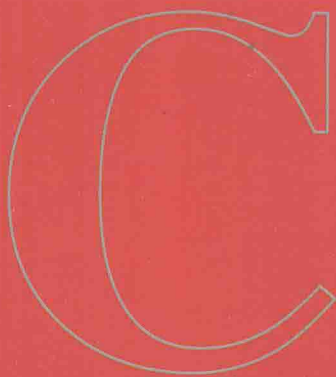


21世纪高等学校计算机**基础**实用规划教材

计算机程序设计 实践教程 —— C语言

童键 刘卫国 主编



清华大学出版社

21世纪高等学校计算机**基础**实用规划教材

计算机程序设计 实践教程 —— C语言

童键 刘卫国 主编

清华大学出版社
北京

内 容 简 介

本书是与《计算机程序设计基础教程——C语言》配套的教学参考书。全书包含 Visual C++ 6.0 集成开发环境、实验指导、常用算法设计、习题选解和模拟试题 5 个部分。前两部分帮助读者熟悉上机环境,方便读者上机操作,让读者通过上机实验更好地掌握程序设计的基本思想和方法。常用算法设计部分针对初学者学习程序设计的难点,总结了常见问题的编程思路,帮助读者提高程序设计能力。习题选解部分帮助读者通过习题练习,复习和掌握课程内容,达到巩固和提高的目的。模拟试题部分参考全国计算机等级考试的基本要求和考试题型,提供了两套笔试模拟试题和两套机试模拟试题,帮助读者检验学习效果。

本书内容丰富,实用性强,适合作为高等院校计算机程序设计课程的教学用书,也可供参加各类计算机等级考试的读者与社会各类计算机应用人员阅读参考。

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

计算机程序设计实践教程: C 语言/童键等主编. —北京:清华大学出版社,2014

21 世纪高等学校计算机基础实用规划教材

ISBN 978-7-302-37026-0

I. ①计… II. ①童… III. ①C 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 143059 号

责任编辑:魏江江 薛 阳

封面设计:常雪影

责任校对:梁 毅

责任印制:王静怡

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 刷 者:北京四季青印刷厂

装 订 者:三河市漂源装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:15.5

字 数:390 千字

版 次:2014 年 12 月第 1 版

印 次:2014 年 12 月第 1 次印刷

印 数:1~2000

定 价:29.00 元

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和教学方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程(简称‘质量工程’)”,通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机公共课程领域,以公共基础课为主、专业基础课为辅,横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 面向多层次、多学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映各层次对基本理论和原理的需求,同时加强实践和应用环节。

(2) 反映教学需要,促进教学发展。教材要适应多样化的教学需要,正确把握教学内容和课程体系的改革方向,在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生的知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材配套,同一门课程可以有针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材、教学参考书,文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优选用。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主题。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机基础实用规划教材

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前言

计算机程序设计最能够体现问题求解方法,是理解计算机工作过程的有效途径,也是计算思维能力培养的重要载体。因此,计算机程序设计课程的重要性不仅体现在一般意义上的程序设计能力的培养,而且体现在引导学生实现问题求解的思维方式的转换,即学生计算思维能力的培养。在这个过程中,上机实践是十分重要的环节。

C语言是常用的程序实现工具之一,掌握其语法规则不应死记硬背,可以通过对比分析、上机练习来加强理解和记忆,而程序设计能力更要通过大量的上机实践来培养。许多程序设计方法不是光靠听课和看书就能学到的,而是通过大量的上机实践积累起来的,所以,学习程序设计不能仅限于纸上谈兵,而必须以实践为重。本书是与《计算机程序设计基础教程——C语言》配套的教学参考书。全书包含以下五个方面的内容。

一是 Visual C++ 6.0 集成开发环境。要上机运行一个 C 语言程序,需要 C 语言编译系统的支持。C 语言编译系统有很多,其中具有较好应用背景的 Visual C++ 6.0 也支持 C 语言程序的编译。在这部分介绍 Visual C++ 6.0 集成开发环境的使用方法,这是上机操作的基础。

