



应用技术型高等教育“十三五”规划教材 >>>>

C 语言程序设计 上机指导与习题解答 (第三版)

主 编 ◆ 施俏春 李素若 任正云 赖 玲
副主编 ◆ 琚 辉 胡玉荣 张 牧 肖 衡
主 审 ◆ 田 原



中国水利水电出版社
www.waterpub.com.cn

应用技术型高等教育“十三五”规划教材

C 语言程序设计上机指导与习题解答

(第三版)

主 编 施俏春 李素若 任正云 赖 玲

副主编 琚 辉 胡玉荣 张 牧 肖 衡

主 审 田 原



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书是中国水利水电出版社出版的《C 语言程序设计》(第三版)的配套教材。修订后的内容分五个部分:第一部分是“C 语言上机指导”,介绍了 Visual C++ 6.0 集成环境下的上机方法;第二部分是“上机实验安排”,介绍了程序设计实验的一般步骤,并给出了一份上机报告的样例,精心设计了 11 套上机实验题,每套实验都给出了实验目的、要求和内容,帮助读者在实验中进一步理解所学的知识;第三部分是“主教材习题和参考解答”,对教材中的每道习题都给出了详细的解答,对容易出错和容易混淆的习题还给出了不同的解题方法,通过进一步讲解可以帮助读者提高对 C 语言的基本概念、算法和各种规则的理解;第四部分是“计算机等级考试(二级 C 语言)高频考点分析与上机典型题解”;第五部分“附录”中汇编了 5 套计算机等级考试(二级 C 语言)的笔试试题和参考答案及解析,供读者学习。

本书内容丰富,注重实践;突出重点、分散难点,知识结构合理、覆盖面广,既可作为高校学生学习“C 语言程序设计”课程的辅导教材,也可作为 C 语言自学者的参考教材。

图书在版编目(CIP)数据

C 语言程序设计上机指导与习题解答 / 施俏春等主编

— 3 版. — 北京:中国水利水电出版社,2016.5

应用技术型高等教育“十三五”规划教材

ISBN 978-7-5170-4354-6

I. ①C… II. ①施… 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(2016)第 110342 号

策划编辑:杨庆川 责任编辑:李 炎 加工编辑:韩莹琳 封面设计:李 佳

书 名	应用技术型高等教育“十三五”规划教材 C 语言程序设计上机指导与习题解答(第三版)
作 者	主 编 施俏春 李素若 任正云 赖 玲 副主编 琚 辉 胡玉荣 张 牧 肖 衡 主 审 田 原
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址:www.waterpub.com.cn E-mail: mchannel@263.net(万水) sales@waterpub.com.cn
经 售	电话:(010) 68367658(发行部)、82562819(万水) 北京科水图书销售中心(零售) 电话:(010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京泽宇印刷有限公司
规 格	184mm×260mm 16 开本 22 印张 543 千字
版 次	2009 年 11 月第 1 版 2009 年 11 月第 1 次印刷 2016 年 5 月第 3 版 2016 年 5 月第 1 次印刷
印 数	0001—3000 册
定 价	39.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换
版权所有·侵权必究

前 言

本书出版以来,经过多年的使用,得到了广大教师和读者的肯定,特别是本书的主教材《C 语言程序设计》(任正云、李素若主编,中国水利水电出版社出版)在 2010 年被教育部高等学校高职高专计算机类专业教学指导委员会评为优秀教材,第二版荣获湖北省第七届教学成果奖。根据教育部高等学校计算机基础课程教学指导委员会《高等学校计算机基础核心课程教学实施方案》的要求,针对应用型人才培养的目标,结合主教材《C 语言程序设计》和实际教学需要,并吸取广大读者的宝贵意见,在第二版的基础之上进行了如下几个方面的改进。

(1) 只介绍了 Visual C++ 6.0 集成环境下的上机方法。

(2) 根据主教材《C 语言程序设计》(第三版)编排体系的要求,重新对全书所有的习题给出了详细的解答,对于部分题目除了给出答案外,还给出了分析过程。这部分题目一般都是容易出错和容易混淆的,通过进一步讲解可以帮助读者提高对 C 语言的基本概念、算法和各种规则的理解。

需要特别强调的,本书给出的程序并非唯一正确的解答,对同一个题目可以编出多种程序,我们给出的只是其中的一种,甚至不一定是最佳的一种。对于有些题目,我们给出了两种答案,供读者参考和比较,以启发思路。读者在使用本书时,千万不要照抄照搬,我们只是提供了一种参考答案,读者完全可以编写出更好的程序。

