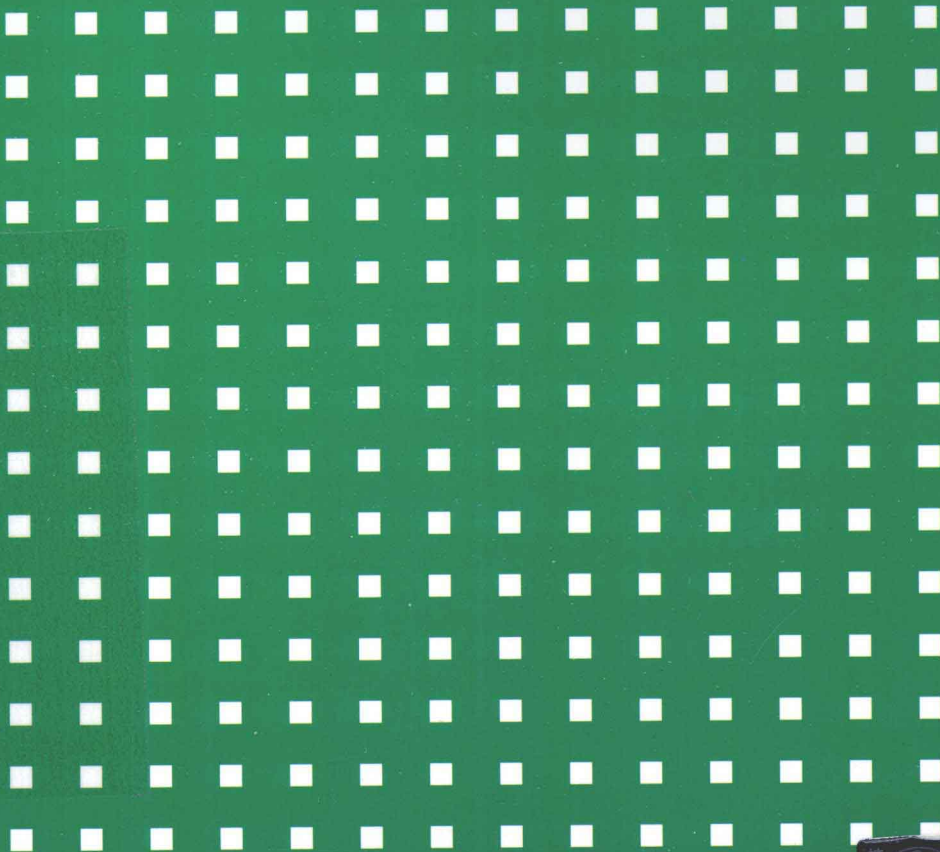


C/C++程序设计 (第2版)

实验与习题解析

张树粹 主编

卢云宏 郭艳燕 刘迎军 副主编



清华大学出版社

高等学校计算机专业教材精选·算法与程序设计

C/C++程序设计 (第2版) 实验与习题解析

张树粹 主 编
卢云宏 郭艳燕 刘迎军 副主编

清华大学出版社
北京

内 容 简 介

本书是《C/C++ 程序设计》(第2版)的配套教学用书,编者基于多年的教学经验,注重 C/C++ 本身的系统性与认知规律的结合,针对初学者的特点,在写法上,力求深入浅出,通俗易懂;在结构和内容上,力求准确定位,强化实验。本书针对主教材各章内容配有实验,对程序的调试过程和方法细致地进行分析讲解,特别是针对编译出现的常见错误给予分析和纠正,对于提高学习者的动手能力有很大帮助。同时,本书针对教材中精选的习题进行精辟解析,并采用 Visual C++ 6.0 编译系统调试成功。对于有提高编程水平要求的学生,书中还提供了补充提高习题,以达到分层次教学的目的。

本书适合作为普通高等院校、高职高专、各类成人教育院校作为开设程序设计基础课程的教材,也可作为编程人员和参加计算机考试(C/C++ 模块)自学者的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C/C++ 程序设计(第2版)实验与习题解析/张树粹主编. —北京:清华大学出版社,2012.6
(高等学校计算机专业教材精选·算法与程序设计)
ISBN 978-7-302-28506-9

I. ①C… II. ①张… III. ①C 语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 064980 号

责任编辑:白立军 战晓雷

封面设计:傅瑞学

责任校对:白 蕾

责任印制:张雪娇

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印装者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:12.5

字 数:297 千字

版 次:2012 年 6 月第 1 版

印 次:2012 年 6 月第 1 次印刷

印 数:1~3000

定 价:25.00 元

产品编号:046826-01

出版说明

我国高等学校计算机教育近年来迅猛发展,应用所学计算机知识解决实际问题,已经成为当代大学生的必备能力。

时代的进步与社会的发展对高等学校计算机教育的质量提出了更高、更新的要求。现在,很多高等学校都在积极探索符合自身特点的教学模式,涌现出一大批非常优秀的精品课程。

为了适应社会的需求,满足计算机教育的发展需要,清华大学出版社在进行了大量调查研究的基础上,组织编写了《高等学校计算机专业教材精选》。本套教材从全国各高校的优秀计算机教材中精挑细选了一批很有代表性且特色鲜明的计算机精品教材,把作者们对各自所授计算机课程的独特理解和先进经验推荐给全国师生。