二是实验指导。为了方便读者上机操作,在这一部分设计了 14 个实验,每个实验都和课程学习内容相配合,以帮助读者通过上机实验加深对课程内容的理解,更好地掌握程序设计的基本思想和方法。实验内容以编写程序练习为主,分为“模仿编程实验”和“独立编程实验”。“模仿编程实验”给出了程序的主体部分,要求将程序补充完整,“独立编程实验”则要求读者独立完成编程练习。

三是常用算法设计。如何编写程序是学习程序设计的难点,也是学习的重点。这部分内容根据程序设计教学基本要求,将常见的程序设计问题进行分类,分别总结每一类问题的算法设计思路,以引导读者掌握基本的程序设计方法和技巧。教学实践表明,这样做对提高初学者的程序设计能力很有帮助。

四是习题选解。这部分以课程学习为线索,编写了十分丰富的习题并给出了参考答案。在使用这些题解时,应重点理解和掌握与题目相关的知识点,而不要死记答案,应在阅读教材的基础上再来做题,通过做题达到强化、巩固和提高的目的。

五是模拟试题。参考全国计算机等级考试对 C 语言程序设计的基本要求和考试题型,提供了两套笔试模拟试题和两套机试模拟试题,旨在帮助读者检验学习效果。提醒读者注意的是,全国计算机等级考试中“计算机基础知识”部分的内容不是本课程的教学内容,需要读者另外阅读相关文献资料。此外,读者还要熟悉全国计算机等级考试的无纸化考试方式。

本书第1章由舒卫真编写,第2章和第3章由刘卫国编写,第4章和第5章由童键编写,全书由童键、刘卫国担任主编。此外,参与讨论与部分编写工作的还有蔡旭晖、刘胤宏、文碧望、石玉、欧鹏杰、胡勇刚、刘苏洲、孙士闯、周克涛等。清华大学出版社的编辑对本书的策划、出版做了大量工作,在此表示衷心的感谢。

由于编者水平有限,书中难免存在不足之处,恳请广大读者批评指正。

编者

2014年4月

目 录

第 1 章 Visual C++ 6.0 集成开发环境	1
1.1 Visual C++ 6.0 的安装与启动	1
1.2 Visual C++ 6.0 主窗口的组成	2
1.3 Visual C++ 6.0 环境下 C 程序的运行	4
1.4 Visual C++ 6.0 程序调试	6
第 2 章 实验指导	8
实验 1 程序的运行环境和方法	8
实验 2 程序的数据描述	9
实验 3 顺序结构程序设计	11
实验 4 选择结构程序设计	13
实验 5 循环结构程序设计	15
实验 6 常用算法	17
实验 7 函数和编译预处理	20
实验 8 数组	24
实验 9 指针	27
实验 10 结构体	30
实验 11 链表	34
实验 12 共用体和枚举	37
实验 13 文件	40
实验 14 综合程序设计	44
第 3 章 常用算法设计	54
3.1 累加与累乘问题	54
3.2 数字问题	59
3.3 数值计算问题	61
3.4 数组的应用	64
3.5 函数的应用	67
3.6 解不定方程	69
思考题	71
第 4 章 习题选解	74
习题 1 程序设计概述	74

习题 2 程序的数据描述	76
习题 3 顺序结构程序设计	81
习题 4 选择结构程序设计	86
习题 5 循环结构程序设计	97
习题 6 函数与编译预处理	117
习题 7 数组	132
习题 8 指针	155
习题 9 结构体	171
习题 10 共用体与枚举	187
习题 11 文件操作	200

第 5 章 模拟试题..... 213

笔试模拟试题 1	213
笔试模拟试题 2	224
上机模拟试题 1	235
上机模拟试题 2	237

参考文献..... 240

第 1 章

Visual C++ 6.0 集成开发环境

运行 C 语言程序需要相应编译系统的支持。C 语言的编译系统有很多,常用的都是集成开发环境,即源程序的输入、修改、调试及运行都可以在同一环境下完成,功能齐全,操作方便。Visual C++ 是 Microsoft 公司开发的面向 Windows 编程的 C++ 语言工具,也兼容 C 语言的编程。Visual C++ 6.0(简称 VC 6.0)是目前常用的版本,被广泛地应用于实际软件开发。VC 6.0 包含的内容十分丰富,本节只介绍一些常用的与 C 语言程序设计相关的操作,以方便读者在 VC 6.0 环境下编写 C 语言程序。