(3) 列举了近几年来全国计算机等级考试(二级 C 语言)高频考点,对高频考点进行了详细分析。

(4) 列举了近几年来全国计算机等级考试(二级 C 语言)上机题题例,对每种类型的题目给出了举例和解答,同时给出了部分上机题习题并附有答案和解析。

(5) 本书附录收录了 2009—2011 年的全国计算机等级考试(二级 C 语言)笔试试题共 5 套,并给出了每套考试试题的解析和答案,供读者参考和学习。

本书由施俏春、李素若、任正云、赖玲任主编,琚辉、胡玉荣、张牧、肖衡任副主编。其中第一、二部分由李素若、赖玲共同编写;第三部分中的第 1~6 章主要由施俏春编写,第 8~11 章由赖玲编写,第 7 章由李素若、琚辉共同编写;第四部分由施俏春编写;第五部分由赖玲编写。全书由施俏春、任正云、李素若共同统稿,胡玉荣、张牧、赖玲负责全书的校对,在本书的编写过程中得到了主审田原教授和相关专家的指导,参与本书大纲讨论和相关内容编写的还有严永松、游明坤、陈万华、李俊梅和李祥琴等,荆楚理工学院对本书的出版给予了极大的支持,在此一并表示衷心的感谢。

由于作者水平有限,书中难免还会存在不足之处,再次恳请广大读者批评指正。

编 者

2016 年 2 月

目 录

前言

第一部分 C 语言上机指导

第 1 章 Visual C++ 6.0 上机操作..... 1	1.5 执行程序..... 3
1.1 使用 VC++ 6.0 调试程序一般步骤..... 1	1.6 关闭程序工作区..... 4
1.2 启动 VC++..... 1	1.7 命令行参数处理..... 5
1.3 新建或打开 C 程序文件..... 2	1.8 程序调试简介..... 5
1.4 程序保存..... 2	

第二部分 上机实验安排

第 1 章 程序设计实验的一般步骤..... 11	2.3 最简单的 C 程序设计实验..... 23
1.1 问题分析..... 11	一、实验目的和要求..... 23
1.2 设计测试计划..... 12	二、实验内容和步骤..... 23
1.3 设计程序方案..... 12	2.4 逻辑结构程序设计实验..... 24
1.4 编写程序..... 13	一、实验目的和要求..... 24
1.5 上机调试..... 14	二、实验内容和步骤..... 24
1.6 实验总结与实验报告内容..... 14	三、上机练习..... 26
1.7 实验报告样例..... 14	2.5 循环结构实验..... 26
一、问题说明..... 15	一、实验目的和要求..... 26
二、测试计划..... 15	二、实验内容和步骤..... 26
三、程序框图..... 15	三、上机练习..... 30
四、源程序..... 15	2.6 数组实验..... 30
五、测试结果..... 16	一、实验目的和要求..... 30
六、本次实验总结..... 16	二、实验内容和步骤..... 30
第 2 章 实验安排..... 18	三、上机练习..... 32
2.1 熟悉 TC 环境下运行 C 程序实验..... 18	2.7 函数实验..... 33
一、实验目的和要求..... 18	一、实验目的和要求..... 33
二、实验内容和步骤..... 18	二、实验内容和步骤..... 33
三、上机需熟悉内容..... 19	三、上机练习..... 36
2.2 数据类型、运算符和表达式实验..... 20	2.8 指针实验..... 36
一、实验目的和要求..... 20	一、实验目的和要求..... 36
二、实验内容和步骤..... 20	二、实验内容和步骤..... 37
三、上机练习..... 22	三、上机练习..... 39

2.9 结构体、共用体实验	40	二、实验内容和步骤	45
一、实验目的和要求	40	三、上机练习	46
二、实验内容和步骤	40	2.11 文件实验	47
三、上机练习	44	一、实验目的和要求	47
2.10 位运算实验	45	二、实验内容和步骤	47
一、实验目的和要求	45	三、上机练习	49

第三部分 主教材习题和参考解答

第1章 C语言程序设计基础	50	五、编程题	99
一、单选题	50	第6章 函数	104
二、填空题	52	一、选择题	104
三、简答题	52	二、填空题	108
四、程序设计部分	53	三、阅读程序，写出执行结果	109
第2章 C语言的数据类型与基本操作	55	四、编程题	113
一、选择题	55	第7章 地址和指针	117
二、填空题	60	一、选择题	117
三、分析下列程序的输出结果	63	二、填空题	119
四、把下列命题写成C语言的表达式	65	三、编程题	119
五、解答题	65	第8章 数组	121
第3章 顺序结构程序设计	67	一、选择题	121
一、选择题	67	二、填空题	125
二、填空题	71	三、编程题	128
三、分析下列程序的运行结果	73	第9章 指针的应用	133
四、编程题	74	一、选择题	133
第4章 选择结构程序设计	76	二、填空题	137
一、选择题	76	三、编程题	140
二、填空题	80	第10章 结构体和共用体	147
三、阅读程序，写出执行结果	82	一、选择题	147
四、编程题	82	二、填空题	149
第5章 循环结构程序设计	89	三、编程题	150
一、选择题	89	第11章 文件	163
二、指出下列程序中的错误	94	一、选择题	163
三、分析下列程序的输出结果	95	二、填空题	164
四、请将下列程序补充完整	97	三、编程题	166

