



高等学校计算机基础课程教材

*Exercises and
Computer Practice*

C语言程序设计

习题与上机实践

主 编 楼 静

副主编 方娇莉 付湘琼



高等教育出版社
HIGHER EDUCATION PRESS

高等学校计算机基础课程教材

C 语言程序设计习题与上机实践

C Yuyan Chengxu Sheji Xiti yu Shangji Shijian

主 编 楼 静

副主编 方娇莉 付湘琼

参 编 普运伟 刘领兵 郭 玲 潘晟旻 马晓静
罗一丹 王 凌 胡 钰 耿植林 黎 志



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书是方娇莉主编《C 语言程序设计》教材的配套教材,也可以单独作为“C 语言程序设计”课程的上机实验指导书或课后学习参考书。

本书包含三个部分的内容。第一部分 C 程序的建立及调试介绍 VisualC++6.0 开发环境及程序调试方法,第二部分习题与上机实验包含知识要点、习题和实验,第三部分是综合设计范例。本书能帮助学生巩固和掌握所学知识,使学生能在有限的上机学时内有目的地、高效地完成实验,提高阅读程序的能力,进一步理解和掌握一些基本的和常用的算法,培养逻辑思维和程序设计能力。

图书在版编目(CIP)数据

C 语言程序设计习题与上机实践/楼静主编. -- 北京:高等教育出版社,2013.2

ISBN 978-7-04-036923-6

I. ①C… II. ①楼 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(2013)第 022349 号

策划编辑 何新权
责任校对 张小镝

责任编辑 何新权
责任印制 张福涛

封面设计 赵 阳

版式设计 于 婕

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100120
印 刷 北京市鑫霸印务有限公司
开 本 787mm×1092mm 1/16
印 张 13
字 数 310 千字
购书热线 010-58581118

咨询电话 400-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landracom.com>
<http://www.landracom.com.cn>
版 次 2013 年 2 月第 1 版
印 次 2013 年 2 月第 1 次印刷
定 价 26.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换

版权所有 侵权必究

物 料 号 36923-00

前 言

“程序设计”是高等学校非计算机专业计算机教育的主要课程,程序设计能力是开展学科研究不可缺少的基础技能之一。学习“程序设计”的目的是学习计算机分析和解决问题的基本过程和思路,培养面向典型计算环境的问题求解方法,掌握语言级的问题求解技术,能使用计算机解决专业领域中的问题。

本教材所用的程序设计语言是 C 语言。使用 C 语言解决某一实际问题的过程包括分析问题、确定数据结构、建立算法、编写程序、调试程序和执行程序等多个阶段。为了使學生能顺利、有效地进行问题求解,本书从以下几个方面加以落实:

1. 针对“教学内容”列出知识要点,以强化学生对重点知识的认识和理解。

2. 针对“语言知识”组织和编写习题,以帮助学生理解基本概念和语法规则,培养学生快速、正确地阅读程序的方法和能力,掌握常用算法及其程序的编写,达到巩固所学知识、提高学习效率和学习效果的目的。

3. 针对“问题求解”组织和编写应用性较强的实验内容,以激发学生的学习兴趣,引导并促进学生理解分析问题和解决问题的基本过程和思路,加强基于算法的逻辑思维训练,提高编写程序的能力和效率。

4. 针对“知识应用”编写综合设计范例,使学生了解开发一个应用程序的过程包含分析问题的需求、划分功能模块、选择适当的数据结构和算法、编写程序、调试程序和执行程序等阶段,从而进一步理解结构化程序设计的思想,掌握程序设计的基本方法,真正将所学知识转化成解决问题的能力。

5. 针对“程序调试”编写能展示查找错误和纠正错误的方法及过程的实例,使学生学会阅读出错信息,分析出错原因,查找出错位置,掌握调试程序的方法,改正程序中的语法错误和逻辑错误,更深层次地理解程序测试和程序执行过程。

本书分为三个部分。第一部分由付湘琼编写;第二部分第 1 章由普运伟编写,第 2 章由刘领兵编写,第 3 章由楼静编写,第 4 章由方娇莉编写,第 5 章由郭玲编写,第 6 章和第 10 章由潘晟旻编写,第 7 章由马晓静编写,第 8 章由罗一丹编写,第 9 章由王凌编写,第 11 章由耿植林编写,第 12 章由胡钰编写;第三部分由黎志编写。全书由楼静任主编并负责统稿和审稿。

本书得到昆明理工大学教务处及部门同仁的大力支持和帮助,在此表示感谢!同时对本书所有参考文献、资料的作者表示感谢!

