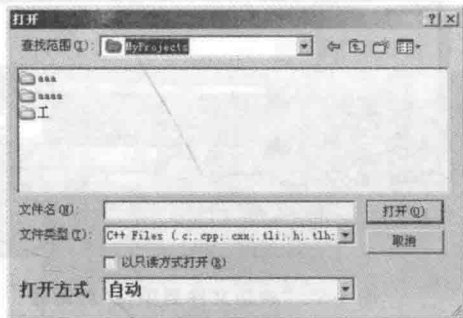


图 1-4 “打开”对话框

2) 在项目文件中添加一个文件

若在已有的项目文件中增加一个文件,如 C++ 源文件 *.cpp 或头文件 *.h,需要执行如下步骤:

打开相应的项目文件,选择“工程”菜单下的“Add to project”级联菜单中的“File”命令,弹出“Insert Files into Project”对话框,在该对话框中选择需要加入的文件名,并在“插入到”编辑栏中选择相应的项目文件名,单击“确定”按钮,即可将选择的文件加入到相应的项目文件中,如图 1-5 所示。

4. 保存源程序文件

编辑完源程序文件,可直接用鼠标单击工具栏中的“Save”按钮,或选择“File”菜单下的“Save”命令,保存源程序文件。

5. 编译源程序文件

先激活相应源程序文件窗口,选择“Build”菜单中的“Compile”命令,或按快捷键 Ctrl+F7,编译该源程序文件形成目标文件.obj 文件。若该项目文件中包括多个源程序文件,可依次激活并编译成.obj 文件。

6. 链接目标程序,形成可执行文件

选择“Build”菜单中的“Build”命令,或按快捷键 F7,可将目标文件(.obj 文件)链接并形成可执行文件。

在编译或链接时,在“Output”窗口显示系统在编译或链接程序时的信息,如图 1-6 所示。若编译或链接时出现错误,则在该窗口中标识出错误文件名、发生错误的行号及错误的原因等。须找出错误原因加以修改,然后再编译、链接直至形成可执行文件。

7. 运行程序

成功地建立了可执行文件后,即可执行“Build”菜单中的“Execute”命令或按 Ctrl+F5,执行该程序。执行 MS-DOS 程序时,Windows 自动显示 DOS 窗口,并在 DOS 窗口中列出运行结果。

读者可以依照上述步骤对本章案例进行编译和输出。

当程序较小时,用一个文件就可以保存所有代码。但是有实际用途的程序一般都不会太小。所以,通常会将程序分成几个文件分别保存,再通过包含语句放到一起。这种做法既有利于模块化开发,也有利于代码的重用。



图 1-5 “添加文件到项目”对话框



图 1-6 编译或链接程序对话框

C++ 程序的文件类型有 .h 和 .cpp 两种,前者是头文件,后者是代码的实现文件。头文件中包含了类、函数、常量、全局变量等的声明,使用时用 #include 语句在程序的预处理部分包含进来即可。代码的实现文件是对头文件中声明的类、函数等的具体实现。不需要显示包含 .cpp 文件,当包含了与它对应的 .h 文件后,编译器会自动去找同名的 .cpp 文件。

1.5 用 VS 建立 C++ 控制台程序

随着操作系统的发展,目前 Visual Studio 6.0 在很多计算机上无法安装,所以需要使
用 VS 的更高版本来编译 C++ 程序,例如 VS2010、VS2012、VS2015 等,编译过程如下。