第四部分 计算机等级考试（二级C语言）高频考点分析与上机典型题解

第1章 等级考试高频考点分析	175	1.1 C程序设计基础	175
----------------------	-----	-------------------	-----

1.2 C 语言的基本结构	177	2010 年 3 月全国计算机等级考试二级 C 语言 笔试试题	289
1.3 函数	180	2010 年 9 月全国计算机等级考试二级 C 语言 笔试试题	298
1.4 指针	182	2011 年 3 月全国计算机等级考试二级 C 语言 笔试试题	307
1.5 数组	183	2009 年 3 月全国计算机等级考试二级 C 语言 笔试试题参考答案与解析	316
1.6 字符串	186	2009 年 9 月全国计算机等级考试二级 C 语言 笔试试题参考答案与解析	322
1.7 结构体、共同体和用户定义类型	188	2010 年 3 月全国计算机等级考试二级 C 语言 笔试试题参考答案与解析	327
1.8 文件	190	2010 年 9 月全国计算机等级考试二级 C 语言 笔试试题参考答案与解析	333
第 2 章 上机题例题选讲	192	2011 年 3 月全国计算机等级考试二级 C 语言 笔试试题参考答案与解析	339
2.1 填空题	192	参考文献	345
2.2 改错题	198		
2.3 程序设计题	203		
2.4 上机题练习题	213		
2.5 上机题参考答案	256		
附录	272		
2009 年 3 月全国计算机等级考试二级 C 语言 笔试试题	272		
2009 年 9 月全国计算机等级考试二级 C 语言 笔试试题	280		

第一部分 C 语言上机指导

第 1 章 Visual C++ 6.0 上机操作

C++语言是在 C 语言的基础上发展而来，它增加了面向对象的编程，成为当今最流行的一种程序设计语言。Visual C++ 6.0 是由微软公司开发的面向 Windows 编程的 C++语言工具。它不仅支持 C++语言的编程，也兼容 C 语言的编程。由于 Visual C++ 6.0 被广泛地用于各种编程，使用面很广。这里简要介绍如何在 Visual C++ 6.0 下运行 C 语言程序。

1.1 使用 VC++ 6.0 调试程序一般步骤

在使用 Visual C++ 6.0 调试程序之前，必须先了解项目和项目工作区两个概念。项目 (Project) 是一组相互关联的文件，是将一个应用程序中需要的所有文件组成一个有机的整体，一般包括源文件、头文件和资源文件等。而项目文件必须在某个项目工作区 (Workspace) 的管理之下，一个项目工作区可以包含一个以上的项目。

使用 Visual C++ 6.0 调试程序的大致步骤是：

- (1) 创建一个项目工作区 (Workspace)。
- (2) 在项目工作区中新建一个项目 (Project)。

也可以将上面两步合并，在建立新项目的同时创建项目工作区。

- (3) 在项目中建立源文件和头文件等 (或将这些文件添加到项目中)。
- (4) 编辑项目中的源文件和头文件。
- (5) 连接、编译应用程序。
- (6) 运行应用程序。

注意：如果不先创建项目工作区和项目，而直接创建源文件，当编译该源文件时会提醒没有项目工作区和项目，单击“确定”按钮会自动创建一个默认的项目工作区和项目。

1.2 启动 VC++

Visual C++ 6.0 是一个庞大的语言集成工具，经安装后将占用几百兆磁盘空间。从“开始”→“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0，可启动 Visual C++ 6.0，屏幕上将显示如图 1.1 所示的窗口。