由于作者水平有限,书中难免存在不足之处,恳请各位专家、各位老师及广大读者给予批评指正,谢谢!

编 者

2013 年 1 月

目 录

第一部分 C 程序的建立及调试 1

第 1 章 Visual C++ 6.0 开发环境 ... 1

- 1.1 开发环境简介 1
- 1.2 菜单命令 3
- 1.3 工具栏 8
- 1.4 一个 C 程序的建立及运行 ... 9
- 1.5 一个项目的建立及运行 ... 12
- 1.6 帮助 15

第 2 章 程序调试 16

- 2.1 语法错误及纠正 16
- 2.2 逻辑错误及调试 21
- 2.3 程序调试实例 25

第二部分 习题与上机实验 30

第 1 章 认识 C 语言 30

- 一、知识要点 30
- 二、习题 31
- 三、实验 37
- 四、习题答案 40

第 2 章 顺序结构 41

- 一、知识要点 41
- 二、习题 42
- 三、实验 48
- 四、习题答案 52

第 3 章 选择结构 53

- 一、知识要点 53
- 二、习题 54
- 三、实验 62
- 四、习题答案 66

第 4 章 循环结构 67

- 一、知识要点 67

- 二、习题 68

- 三、实验 77

- 四、习题答案 80

第 5 章 数组 81

- 一、知识要点 81

- 二、习题 81

- 三、实验 88

- 四、习题答案 90

第 6 章 函数 91

- 一、知识要点 91

- 二、习题 92

- 三、实验 98

- 四、习题答案 101

第 7 章 自定义数据类型 102

- 一、知识要点 102

- 二、习题 103

- 三、实验 106

- 四、习题答案 110

第 8 章 指针 111

- 一、知识要点 111

- 二、习题 112

- 三、实验 119

- 四、习题答案 123

第 9 章 文件 124

- 一、知识要点 124

- 二、习题 125

- 三、实验 133

- 四、习题答案 136

第 10 章 位运算 137

- 一、知识要点 137

- 二、习题 138

- 三、实验 141

四、习题答案	145	一、系统概述	179
第 11 章 面向对象的程序设计	146	二、功能模块	179
一、知识要点	146	三、问题分析	179
二、习题	147	四、函数设计	180
三、实验	157	五、程序清单	180
四、习题答案	164	六、程序操作说明	186
第 12 章 程序设计基础知识	165	范例 2——学生信息管理系统	187
一、知识要点	165	一、系统概述	187
二、习题	166	二、功能模块	187
三、实验	171	三、数据结构及说明	187
四、习题答案	176	四、详细设计	187
第三部分 综合设计范例	179	五、程序清单	188
范例 1——日历系统	179	六、程序操作说明	196
		参考文献	198

第一部分 C 程序的建立及调试

第 1 章 Visual C++ 6.0 开发环境

Visual C++ 6.0 是功能强大的可视化集成开发环境,集代码编辑、编译、连接和调试等功能于一身,是 Windows 平台下专业程序员进行软件开发的首选工具。Microsoft 公司于 1993 年推出 Visual C++ 1.0 后,新版本不断推出,如 Visual C++ 6.0、Visual C++ 7.0、Visual C++ 2010 等。由于后续版本主要在 .Net 环境中应用,有一定的局限性,所以 6.0 版本仍是使用较多的版本。本章介绍 Visual C++ 6.0 集成开发环境的使用方法及 C 程序的建立和调试方法。

1.1 开发环境简介

在 Windows 操作系统下正确安装 Visual C++ 6.0 后,执行“开始”菜单中“程序”下“Microsoft Visual C++ 6.0”中的 Microsoft Visual C++ 6.0 程序,或双击桌面快捷图标,即可运行并进入 Visual C++ 6.0 集成开发环境,如图 1-1-1 所示。