本系列教材特点如下。

(1) 编写目的明确。本套教材主要面向广大高校的计算机专业学生和理工科学生,使学生通过本套教材,学习计算机科学与技术方面的基本理论和基本知识,接受应用计算机解决实际问题的基本训练。

(2) 注重编写理念。本套教材作者群为各校相应课程的主讲,有一定经验积累,且编写思路清晰,有独特的教学思路和指导思想,其教学经验具有推广价值。本套教材中不乏各类精品课配套教材,并力图努力把不同学校的教学特点反映到每本教材中。

(3) 理论知识与实践相结合。本套教材贯彻从实践中来到实践中去的原则,书中的许多必须掌握的理论都将结合实例来讲,同时注重培养学生分析、解决问题的能力,满足社会用人要求。

(4) 易教易用,合理适当。本套教材编写时注意结合教学实际的课时数,把握教材的篇幅。同时,对一些知识点按教育部教学指导委员会的最新精神进行合理取舍与难易控制。

(5) 注重教材的立体化配套。大多数教材都将配套教师用课件、习题及其解答,学生上机实验指导、教学网站等辅助教学资源,方便教学。

随着本套教材陆续出版,相信能够得到广大读者的认可和支持,为我国计算机教材建设及计算机教学水平的提高,为计算机教育事业的发展做出应有的贡献。

清华大学出版社

前 言

C/C++ 语言是国际上广泛应用的计算机程序设计语言,以功能强大、表达灵活、代码效率高和可移植性好而著称。因此,C/C++ 广泛应用于编写各种系统软件和应用软件。

编者的《C 语言程序设计》于 2005 年出版以来,得到了广大师生的认可,经过几年的教学检验后被评为山东省优秀教材。经过修订,编者对全书内容书做了认真细致的调整,并扩充了有关 C++ 的部分内容,力求以通俗、简练语言叙述 C/C++ 语言中的概念、语法和程序设计方法,从而编写了《C/C++ 程序设计》。

《C/C++ 程序设计》以面向过程程序设计为基础,引进了 C++ 新增的运算符重载、函数重载、类和对象的封装性等,使 C 语言和 C++ 有机结合在一起,使学生在编写面向过程程序的同时,能够运用 C++ 的新增功能简化程序、提高效率,使读者在具备基本编程能力的基础上循序渐进地向面向对象程序设计过渡。

本教材为《C/C++ 程序设计》(第 2 版)的配套教材,基于多年的教学经验,编者注重 C/C++ 本身的系统性与认知规律的结合,针对初学者的特点,在写法上,力求深入浅出,通俗易懂;在结构和内容上,力求准确定位,强化实验。全书针对主教材各章的内容配有实验,将程序的调试过程和方法细致地进行分析讲解,特别是针对编译出现的常见错误给予分析和纠正,对于提高学习者的动手能力有很大帮助。同时针对教材中精选的习题进行精辟解析,并采用 Visual C++ 6.0 编译系统调试成功。对于有提高编程水平要求的学生,本书还提供了补充提高习题,以达到分层次教学目的。

本书的主要特点可以概括如下。

1. 定位准确,取舍合理。本书是针对高校计算机及其相关专业、非计算机专业计算机基础教育、高职高专计算机及相关专业的程序设计基础课而编写的。根据不同层次的教学要求,本书内容可灵活取舍,而不失其内容的科学性与系统性。

2. 精选例题,通俗易懂。为使 C/C++ 程序设计的基本概念、基本理论叙述更加通俗易懂,本书选用了作者多年积累下来的大量教学实例,为了使学生熟练掌握 C/C++ 语序设计的基本方法,作者精心调试了所有习题,并加以解析,并把重点始终放在分析求解问题的思想方法上。

3. 合理设计,综合实例。每一章后面都选编了与本章教学内容紧密相关的实验题目,配合调试技术对题目进行剖析,方便了教与学。

4. 循序渐进,为面向对象程序设计打下基础。本教材以面向过程程序设计为主,同时介绍了 C++ 对 C 语言的改进,引进了 C++ 的运算符、允许函数重载,对类和对象的封装性进行了叙述,为面向对象编程打下基础。

5. 重视基础教学的同时,采取分层次教学。对于有提高编程水平要求的学生提供了补充提高习题,以达到分层次教学目的。

6. 本书提供教师教学使用的电子教案、例题源代码和习题参考答案等。本书配套电子教案及书中相关源程序代码等教学资源均可从清华大学出版社网站(<http://www.tup.>

com.cn)本书页面中下载。

本书由张树粹主编,卢云宏、郭艳燕、刘迎军副主编。具体分工如下:本书实验部分由卢云宏编写;习题解析部分的第1、2章由张树粹编写,第3、5、6章由刘迎军编写,第4、7、8、9章由郭艳燕编写,于京艳参与了部分章节的编写和程序调试。本书统稿工作由张树粹、卢云宏共同完成。

本书适合作为普通高等院校、高职高专、各类成人教育院校作为开设程序设计基础课程的教材,也可作为编程人员和参加计算机考试(C/C++模块)自学者的参考书。

在本书的编写过程中,参考了有关C语言和C++程序设计的书籍和资料,编者在此对有关作者表示最诚挚的谢意!