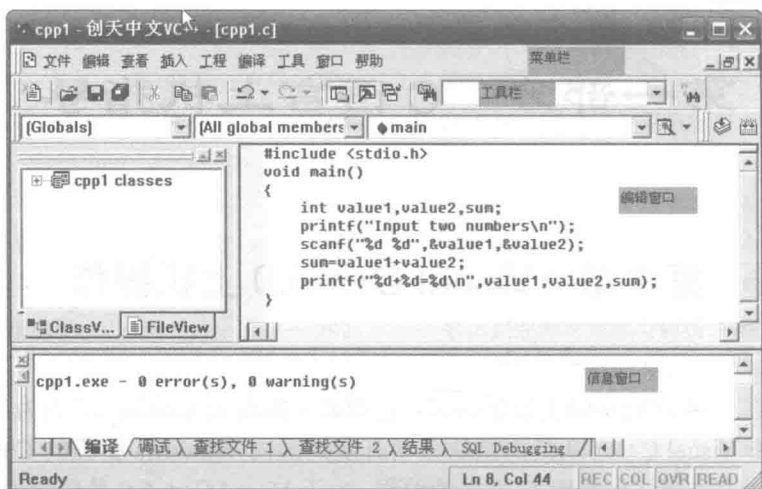


图 1.1 VC++ 6.0 窗口

1.3 新建或打开 C 程序文件

选择“文件”菜单中的“新建”命令，单击如图 1.2 所示的“文件”标签，选中“C++ Source File”，单击“确定”按钮，然后在编辑窗口中输入程序。

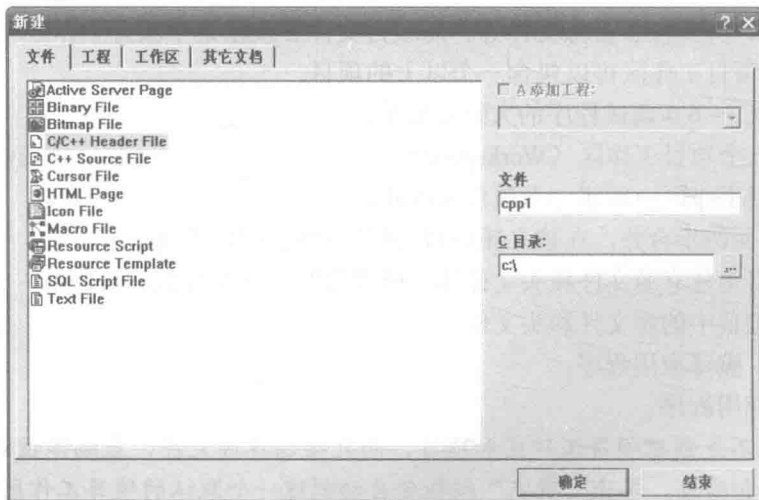


图 1.2 “新建”对话框

如果程序已经输入过，可选择“文件”菜单中的“打开”命令，并在查找范围中找到正确的文件夹，调入指定的程序文件。

1.4 程序保存

在打开的 VC++ 界面上，可直接在编辑窗口输入程序，由于完全是 Windows 界面，输入及修改可借助鼠标和菜单进行，十分方便。当输入结束后，保存文件时，应指定扩展名为“.C”，

否则系统将按 C++扩展名 “.CPP” 保存。程序保存对话框如图 1.3 所示。

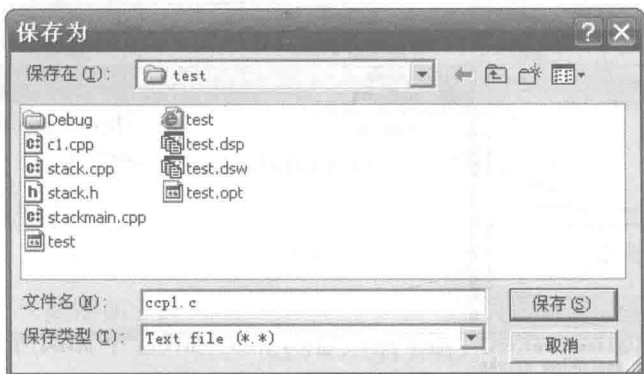


图 1.3 “保存为”对话框

1.5 执行程序

首先要生成可执行文件。使用 VC++ “编译” 菜单中的 “构件” 命令, 如图 1.4 所示, 也可使用快捷键 F7。在编译、链接过程中 VC++ 将保存新输入的程序, 并生成一个同名的工作区。保存文件时需填入文件名, 如 “4-1.C”。假如不指定扩展名 “.C”, VC++ 会把扩展名自动定义为 “.CPP”, 即 C++ 程序。

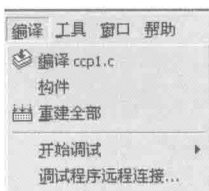


图 1.4 编译菜单

如果程序没有错误, 将在如图 1.5 所示的信息窗口中显示内容:
0 error(s), 0 warning(s)

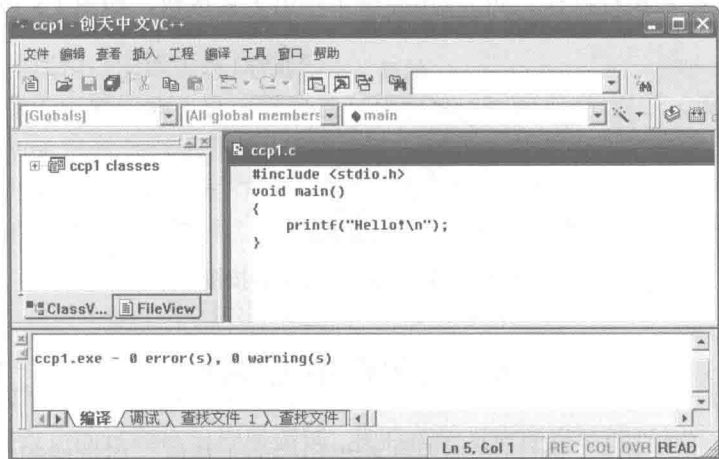


图 1.5 编译链接正确

表示没有任何错误。有时出现几个警告性信息 (warning) 并不影响程序执行。假如有致命性错误 (error), 如图 1.6 所示, 双击某行出错信息, 程序窗口中会指示对应的出错位置, 可根据信息窗口的提示分别予以纠正。然后用 “编译” 菜单中的 “执行” 命令 (或快捷键 Ctrl+F5) 执行程序。