1.1 Visual C++ 6.0 的安装与启动

在启动 VC 6.0 之前,首先要安装 VC 6.0。它既可以单独安装,也可以随 Visual Studio 一起安装。VC 6.0 的安装方法和其他 Windows 应用程序的安装方法类似。将 VC 6.0 系统安装盘放入光驱,一般情况下系统能自动运行安装程序,否则运行安装盘中的 setup.exe 文件。启动安装程序后,根据屏幕提示依次回答有关内容,便可完成系统安装。

启动 VC 6.0 的过程十分简单。常用的方法是:在 Windows 桌面选择“开始”|“所有程序”|Microsoft Visual Studio 6.0|Microsoft Visual C++ 6.0 命令,即可启动 VC 6.0,屏幕上将显示如图 1-1 所示的 VC 6.0 主窗口。

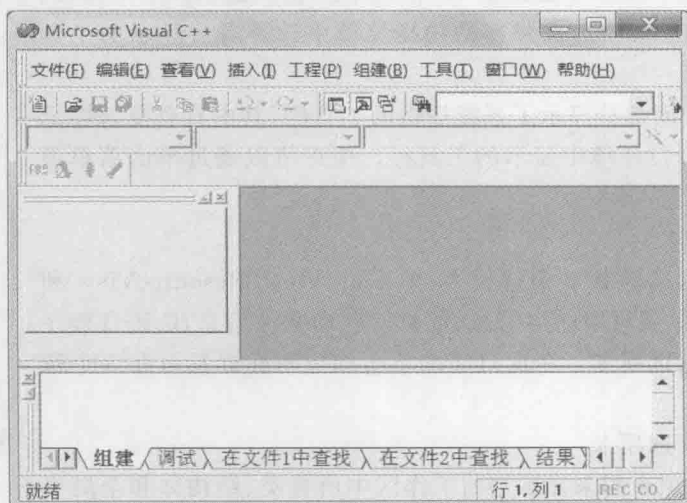


图 1-1 VC 6.0 主窗口

也可以在安装完成后,在 Windows 桌面建立 VC 6.0 的快捷方式图标,这样双击快捷方式图标就能进入 VC 6.0 主窗口。

1.2 Visual C++ 6.0 主窗口的组成

和其他 Windows 应用程序一样,VC 6.0 主窗口也具有标题栏、菜单栏和工具栏。标题栏的内容是 Microsoft Visual C++。菜单栏提供了编辑、运行和调试 C/C++ 程序所需要的命令。工具栏是一些命令的快捷按钮,单击工具栏上的按钮,即可执行该按钮所代表的操作。

在 VC 6.0 主窗口的左侧是工作区(Workspace)窗口,右侧是程序编辑窗口,下方是输出(Output)窗口。工作区窗口用于显示所设置的工作区的信息,程序编辑窗口用于输入和修改源程序,输出窗口用于显示程序编译、运行和调试过程中出现的状态信息。

1. 菜单栏

VC 6.0 的菜单栏共有 9 个菜单项:文件(File)、编辑(Edit)、查看(View)、插入(Insert)、工程(Project)、组建(Build)、工具(Tools)、窗口(Window)和帮助(Help),每个菜单项都有下拉菜单,下拉菜单中的每个命令执行不同的功能。

“文件”菜单项包含用于对文件进行各种操作的命令,“编辑”菜单项包含所有与文件编辑操作有关的命令,“查看”菜单项包含用于检查源代码和调试信息的各种命令,“插入”菜单项包含用于向当前项目中插入新类、新资源等的命令,“工程”菜单项包含用于管理项目和工作区的一系列命令,“组建”菜单项包含用于编译、创建、调试及执行应用程序的命令,“工具”菜单项中包含 VC 6.0 提供的各种工具,“窗口”菜单项用于设置 VC 6.0 集成开发环境中窗口的属性,“帮助”菜单项提供了详细的帮助信息。

2. 工具栏

在默认情况下,VC 6.0 提供了 11 个工具栏,但只显示 3 个工具栏。用户可以通过右击工具栏,在弹出的快捷菜单中选择需要显示的工具栏,如图 1-2 所示。

在图 1-2 的中间部分显示了系统提供的工具栏,其中具有复选标记的菜单项表示在开发环境中显示的工具栏。用户可以通过单击菜单项来控制工具栏是否显示。

3. 工作区窗口

工作区窗口通常包括 3 个选项卡,即 ClassView、ResourceView 和 FileView,分别显示项目中的类信息、资源信息和文件信息,C 语言程序一般没有资源信息选项卡。在窗口底端单击相应图标标签可在 3 个选项卡之间切换。