由于水平有限,疏漏之处在所难免,请各位读者不吝指正。

编者

2012年3月

目 录

第一部分 实 验

实验 1	C/C++ 语言程序调试初步	3
实验 2	认识数据类型	15
实验 3	使用运算符与表达式进行计算	19
实验 4	顺序结构程序设计	23
实验 5	选择结构程序设计	29
实验 6	循环结构程序设计	34
实验 7	数组及其应用	39
实验 8	函数及其应用	44
实验 9	指针及其应用	49
实验 10	结构体和共用体及其应用	54
实验 11	文件操作	58
实验 12	C 语言编译预处理	62

第二部分 习 题 解 析

第 1 章	概述	69
第 2 章	数据类型及表达式	74
第 3 章	程序控制结构	82
第 4 章	数组	98
第 5 章	函数	112
第 6 章	指针	131
第 7 章	构造数据类型	144
第 8 章	文件	163
第 9 章	编译预处理	166
附录 A	补充习题答案	171
附录 B	实验报告样本	186
附录 C	Visual C++ 常见错误信息	187
参考文献		190

第一部分

实 验

实验 1 C/C++ 语言程序调试初步

一、实验目的

(1) 熟悉 Windows 环境下 Visual C++ 6.0 的运行环境,学会独立使用系统进行编辑、编译、连接和运行 C/C++ 程序。

(2) 掌握调试一个简单的 C/C++ 程序的方法和步骤,并学会分析出错信息和修改程序当中的错误。

(3) 通过运行简单的 C/C++ 程序,初步了解 C/C++ 程序的一般编写方法。

二、实验内容

(1) 认真阅读教材中介绍的 Visual C++ 6.0 上机环境及方法,按照实验范例中介绍的内容进行操作,掌握一个简单的 C/C++ 程序的实现方法和步骤,为调试后续程序打下基础。

(2) 通过本次实验,了解运行一个 C/C++ 程序,从输入源程序开始,经过编辑源程序文件(.cpp)、编译生成目标文件(.obj)、连接生成可执行文件(.exe)和执行 4 个步骤。

(3) 要求了解 Visual C++ 6.0 的主要功能,学会调试简单程序,读懂出错信息,并根据出错信息提示排除程序中的错误。

(4) 完成实验报告(实验报告样本见附录 B)。实验报告要求:书写清楚操作步骤、每步操作的结果或观察到的现象,总结实验中的问题并写出实验总结与实验心得。

三、实验范例

【例 1.1】 编程实现在屏幕上显示如下一行文字。

Welcome to the C language world!

在 Visual C++ 6.0 的集成环境下,编辑运行以下源文件。要求调试成功后,将源程序存盘。

源程序文件名为 example1.cpp,代码如下:

```
#include<iostream>
using namespace std;
void main()
{
    printf("Welcome to the C language world!\n");
    cout<<"Welcome to the C language world!"<<endl;
}
```

操作提示

(1) 设置存储位置。在读/写存储设备(如硬盘或可移动存储介质)上建立用于保存实验程序用的专用文件夹 cexer,如图 1.1 所示。

(2) 启动 Visual C++ 6.0。

方法 1: 从开始菜单栏或桌面图标启动,如图 1.2 所示。

方法 2: 在开始/运行对话框输入命令 C:\Program Files\Microsoft Visual Studio\COMMON\MSDev98\Bin\MSDEV.EXE,直接运行,如图 1.3 所示。



图 1.1 在 D 盘建立实验专用文件夹 cexer



图 1.2 从菜单栏启动 Visual C++ 6.0

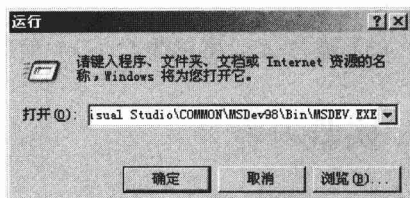


图 1.3 从开始/运行对话框启动 Visual C++ 6.0

说明: 本实验假设 Visual C++ 安装在 C 盘的默认文件夹中,以下所有实验使用相同设置。注意 Visual C++ 6.0 的可执行文件名是 MSDEV.EXE。