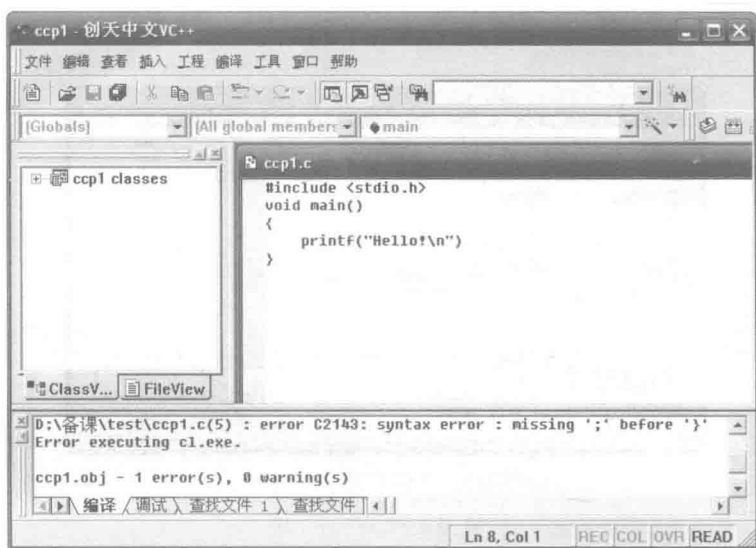


图 1.6 编译链接出错

当运行 C 程序后, VC++ 将自动弹出程序运行窗口, 如图 1.7 所示。按任意键将关闭该窗口。



图 1.7 程序运行窗口

对于编译、链接和执行操作, VC++ 还提供了一组工具按钮, 如图 1.8 所示。

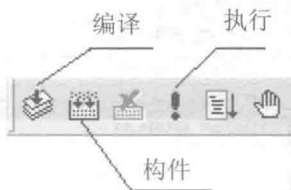


图 1.8 编译连接执行工具按钮

1.6 关闭程序工作区

当一个程序编译链接后, VC++ 系统自动产生相应的工作区, 以完成程序的运行和调试。若想执行第二个程序时, 必须关闭前一个程序的工作区, 然后通过新的编译链接, 产生第二个程序的工作区。否则, 运行的将一直是前一个程序。“文件”菜单提供关闭程序工作区功能, 如图 1.9 所示, 执行“关闭工作区”命令, 在如图 1.10 所示的对话框中单击“否”按钮。如果单击“是”按钮将同时关闭源程序窗口。

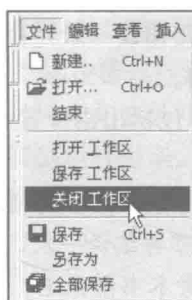


图 1.9 “文件”菜单

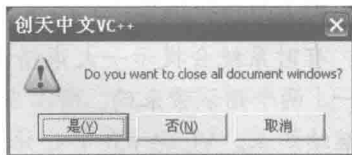


图 1.10 文件关闭对话框

VC++ 6.0 中常用快捷键如下：

- | | |
|-----------------------------------|---|
| Ctrl+N: 新建程序; | Ctrl+O: 打开程序; |
| Ctrl+S: 保存程序; | Ctrl+F7: Compile (编译); |
| F7: Build (编译链接); | Ctrl+F5: Execute Program (执行程序); |
| F5: Go (开始调试); | F9: Insert/Remove BreakPoint (插入/删除断点); |
| F11: Step into (单步调试, 可进入函数体内部); | |
| F10: Step over (单步调试, 不能进入函数体内部); | |
| Ctrl+Break: Stop Build (停止编译链接); | |
| Ctrl+F10: Run to Cursor (运行到光标处); | |
| Shift+F5: Stop Debugging (取消调试)。 | |

1.7 命令行参数处理

VC++是一个基于窗口操作的 C++系统, 没有提供命令行参数的功能, 需要在 Windows 的“MS-DOS 方式”窗口里以命令方式实现。具体步骤参考如下:

- (1) 正确编译连接, 生成可执行程序。
- (2) 通过“我的电脑”或“资源管理器”找到要运行的 C 源程序 (设为 a.c)。
- (3) 进入 debug 文件夹 (它包含 a.c 程序的可执行文件 a.exe)。
- (4) 执行“开始”菜单的“运行”命令, 填入 a, 然后单击“确定”按钮。
- (5) 在打开的“MS-DOS 方式”窗口中输入: a 参数 1 参数 2..., 带参数运行程序。

1.8 程序调试简介

除了较简单的情况之外, 一般的程序都很难一次就能完全正确。在上机过程中, 根据出错现象找出错误并改正的过程称为程序调试。我们在学习程序设计的过程中, 逐步培养调试程序的能力是非常重要的。这是一种经验的积累, 不可能单单凭几句话就可以描述清楚, 要靠读者在上机练习中不断进行摸索、总结。