(1) 打开 VS 的主界面,然后选择“文件”→“新建”→“项目”来创建一个空工程,在弹出的对话框中选择 Win32 控制台程序,如图 1-7 所示。

(2) 在图 1-7 中输入相应的应用程序名称和应用程序位置,单击“确定”按钮,弹出图 1-8 所示的界面。



图 1-7 新建项目

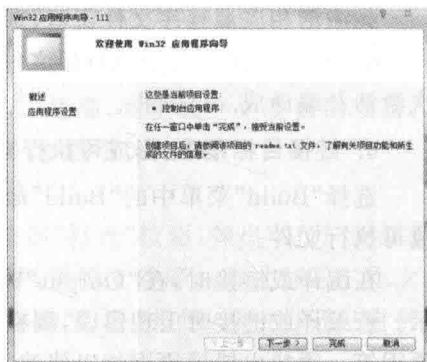


图 1-8 Win32 应用程序向导

(3) 单击“下一步”按钮,选中控制台应用程序并勾选“空项目”,其他默认,然后单击“完成”按钮,如图 1-9 所示。

(4) 在生成的界面中,右键单击左侧的“源文件”,然后选择“添加”→“新建项”命令,选中“C++ 文件(.cpp)”,如图 1-10 和图 1-11 所示。

如果在左侧边栏没有出现“解决方案资源管理器”,则只需将重置窗口布局即可,即选择“窗口”→“重置窗口布局”命令。

(5) 在编辑页面编写代码并编译运行,如图 1-12 所示。

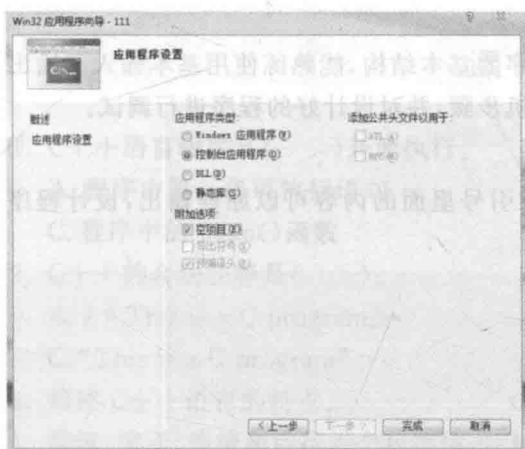


图 1-9 Win32 应用程序设置

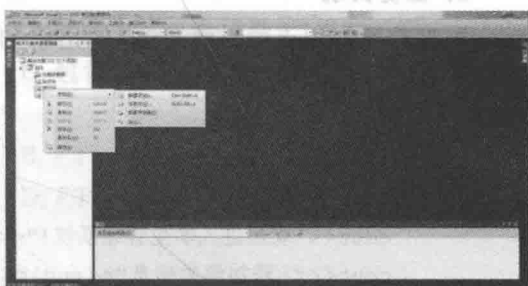


图 1-10 项目主界面



图 1-11 文件类型选择

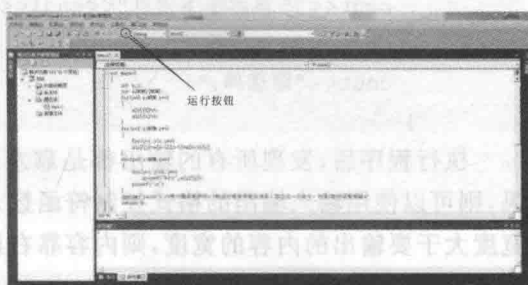


图 1-12 编辑并运行

用 VS 编译 C++ 程序时,编译出的结果闪一下很快就会消失。在编写程序的过程中,如果在程序的最后加上一句代码“system(“pause”);”,VS 就能让编译框暂停。



本章任务实践

1. 任务需求说明

编程实现学生管理系统的主页面,可以提示用户将要进行的操作,如图 1-13 所示。

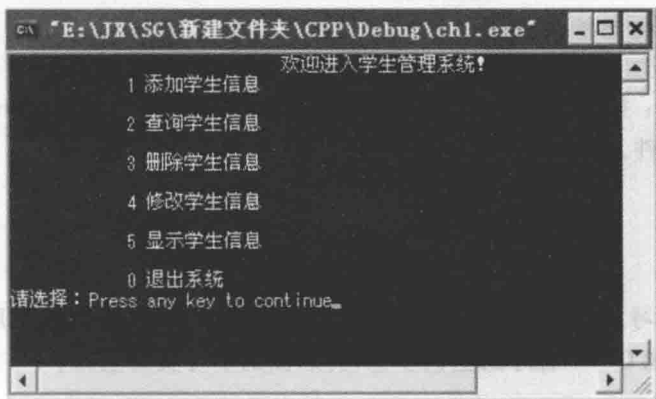


图 1-13 学生管理系统的主页面