1) ClassView 选项卡

ClassView 选项卡用来显示当前工作区中所有类、结构体和全局变量,如图 1-3 所示。

ClassView 选项卡提供了项目中所有类、结构体和全局变量的层次列表,通过单击列表

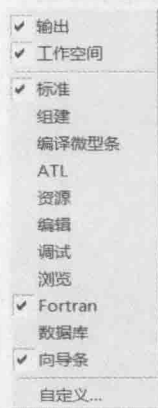


图 1-2 工具栏快捷菜单

左侧小的加号(+)或减号(-)图标可以扩展或折叠列表。双击列表开头靠近文件夹或书本形状图标的文字也可以扩展或折叠列表。

在层次列表的每个项目前面都有一个特殊的图标。例如,私有函数的图标是一个紫色菱形框,全局变量是一个青绿色图标。当用户双击某函数名时,在程序编辑窗口将打开定义该函数的源代码。

用户在某一个列表项目名上右击时,将弹出一个快捷菜单,从中可以选择要执行的命令。右击的项目名不同,快捷菜单中的命令也就不同。

2) ResourceView 选项卡

ResourceView 选项卡在层次列表中列出了项目中用到的所有资源。任何图像、字符串以及程序所需要的其他编程部件都可以作为资源使用。C 语言程序用不到此项。

3) FileView 选项卡

FileView 选项卡用于管理项目中使用的文件。它根据文件类型的不同,将其放置在不同的结点下。例如,源文件被放置在 Source Files 结点下,头文件(.h)被放置在 Header Files 结点下,如图 1-4 所示。

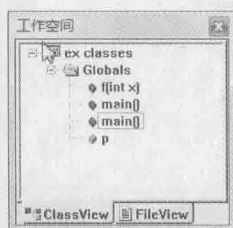


图 1-3 ClassView 选项卡

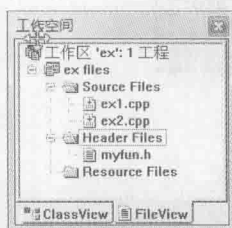


图 1-4 FileView 选项卡

在 FileView 选项卡中,用户不仅可以把文件从一个文件夹移动到另一个文件夹中,也可以创建保存特定类型文件(根据其扩展名)的新文件夹。创建新文件夹的方法是:右击要添加新文件夹的文件夹或项目,然后在快捷菜单中选择“新建文件夹”命令,系统将显示“新建文件夹”对话框,输入文件夹的名称以及相应的文件扩展名,然后单击“确定”按钮完成创建过程。双击在 FileView 选项卡中显示的文件名,即可编辑该文件。

4. 程序编辑窗口

VC 6.0 提供的程序编辑窗口是一个功能齐全的文本编辑器,可用于编辑 C/C++ 头文件、C/C++ 程序文件、Text 文本文件和 HTML 文件等。当打开或建立上述类型的文件时,该编辑器将自动打开。VC 6.0 编辑器除了具有复制、查找、替换等一般文本编辑器的功能外,还具有很多特色功能,如根据 C/C++ 语法将不同元素按照不同颜色显示、根据合适长度自动缩进等。

文本编辑器还具备自动提示的功能。当用户输入程序代码时,文本编辑器会显示对应的成员函数和变量,用户可以在成员列表中选择需要的成员,这样既可以减少输入工作量,又可以避免手动输入错误。

5. 输出窗口

输出窗口主要用于显示编译、调试结果以及文件的查找信息等。

1.3 Visual C++ 6.0 环境下 C 程序的运行

1. 新建或打开 C 程序文件

在 VC 6.0 主窗口中选择“文件”|“新建”命令,这时屏幕出现一个“新建”对话框,如图 1-5 所示。单击该对话框中的“文件”标签,选中 C++ Source File 选项,表示要建立新的源程序文件。在对话框右半部分的“位置”文本框中输入源程序文件的存储路径(如 e:\cp),在“文件名”文本框中输入源程序文件的名称(如 test.c)。