(3) 建立实验文件 example1.cpp。从 Visual C++ 开发环境主菜单选择 File(文件)/New(新建)命令,在弹出的对话框中选择 Files (文件)标签页,在其列表项中选择 C++ Source File。然后在右侧的 Location(目录)对话框输入第一步建立的保存实验程序用的专用文件夹 D:\cexer。最后在上方的 File(文件)对话框输入 example1.cpp,注意此时可以不输入扩展名 cpp,程序会自动创建,如图 1.4 所示。

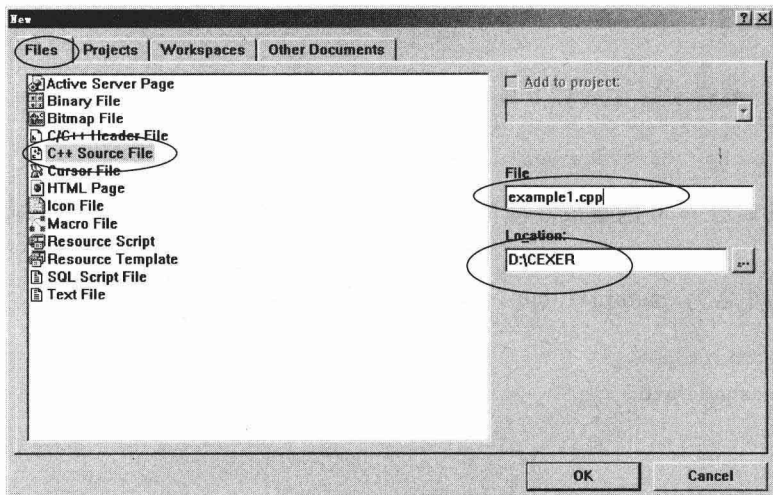


图 1.4 新建 C++ 源文件

说明：单击 OK 按钮后，如果文件 example1.cpp 已经存在，则出现如图 1.5 所示的提示信息，选择 Y(是)，表示覆盖原来的文件；选择 N(否)，则返回图 1.4 所示的对话框重新选择。

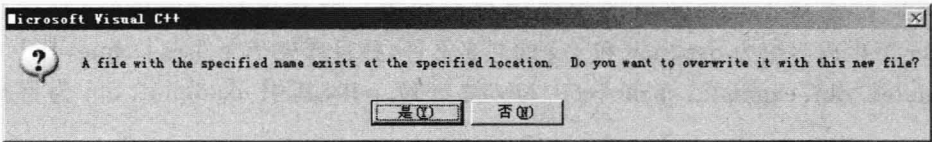


图 1.5 源文件存在提示

(4) 保存源程序。在如图 1.6 所示的代码编辑窗口正确输入本例的源程序；单击工具栏中的保存按钮，或从菜单选择 File(文件)/Save(保存)命令。注意文件名后的 * 表示文件未保存，保存后该标志消失。

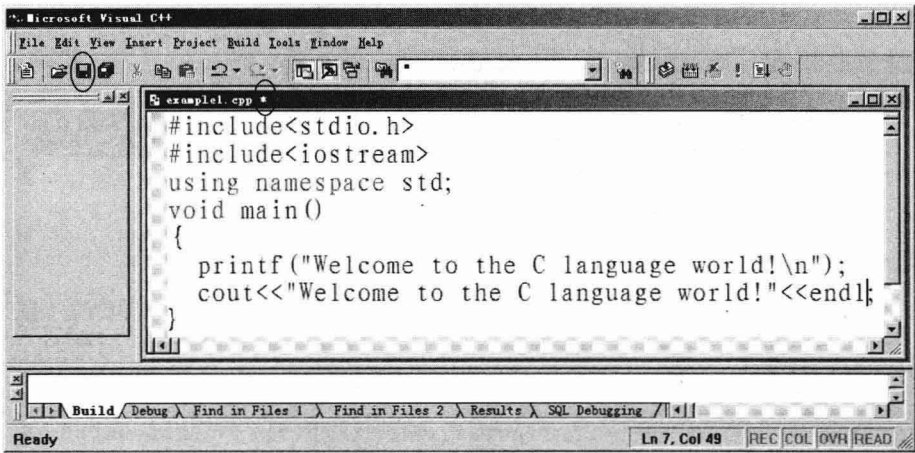


图 1.6 程序编辑区域

说明：如果需要调整编辑区中源程序字体的大小，可以从菜单栏选择 Tools(工具)/Options(选项)/Format(格式)命令，在如图 1.7 所示的对话框中设置字体。