程序中的错误大致可分为三类:

- (1) 编译错误。编译错误是指程序编译时检查出来的语法错误。编译错误通常是编程人员违反了 C 语言的语法规则, 如大括号不匹配、语句少分号等。

(2) 链接错误。链接错误是指程序链接时出现的错误。链接错误一般是由于未定义或未指明要链接或包含的函数, 或者函数调用不匹配等因素而引起的。

对于编译错误和链接错误, C 语言系统会提供出错信息, 包括出错位置 (行号)、出错提示信息。编程人员可以根据这些信息, 找出错误所在。

注意: 有时系统会提示一大串错误信息, 但并不表示真的有那么多的错误。这往往是因为前面的一、两个错误带来的。所以当纠正了前几个错误后, 可再编译一次, 然后根据最新的出错信息继续纠正。VC++ 6.0 中 C 语言编程的常见错误可参考本书附录 D。

(3) 运行错误。运行错误是指程序执行过程中的错误。有些程序虽然通过了编译链接, 并能够在计算机上运行, 但得到的结果不正确。这类错误相对前两种错误较难改正, 须要求编程人员认真分析程序的执行过程, 从而找出错误所在。

错误的原因一种可能是程序书写错误带来的, 例如应该使用变量 x 的地方写成了变量 y , 虽然没有语法错误, 但意思完全错了; 另一种可能是程序的算法不正确, 解题思路不对。还有一些程序的计算结果有时正确, 有时错误, 这往往是编程时对各种情况考虑不周所致。解决运行错误的首要步骤就是错误定位, 即找出出错的位置, 才能予以纠正。通常先设法确定错误的大致位置, 然后通过调试工具找出真正的错误。

程序调试可通过 Build 菜单下 Start Debug 子菜单进行, 或在工具栏空白处右击再选择快捷菜单中的 Debug 工具条选项, 还可以使用调试微型条辅助进行, 如图 1.11 所示。



图 1.11 调试微型条

以下通过三个例子来讲解如何进行程序调试。

【例 1.1】调试编译错误和链接错误。

下面的程序已经按照要求完成了编辑过程, 命名为: Hello World.c, 以该源文件为例进行编译错误调试。

调试前, 故意把倒数第二行末尾的分号删除后, 再进行编译。如图 1.12 所示, 得到错误提示信息为: “语法错误, 在 ‘}’ 前缺少 ‘;’”。

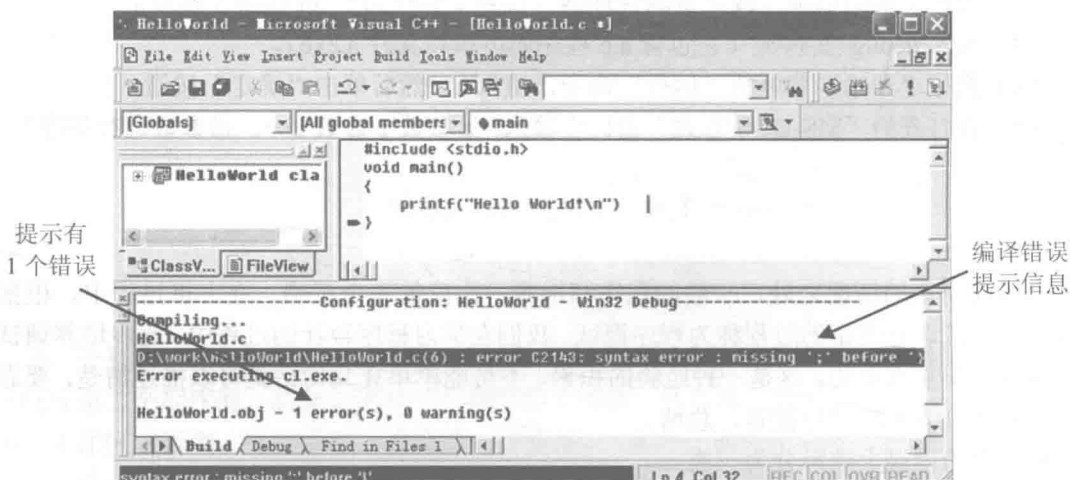


图 1.12 编译错误提示窗口

此时, 可在错误信息上双击, VC++ 6.0 将在输出窗口高亮显示该行提示信息, 并切换到

出错的源文件编辑窗口。可以看到，在编辑窗口左侧的蓝色箭头指向了错误所在行。根据错误提示信息，把“;”加在错误行前一行的末尾，修改后再进行调试。

注意：根据错误信息直接修改错误是改正编译错误和链接错误的通用方法。

【例 1.2】调试运行错误。

程序的源文件为：

```
#include <stdio.h>
void main()
{
    int a, b, c;           //定义 3 个整型变量 a, b, c
    a = 3;                 //把数值 3 存入变量 a 中
    b = 2;                 //把数值 2 存入变量 b 中
    c = a + b;             //把 a+b 的结果存入变量 c 中
    printf("%d + %d = %d\n", a, b, c); //在屏幕上输出 c 的值，屏幕显示为：3+2=5
}
```