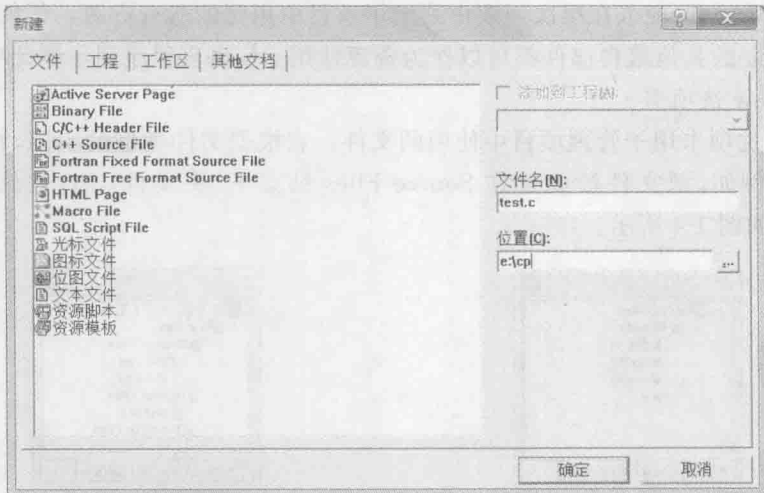


图 1-5 “新建”对话框

注意:

(1) 源程序文件的存储路径一定要事先建好。即首先要在 e 盘建立文件夹 cp,然后才能输入上面的存储路径 e:\cp。

(2) 输入文件名时,如果不指定扩展名.c,系统将按 C++ 源程序文件的扩展名.cpp 保存。因为 C++ 基本兼容 C,所以用 C 语言编写的程序一般可以直接当作 C++ 程序来运行。但在学习 C 语言的时候建议选用.c 文件,主张在 C++ 的开发环境下编写纯 C 语言程序。

在单击“确定”按钮后,回到 VC 6.0 主窗口,可以在编辑窗口中输入或修改源程序。由于完全是 Windows 界面,可以使用鼠标操作,因此输入和修改都十分方便。

如果源程序文件已经存在,可选择“文件”|“打开”命令,并在查找范围下拉列表框中找到正确的文件路径,打开指定的程序文件。

文件修改后要进行存盘操作。

2. 程序的编译

在主窗口中选择“组建”|“编译[test.c]”命令,如果是首次编译,则屏幕出现一个提示框,编译命令要求建立一个工程工作区,询问用户是否同意建立一个默认的工程工作区,单击“是”按钮,表示同意由系统建立默认的工程工作区,然后开始编译。

也可以不用菜单操作,而直接按 Ctrl+F7 组合键来完成编译。

在编译过程中,编译系统检查源程序中是否有语法错误,然后在输出窗口显示编译信息。如果程序没有语法错误,则生成目标文件 test.obj,并将在输出窗口中显示,如图 1-6 所示。其中“0 error(s), 0 warning(s)”表示没有任何错误。

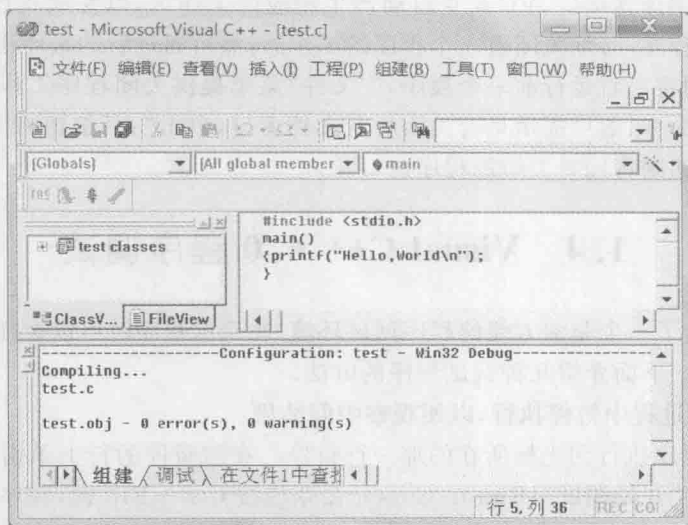


图 1-6 编译正确提示信息

出现几个警告信息(warning)并不影响程序执行。假如有错误信息(error),则会指出错误的位置和信息,双击某行出错信息,程序窗口中会指示对应出错位置,根据信息窗口的提示分别予以修改。为了“造”一个错误,将上面 test.c 中 printf 函数调用语句后面的分号去掉,重新编译,错误提示如图 1-7 所示。要根据错误信息分析错误原因并找到错误位置,对源程序进行修改。