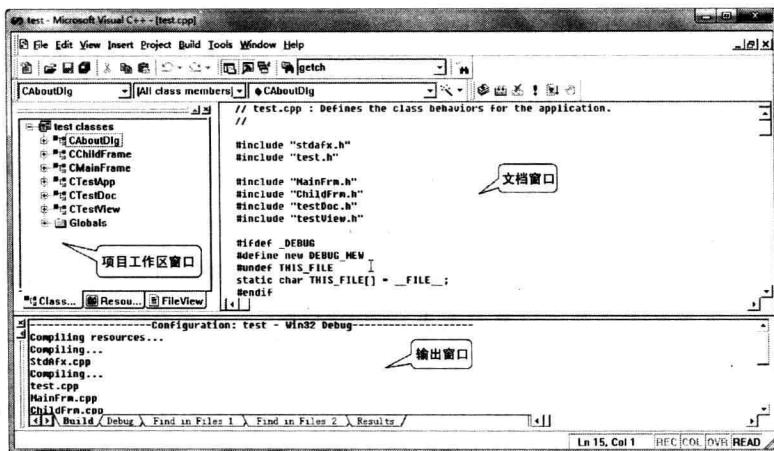


图 1-1-1 Visual C++ 6.0 集成开发环境

Visual C++ 6.0 集成开发环境中除标题栏、菜单栏、工具栏及状态栏外,还有项目工作区窗口、文档窗口、输出窗口等。

(1) 标题栏。标题栏的左端显示当前文档窗口中文档的文件名,最右端有最小化、最大化或还原及关闭按钮,单击关闭按钮可退出开发环境。

(2) 菜单栏。菜单栏中包含了开发环境中几乎所有的命令,为用户提供了文档操作、程序编译、调试、窗口操作等一系列功能。

(3) 工具栏。工具栏上是一些常用命令的按钮,其功能与相应菜单命令相同。

(4) 状态栏。状态栏一般位于开发环境的最底部,用来显示当前操作状态、注释、文本光标所在的行、列号等信息。

(5) 项目工作区窗口。项目工作区窗口中包含三个选项卡:ClassView、ResourceView 和 File-View。使用选项卡能方便地用不同的方式管理项目中的各个部分。

① ClassView(类视图):按类的方式定位和编辑源代码。

② ResourceView(资源视图):对项目中所需的各种资源进行管理和编辑,如对话框、图标和菜单等。

③ FileView(文件视图):按项目中组成文件的方式来定位和编辑源代码。

在项目工作区窗口中的任何标题或图标处单击鼠标右键,都会弹出相应的快捷菜单,其中包含当前状态下的一些常用操作。

(6) 文档窗口。一般位于开发环境窗口的右边,是编辑、修改程序代码的区域。各种程序代码的源文件、资源文件、文档文件等都可以在该窗口中显示和进行编辑。这个区域会根据编辑对象的不同而发生变化:当编辑源代码时为普通的文档编辑,在编辑对话框时会出现窗口设计器,在编辑图标时将显示图标绘制器。

(7) 输出窗口。一般位于开发环境窗口的底部,用于显示编译、链接、调试、查找时输出相关信息。例如,可以看到编译程序的进展情况、警告及出错信息,在调试代码时可以显示变量当前值等。

第一次运行 Visual C++ 6.0 时,屏幕上会显示“Tip of the Day(每日提示)”对话框,如图 1-1-2 所示。单击“Next Tip”按钮可看到有关各种操作的提示,单击“Close”按钮将关闭对话框。如果不希望每次启动时都出现该对话框,则去掉“Show tips at startup”复选框中的勾。

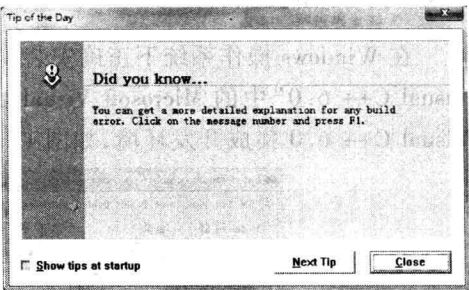


图 1-1-2 每日提示对话框

要在 Visual C++ 6.0 集成开发环境中进行快速操作,可使用表 1-1-1 所示的快捷键。

表 1-1-1 常用快捷键

快捷键	含义	快捷键	含义
Ctrl+ O	打开文件	Ctrl+ F	编辑查找
Ctrl+ S	保存文件	Ctrl+ Z	编辑撤销
Ctrl+ C	编辑复制	Ctrl+ Y	编辑重做
Ctrl+ X	编辑剪切	Ctrl+ A	编辑全选
Ctrl+ V	编辑粘贴	Ctrl+ F6	下一个窗口