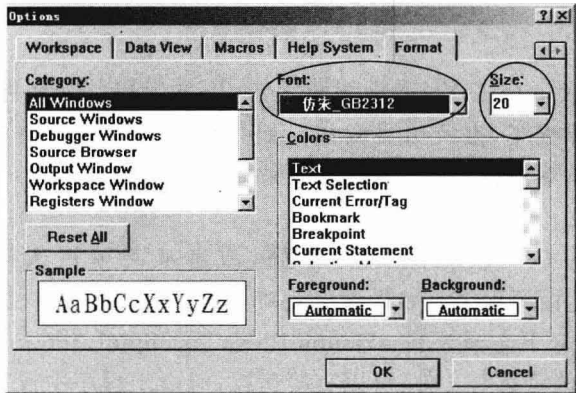


图 1.7 设置字体

(5) 编译源程序。从工具栏选择按钮, 或者从菜单栏选择 Build(组建)/Compile example1.cpp 命令, 编译源程序。如果是首次编译该程序, 则出现如图 1.8 所示的默认工作区创建对话框, 选择 Y(是)创建工作区。如果源程序正确, 则出现如图 1.9 所示的成功编译提示, 同时激活工具栏对应的程序运行按钮。源程序正确编译后, 所在文件夹 D:\cexer 下生成后缀为 dsp、ncb 和 plg 的同名文件, 并且产生子文件夹 Debug, 其下生成文件 example1.obj、example1.pch、vc60.idb 和 vc60.pdb, 其中 example1.obj 为目标代码文件。

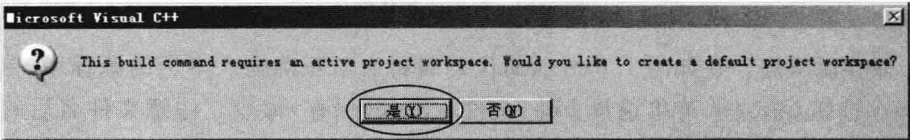


图 1.8 默认工作区创建对话框

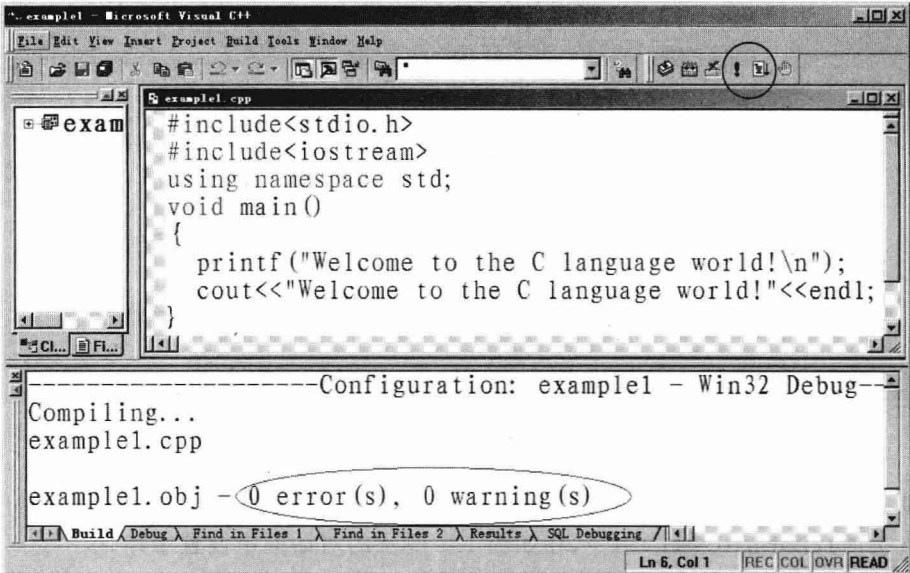


图 1.9 编译源程序

说明: 如果出现“0 error(s)”, 表示编译无错, 正常生成 .obj 文件; 否则程序一定存在语法错误, 必须修改源程序, 重新执行编译。如果出现“0 warning(s)”, 表示程序编译无警告错误, 否则程序虽然可以运行, 但不能确保结果一定是预期的, 取决于产生警告信息的原因。

(6) 连接源程序。从工具栏选择按钮, 或者从菜单栏选择 Build(组建)/Build example1.exe 命令, 连接源程序。如果连接正确, 则出现如图 1.10 所示的成功连接提示。同时在子文件夹 Debug 下生成文件 example1.exe、example1.ilc 和 example1.pdb。其中 example1.exe 是对应的可执行文件, 可以直接运行, 但由于程序中没有等待语句, 所以直接执行时结果一闪即逝。