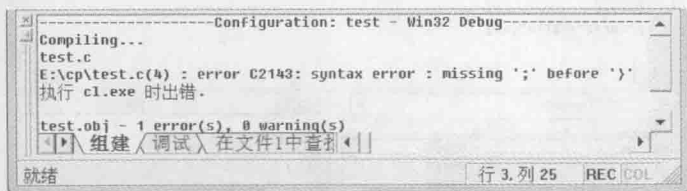


图 1-7 编译出错提示信息

3. 程序的连接

在生成目标程序后,还要把程序和系统提供的资源(如库函数、头文件等)连接起来,生成可执行文件后才能运行。此时在主窗口的菜单栏中选择“组建”|“组建[test.exe]”命令,表示要求连接并生成一个可执行文件 test.exe。同样,在输出窗口会显示连接信息,如果有错误,则要返回修改源程序。

以上介绍的是分别进行程序的编译和连接,也可以在主窗口的菜单栏中选择“组建”|“组建”命令(或按 F7 键)一次完成编译与连接。

4. 程序的执行

在生成可执行文件后,就可以执行程序了。在主窗口的菜单栏中选择“组建”|“! 执行

[test.exe]”命令(或按 Ctrl+F5 键)执行程序。当程序执行后,VC 6.0 将自动弹出数据输入输出窗口,在其中实现数据输入输出,按任意键将关闭该窗口。

5. 关闭程序工作区

当一个程序编译连接后,VC 6.0 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,必须关闭前一个程序的工作区,然后通过新的编译连接产生第二个程序的工作区,否则将一直运行前一个程序。“文件”菜单提供关闭程序工作区的功能。选择“关闭工作空间”命令,然后提示是否关闭所有文档窗口,单击“否”按钮将关闭程序工作区;如果单击“是”按钮则将同时关闭源程序窗口。

1.4 Visual C++ 6.0 程序调试

VC 6.0 设计了一个非常方便的程序调试环境,在调试程序时可以选择调试命令或工具对程序进行调试。下面介绍几种调试程序的方法。

1. 程序执行过程中暂停执行,以便观察中间结果

方法 1: 使程序执行到光标所在的那一行暂停。在需暂停的行上单击鼠标,定位光标。然后选择“组建”|“开始调试”|Run to Cursor 命令或按 Ctrl+F10 键,程序在执行到光标所在行后会暂停。如果把光标移动到后面的某个位置,再按 Ctrl+F10 键,程序将从当前的暂停点继续执行到新的光标位置,第二次暂停,如图 1-8 所示。

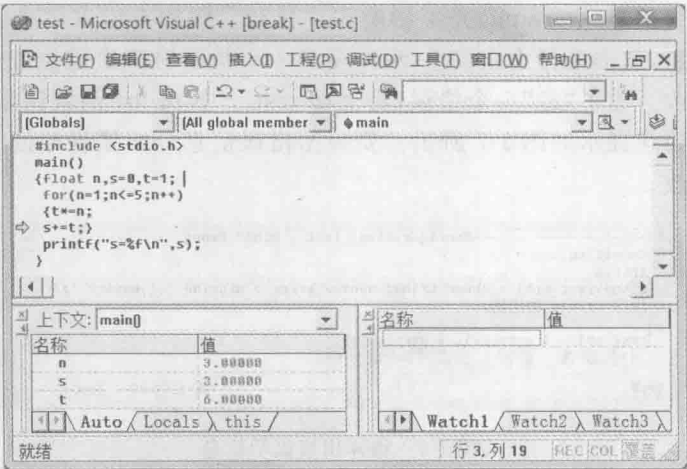


图 1-8 执行到光标所在行暂停

方法 2: 在需暂停的行上设置断点。在需设置断点的行上单击鼠标,定位光标,然后单击“编译微型条”工具栏或“组建”工具栏中最右面的 Insert/Remove Breakpoint 按钮或按 F9 键,如图 1-9 所示。被设置了断点的行前面会有一个红色圆点标志。