以该源文件为例进行运行错误调试。

调试前，故意把程序倒数第三行的“+”号改成“-”号后，再进行编译。编译和链接均通过，但运行结果为 1，而不是 5。因此，初步估计错误发生在倒数第三行 $c=a-b$ 。下面通过调试找出真正的错误。

① 在该行位置上单击，选择图 1.11 所示调试微型条的  按钮，设置一个断点。此时，该行前面出现一个红色圆点标志。也可以先将鼠标定位在此行，再选择调试微型条的  按钮也可以达到同样效果。

注意：断点通常用于调试较长的程序，且程序可同时设置多个断点；而  按钮的功能是程序运行到光标处暂停。

② 选择调试微型条的  按钮开始进行调试。当程序运行到该行时就会暂停，如图 1.13 所示，编辑窗口中左侧的彩色箭头表示当前程序暂停的位置。

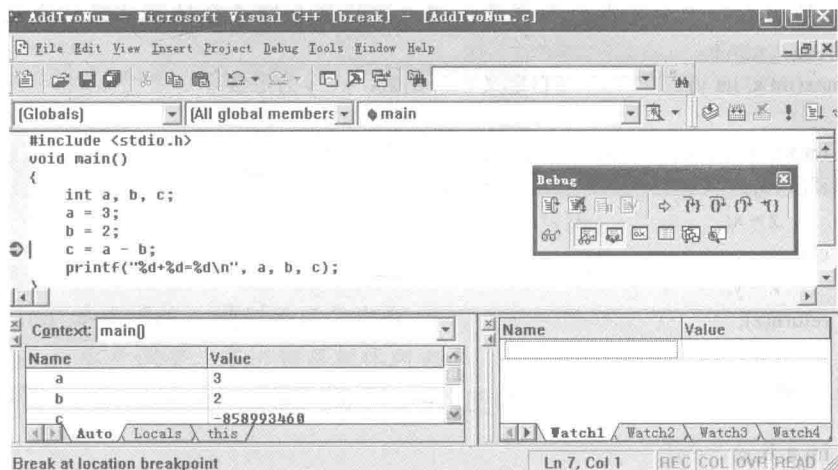
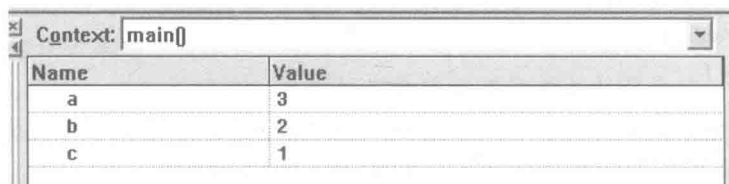


图 1.13 程序在断点处暂停

此时，在图 1.13 的左下角窗口中系统自动显示了有关变量的值，其中 a 和 b 的值分别是 3、2，但变量 c 的值是任意值。这是因为程序刚运行到此处，并未执行 $c=a-b$ 语句，因而还未对

变量 `c` 赋值。

③ 选择调试微型条的  按钮进行单步调试。此时，箭头下移一行。如图 1.14 所示，左下角窗口中 `c` 的值被更新为 1（注意：变量的值更新后用红色表示）。真正的错误就是发生在这一行。因为程序需要完成的是 $3+2$ 的运算，而非 $3-2$ 的运算。



Name	Value
a	3
b	2
c	1

图 1.14 变量观察窗口

④ 找到真正的错误后，单击调试微型条上的  按钮结束调试，同时返回到程序编辑窗口进行修改。

如程序仍需再次调试，可重复以上步骤。

注意：使用断点可以使程序暂停。一旦设置了断点，无论是否还需要调试，每次执行程序时都会在断点上暂停。因此调试结束后应取消所定义的断点。方法是先把光标定位在断点所在行，再单击“调试微型条”中的  按钮。该按钮是一个开关，第一次单击是设置断点，第二次单击是取消断点。如果想取消全部断点，可单击“Edit”菜单中的“Breakpoints”菜单项，屏幕上会显示“Breakpoints”窗口。窗口下方会列出程序中设置的所有断点，单击“Remove All”按钮，将取消所有断点。

如果一个程序设置了多个断点，按一次快捷键 `Ctrl+F5` 会暂停在第一个断点，再按一次快捷键 `Ctrl+F5` 会继续执行到第二个断点暂停，依次执行下去。

【例 1.3】 函数跟踪调试。输入两个数，输出其中的较大值。

为讲解此部分，给出下面的例子，程序命名为 `max.c`。

注意：函数相关知识点将在后面章节介绍，此处仅介绍函数的调试跟踪方法。

```
#include <stdio.h>
int max(int x, int y)           //自定义 max 函数，x、y 为形参
{
    int z;
    if (x > y)
        z = x;
    else
        z = y;
    return(z);
}
void main( )
{
    int a, b, c;
    scanf("%d, %d", &a, &b);
    c = max(a, b);              //调用 max 函数，求 a 和 b 中大数，其中 a 和 b 为实参
    printf("max = %d\n", c);
}
```