(7) 运行程序。在工具栏中单击按钮, 或者从菜单栏选择 Build(组建)/Execute

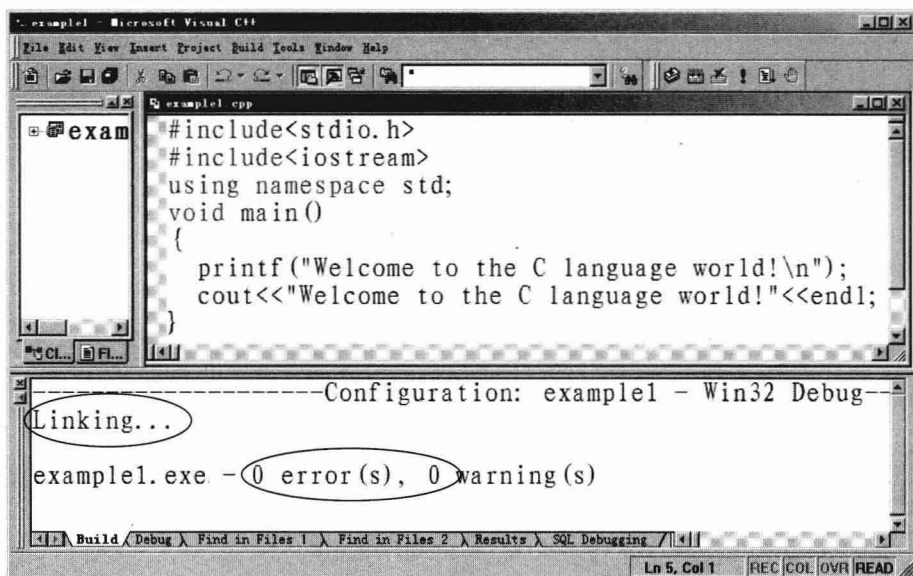


图 1.10 连接源程序

example1.exe 命令(或按快捷键 Ctrl+F5),执行源程序。程序运行后结果如图 1.11 所示。

说明: 图 1.11 的前两行文字是程序 example1.exe 的输出结果,第三行“Press any key to continue”是 Visual C++ 自己附加的操作提示,表示阅读运行结果后按任意键返回到 Visual C++ 编辑界面。

源程序在修改后,也可以不经过(5)编译和(6)连接两步,直接执行第(7)步,此时会弹出提示窗口,如图 1.12 所示。选择 Y(是)会自动执行编译和连接两步,如果没有错误则运行程序。否则会提示错误信息,修改错误后再重新运行程序。

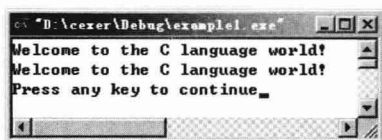


图 1.11 运行源程序的用户窗口

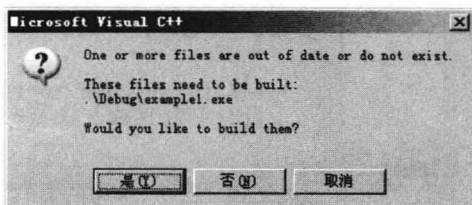


图 1.12 程序修改后执行运行

易错分析

如果源程序有错误,在第(5)步编译时会出现错误提示,以下对几种常见错误进行分析。
注:借助添加错误方式,讲解调试程序方法。

(1) 错误代码“error C2144: syntax error: missing ';' before...”。

① 添加错误: 假如将源程序第 3 行末尾漏掉了分号(;),编译源程序,错误提示如图 1.13 所示。在下方编译信息(Build)窗口显示,错误数目为 2 error(s);出错行号为 example1.cpp(4),括号中的数字表示出错行号;错误代码为 error C2144;错误提示为 syntax error: missing ';' before type 'void'。