续表

快捷键	含义	快捷键	含义
F4	下一个错误	Ctrl+F5	运行 (Execute)
Shift+F4	上一个错误	Shift+F5	退出调试
Alt+Enter	对象属性	F9	设置/取消断点
F7	建立可执行文件	F10	调试单语句执行
F5	以调试方式运行	F11	调试单指令执行

1.2 菜单命令

Visual C++ 6.0 的菜单栏中包含 File(文件)、Edit(编辑)、View(查看)、Insert(插入)、Project(工程)、Build(组建)、Tools(工具)、Windows(窗口)和 Help(帮助)9 个菜单选项。每个菜单选项下又包含一系列的菜单命令,通过执行这些命令可以方便地完成程序编制的各项任务。

(1) “File”菜单。用于对文件和项目进行操作,其中各命令的功能如表 1-1-2 所示。

表 1-1-2 File 的菜单项

菜单项	功能描述
New	新建文件或工程
Open	打开一个文件
Close	关闭当前文件
Open Workspace	打开一个工作空间,工作空间可以包含多个项目
Save Workspace	保存当前的工作空间
Close Workspace	关闭当前的工作空间
Save	保存当前的文件
Save As	将当前的文件另存为其他名称
Save All	保存所有文件
Page Setup	打印时的页面设置
Print	打印文件
Recent Files	显示最近编辑过的文件,以便快速打开
Recent Workspace	显示最近使用过的工作空间,以便快速打开
Exit	退出

(2) “Edit”菜单。用于对文件内容进行编辑、修改操作,其中各命令的功能如表 1-1-3 所示。

表 1-1-3 Edit 的菜单项

菜单项	功能描述
Undo	恢复上次操作
Redo	重复上次操作
Cut	剪切选中文本
Copy	复制选中文本
Paste	粘贴选中文本
Delete	删除选中文本
Select All	选中文件的全部内容
Find	在当前文件中查找字符串
Find in Files	在某个目录中的所有文件中查找字符串
Replace	在当前文件中查找并替换字符串
Go to	定位到文件中某一地方
Bookmarks	设置、管理书签
Advanced > Incremental Search	渐增型搜索
Advanced > Format Selection	将选中的文本转换成标准格式
Advanced > Tabify Selection	将选中的文本中的 4 个空格转换成 Tab
Advanced > Untabify Selection	将选中的文本中的 Tab 转换成 4 个空格
Advanced > Make Selection Uppercase	将选中的文本全部转换成大写字母
Advanced > Make Selection Lowercase	将选中的文本全部转换成小写字母
Advanced > View Whitespace	查看文件中的空格和制表符(Tab),空格用`来显示,制表符用^显示
Breakpoints	设置、管理断点
List Members	列出类的成员
Type Info	列出类的类型信息
Parameter Info	列出类的参数信息
Complete Word	自动补全代码

(3) “View”菜单。用于改变窗口的显示方式,以及调出“ClassWizard(建立类向导)”、“Debug(调试)”各个窗口等,其中各命令的功能如表 1-1-4 所示。

表 1-1-4 View 的菜单项

菜单项	功能描述
ClassWizard	启动类设计向导
Resource Symbols	查看资源符号

续表

菜单项	功能描述
Resource Includes	查看资源头文件
Full Screen	全屏幕方式
Workspace	显示 Workspace 窗口
Output	显示输出窗口
Debug Windows > Watch	显示 Watch 窗口,在调试时 Watch 窗口默认为打开
Debug Windows > Call Stack	显示调用堆栈窗口
Debug Windows > Memory	显示存储器窗口
Debug Windows > Variables	显示变量窗口,在调试时变量窗口默认为打开
Debug Windows > Registers	显示寄存器窗口
Debug Windows > Disassembly	显示反汇编窗口
Refresh	刷新显示
Properties	查看当前对象的属性

(4) “Insert”菜单。用于项目中对象的创建和添加,其中各命令的功能如表 1-1-5 所示。

表 1-1-5 Insert 的菜单项

菜单项	功能描述
New Class	添加一个新类
New Form	添加一个新的对话框类,基类只能在 CDialog、CFormView 和 CRecordView 之间选择
Resource	添加一个新的资源
Resource Copy	复制当前选中的资源,并添加进工程中
File As Text	将一个文件以文本形式添加进当前工程
New ATL Object	插入一个新的 ATL 对象