输入及程序运行结果如下:

```
2,3↵  
max = 3
```

① 单击主函数的 `scanf` 语句所在行,再单击调试微型条的  按钮。程序运行到此处会暂停。

② 单击调试微型条的  按钮,进行单步调试,程序执行“`scanf("%d,%d",&a,&b);`”语句。此时,需要在已经打开的控制台窗口中输入变量 `a,b` 的值,如输入 `2,3↵`。输入完成后,编辑窗口中光标已经下移一行,指向“`c = max(a,b);`”语句。

③ 由于函数调用的实质是实参传递给形参,因此需要跟踪进入到函数体内部进行检查。此时,单击调试微型条的  按钮进入 `max` 函数,如图 1.15 所示,光标进入 `max` 函数体。同时,在左下角窗口中可看到形参 `x、y` 接收了实参 `a、b` 传递过来的值,因此 `x=2,y=3`。

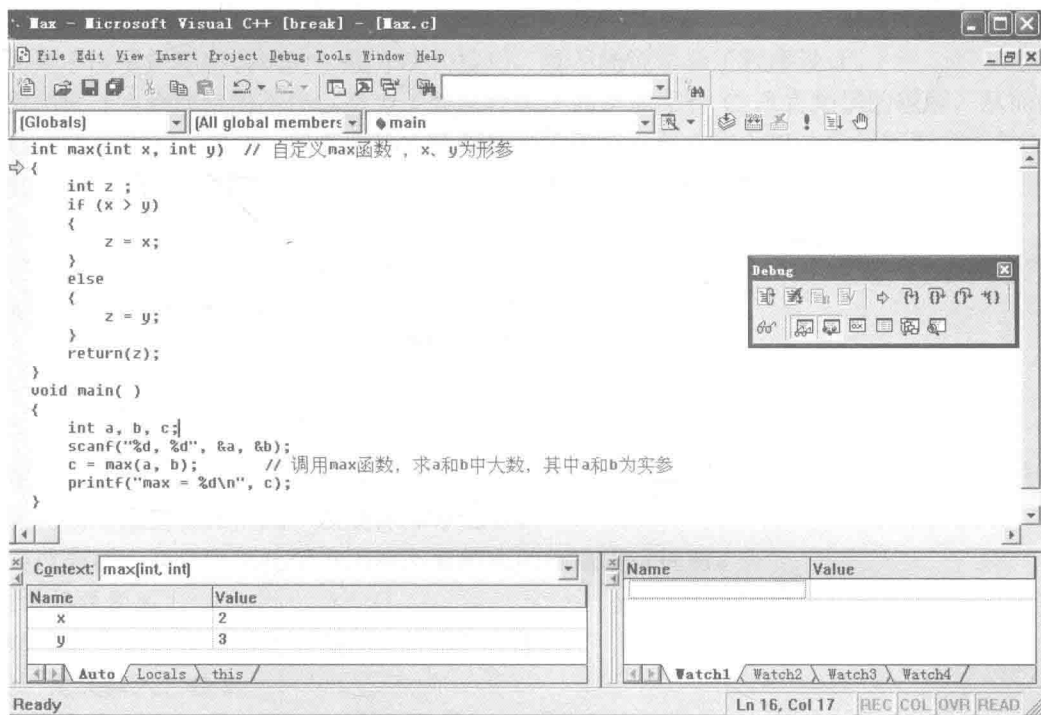


图 1.15 跟踪进入函数体内部

注意: 两个单步调试按钮的区别。

 `step into` 单步调试, 进到函数体内部。

 `step over` 单步调试, 不进到函数体内部。

④ 单击  按钮,继续单步调试。由于 `x < y`, 所以程序运行到双分支 `if` 语句结构时将执行 `else` 后的语句。

⑤ 单击  按钮,程序执行“`z = y`”语句。此时, `z` 被赋值为 3。

⑥ 单击  按钮,程序执行“`return(z)`”语句,把 `z` 的值返回给调用函数,即 `main` 函数。

⑦ 单击  按钮(或单击  按钮,跳出 `max` 函数体),程序已返回到 `main` 函数“`c = max(a, b);`”语句行。

⑧ 再次单击  按钮, 如图 1.16 所示, c 的值被更新为 3, 即 max 函数返回值。

Name	Value
a	2
b	3
c	3

图 1.16 函数调用后的变量观察窗口

⑨ 单击  按钮, 程序执行 “printf(“max = %d\n”, c);” 语句, 控制台窗口显示 “max=3”。

⑩ 单击调试微型条上的  按钮, 结束调试。