② 定位错误: 在编译信息(Build)窗口双击第一个错误信息,则程序编辑窗口的光标会

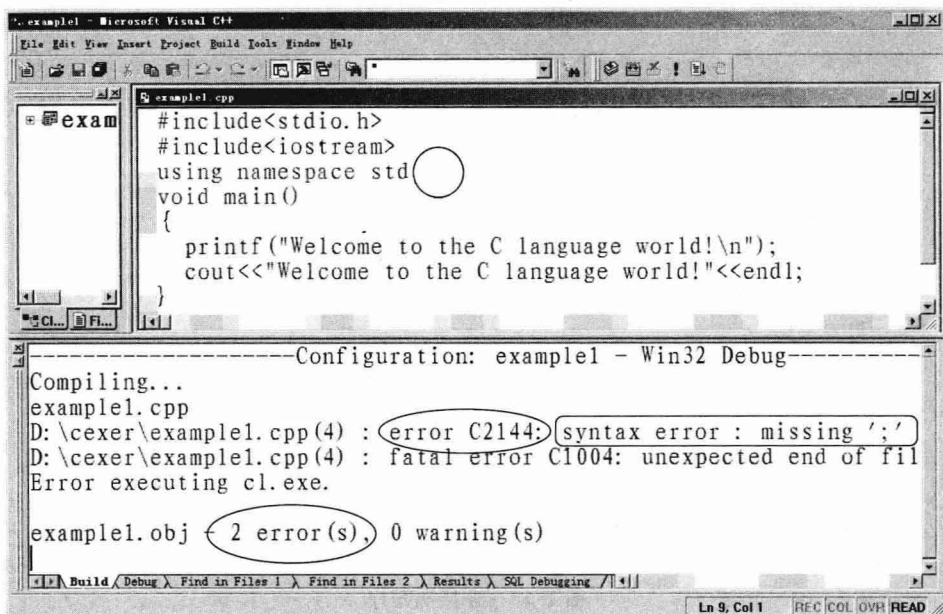


图 1.13 源程序编译后出现错误

自动定位到源程序的第 4 行,如图 1.14 所示。在编译信息(Build)窗口双击不同的错误提示行,则编辑窗口的光标会在不同的源程序错误行间切换。

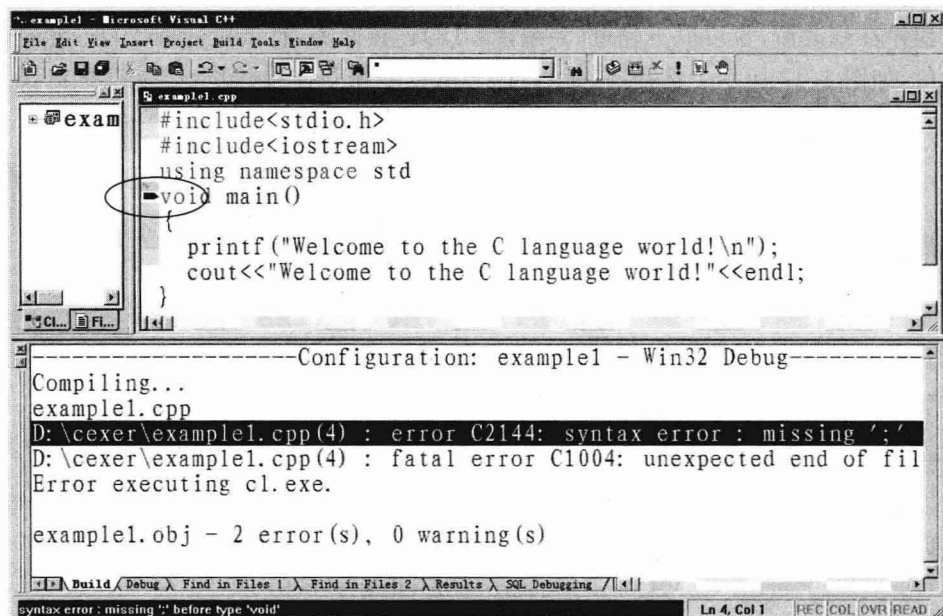


图 1.14 定位程序错误信息

③ 错误分析: 本例中的错误信息解释为“在类型 void 前面缺少 ';'”,虽然提示错误是第 4 行,但是错误实际产生在上一行的末尾,所以,发生错误时,同时也要检查出错提示行的上一行是否有错。

④ 改正错误：修改程序的第 3 行，在行末添加“;”，重新编译源程序。

(2) 错误代码“error C2018: unknown character '0xa3'”。

① 添加错误：假如源程序第 6 行 printf 函数末尾的半角右括号输入的是全角的右括号“)”(可通过输入法切换半角/全角状态)，编译源程序，错误提示如图 1.15 所示。在下方编译信息(Build)窗口显示，错误数目为 3 error(s)；错误信息为 error C2018: unknown character '0xa3'、error C2018: unknown character '0xa9'和 error C2143: syntax error: missing ')’。当然缺少“)”符号同样出现此错误信息。

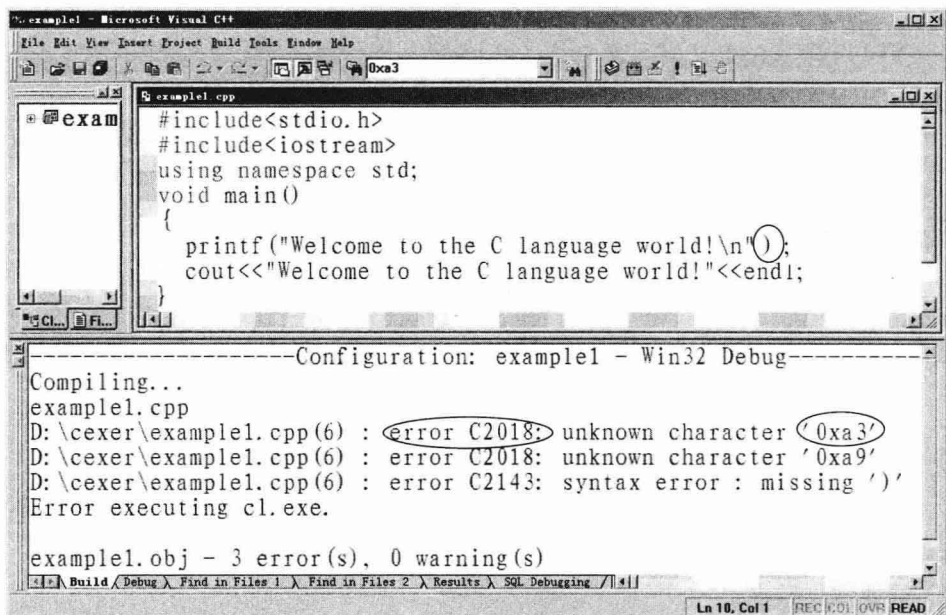


图 1.15 全角字符“)”错误信息

② 错误分析：程序除双引号中原样输出字符外，都是半角字符。如果只看第一个错误信息 error C2018: unknown character '0xa3'，从源程序不容易看出是哪个字符输入错误，如果此时参考相邻的错误信息 error C2143: syntax error: missing ')’，该错误表示语句缺少右括号或右括号误输入为全角字符；如果附近的行没有可以帮助理解的错误提示，则可能表示该行的错误信息过多，需要重新输入该行。提示：该错误经常会在从非文本文件中将代码复制到当前文件中时发生。