图 1-9 设置断点

不管是通过光标位置还是断点设置,其所在的程序行必须是程序执行的必经路线,亦即不应该是选择结构中的语句,因为该语句在程序执行中受到条件判断的限制,有可能因条件的不满足而不被执行,这时程序将一直执行到结束或下一个断点才停止。

一旦设置了断点,不管是否还需要调试程序,每次执行程序都会在断点上暂停,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“编译微型条”工具栏或“组建”工具栏中最右面的 Insert/Remove Breakpoint 按钮或按 F9 键,该按钮是一个开关按钮,按一次是设置,按两次是取消设置。如果有多个断点想全部取消,可选择“编辑”菜单中的“断点”命令,屏幕上会显示 Breakpoints 对话框,该对话框下方列出了所有断点,单击“全部移除”按钮即可取消所有断点。

断点通常用于调试较长的程序,可以避免使用 Run to Cursor(快捷键是 Ctrl+F10,运行程序到光标处暂停)功能时,经常要把光标定位到不同的地方。而对于程序行数较多时,要寻找某位置并不太方便。

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

2. 设置需观察的结果变量

按照上面的操作,使程序执行到指定位置时暂停,目的是查看有关的中间结果。在图 1-8 中,左下角窗口中系统自动显示了有关变量的值,其中 n,s,t 的值分别是 3,3,6。图中左侧的箭头表示当前程序暂停的位置。如果还想增加观察变量,可在图中右边的“名称”文本框中输入相应变量名。

3. 单步执行

当程序执行到某个位置时发现结果不正确,说明在此之前肯定有错误存在。如果能确定一小段程序可能有错,先按上面步骤暂停在该小段程序的头一行,再输入若干个查看变量,然后单步执行,即一次执行一行语句,逐行检查下来,看看到底是哪一行造成结果出现错误,从而确定错误的语句并加以改正。

设置光标暂停或断点后,选择“组建”|“开始调试”|Go 命令或按 F5 键,进入调试状态,“组建”菜单项变成“调试”菜单项。如果要单步执行,选择“调试”|Step Over 命令或按 F10 键。如果遇到自定义函数调用,想进入函数进行单步执行,可单击 Step Into 按钮或按 F11 键。如果想结束函数的单步执行,可单击 Step Out 按钮或按 Shift+F11 键。对不是函数调用的语句来说,F11 键与 F10 键的作用相同。

4. 停止调试

选择“调试”菜单中的 Stop Debugging 命令或按 Shift+F5 键可以结束调试,回到正常的运行状态。

第2章

实验指导

学习程序设计,上机实验是十分重要的环节。为了方便读者上机练习,本章设计了14个实验。这些实验和课堂教学紧密配合,读者可以根据实际情况从每个实验中选择部分内容作为上机练习,各个实验后面的实验思考题可以作为实验内容的补充。

为了达到理想的实验效果,读者务必做到:

(1) 实验前认真准备,要根据实验目的和实验内容,复习好实验中要用到的概念、语句,想好编程的思路,做到胸有成竹,提高上机效率。

(2) 实验过程中积极思考,要深入分析实验现象、程序的执行结果以及各种屏幕信息的含义、出现的原因并提出解决办法。

(3) 实验后认真总结,要总结本次实验有哪些收获,还存在哪些问题,并写出实验报告。实验报告应包括实验目的、实验内容、流程图、程序清单、运行结果以及实验的收获与体会等内容。

实验1 程序的运行环境和方法

一、实验目的

1. 熟悉 VC 6.0 集成开发环境的使用方法。
2. 熟悉 C 语言程序从编辑、编译、连接到运行并得到运行结果的过程。
3. 掌握 C 语言程序的结构特征与书写规则。

二、模仿编程实验

1. 输入下列程序,练习 VC 6.0 下程序的编辑、编译、连接和运行。

```
#include <stdio.h>
main()
{
    printf("This is the first C program!\n");
}
```

2. 计算并输出 π^2 。请补充程序,并上机运行该程序。

```
#include <stdio.h>
main()
{
    double p;
```