③ 改正错误：修改程序中的全角字符右括号为半角字符，重新编译源程序，则程序中的所有错误消失。当程序中出现多个错误时，可能有些错误是相关的，是由同一个原因引起的。因此在改错阶段，不需要将所有错误一次性改完，可以先修改一部分错误，再重新编译，可能会提高修改效率。

(3) 错误代码“error LNK2001: unresolved external symbol _main”。

① 添加错误：假如将源程序第 4 行主函数的名字错误录入成 mian，编译源程序，此阶段没有错误，而在连接源程序时出现错误提示，如图 1.16 所示。错误信息为 error LNK2001: unresolved external symbol _main，表示错误发生在连接阶段，不能找到外部符号 _main，注意 main 前的下划线是编译系统内部表示，实际上的名字是没有下划线的。

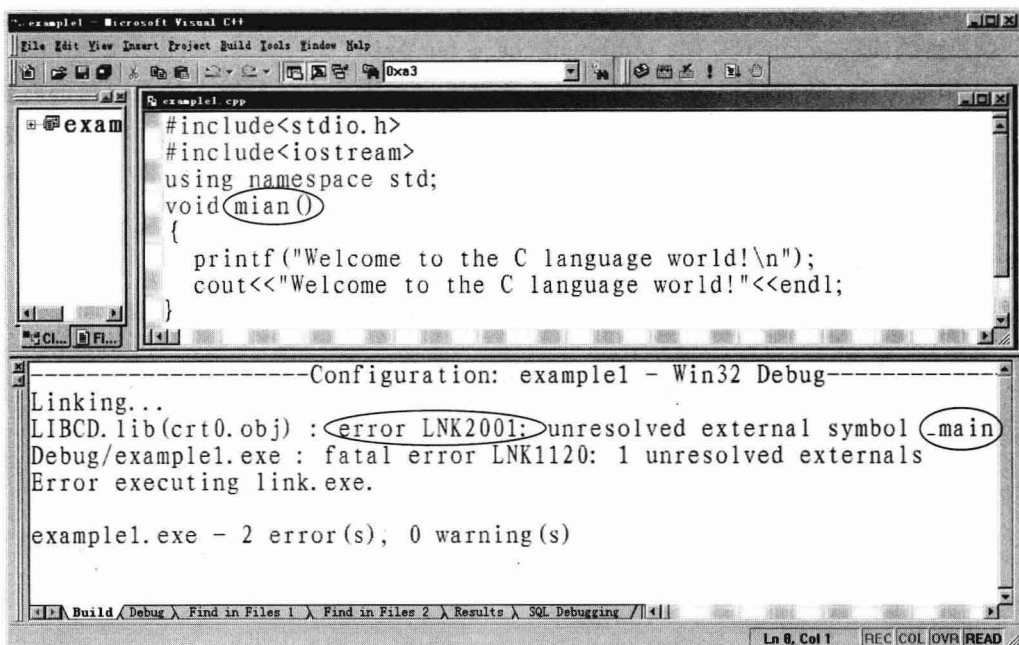


图 1.16 主函数名错误

② 错误分析：由于源程序缺少主函数，所以程序无法运行。

③ 改正错误：修改程序第 4 行中的 mian 为 main，重新编译连接源程序，则程序中的错误消失。

(4) 其他常见错误。

① 假如第一行首部的 # 漏掉，出现部分错误提示如下：

- a. error C2143: syntax error: missing ';' before '<'
- b. error C2501: 'include': missing storage-class or type specifiers
- c. error C2143: syntax error: missing ';' before '<'

分析：通过错误提示无法知道该行是由于缺少 # 而引起的错误，但根据 Visual C++ 环境采用了关键字变色显示技术，比较第 1 行和第 2 行，很容易发现该行的关键字 include 没有变色。

② 假如第 6 行 printf 错误录入成 print，出现错误提示如下：

error C2065: 'print': undeclared identifier

分析：错误信息表示 print 是未定义的标识符，此种类型的错误经常发生，如变量名在定义时和引用时前后不一致，主要是由拼写错误引起的。

③ 假如第 6 行 printf 函数的右双引号“”漏掉，出现错误提示如下：

- a. error C2001: newline in constant
- b. error C2146: syntax error: missing ')' before identifier 'cout'

分析：错误提示并没有直接指出是由于缺少了“”引起的错误，这种错误信息的修改需要改错经验的积累。