(5) “Project”菜单。用于管理项目和工作区,如向项目中添加文件,改变编译器和链接器选项等,其中各命令的功能如表 1-1-6 所示。

表 1-1-6 Project 的菜单项

菜单项	功能描述
Set Active Project	设置当前活动工程,仅在同时打开多个工程时有用
Add To Project > New	创建新的文件或工程并添加到当前工作空间中
Add To Project > New Folder	添加新的文件夹到当前工程中
Add To Project > Files	添加文件到当前工程中
Add To Project > Data Connection	添加数据库连接到当前工程中,只对数据库程序有用

续表

菜单项	功能描述
Add To Project > Components and Controls	添加 Visual C++提供的组件或已经登记的 ActiveX 控件到当前工程中
Dependencies	设置工程间的依赖关系
Settings	设置工程的各个属性选项
Export Makefile	将工程文件导出成 Make 文件
Insert Project into Workspace	向当前的工程工作空间中插入工程

(6) “Build”菜单。用于应用程序的编译、链接、调试、运行,其中各命令的功能如表 1-1-7 所示。

表 1-1-7 Build 的菜单项

菜单项	功能描述
Compile	编译当前激活的源文件或头文件
Build	编译并链接当前激活的工程配置,生成 .exe 文件
Rebuild All	对当前激活的工程配置重新进行编译链接
Batch Build	可以同时选中多个工程来进行编译链接
Clean	将当前激活的工程配置产生的所有中间文件和输出文件删除
Start Debug > Go	执行程序且遇到第一个断点停止,用于启动一次调试
Start Debug > Step Into	单步执行程序,对于函数调用跟踪进入函数内部
Start Debug > Run to Cursor	从当前位置执行程序,执行到光标所在行停止
Start Debug > Attach to Process	将系统进程挂到 Visual C++调试器中进行调试
Debug Remote Connection	调试远程计算机上的程序
Execute	运行程序(正常状态,不进行调试)
Set Active Configuration	设置哪个工程配置被激活
Configurations	设置工程的各个配置,包括 Debug、Release 版本,还可以添加自己的配置
Profile	生成 Profile 文件

(7) “Tools”菜单。用于浏览程序符号,选择或定制集成开发环境中的一些实用工具,其中各命令的功能如表 1-1-8 所示。

表 1-1-8 Tools 的菜单项

菜单项	功能描述
Source Browser	调出 Browser 工具,该工具可辅助浏览源代码信息
Close Class Browser File	关闭 Browser 打开的浏览信息窗口

续表

菜单项	功能描述
Visual Component Manager	启动 Component Manager(组件管理)工具
Register Control	启动 Register Control(注册控制)工具
Error Lookup	启动 Error Lookup(错误查找)工具
ActiveX Control Test Container	启动 ActiveX Control Test Container 工具
OLE/COM Object Viewer	启动 OLE/COM Object Viewer 工具
Spy++	启动 Spy++工具
MFC Tracer	启动 MFC Tracer 工具
Customize	显示用以定制菜单条、工具条、Tools 菜单、快捷键等对话框
Options	显示对话框,用以设置 IDE 开发环境的一些选项
Macro	管理、编辑、运行宏
Record Quick Macro	将鼠标、键盘操作录制成宏
Play Quick Macro	播放录制的宏

(8) “Window”菜单。用于控制窗口属性的命令选项,其中各命令的功能如表 1-1-9 所示。

表 1-1-9 Window 的菜单项

菜单项	功能描述
New Window	复制当前激活的窗口
Split	分割当前窗口
Docking View	在停靠/悬浮间切换停靠窗口的状态
Close	关闭当前窗口
Close All	关闭所有打开的窗口
Next	切换到下一个窗口
Previous	切换到前一个窗口
Cascade	层叠放置所有打开的窗口
Tile Horizontally	水平平铺所有打开的窗口
Tile Vertically	垂直平铺所有打开的窗口
Windows	调出 Windows 对话框,显示所有打开的窗口的列表,以控制窗口的激活、关闭、排列等

(9) “Help”菜单。用于提供联机帮助信息,该菜单大部分功能必须安装了 MSDN 才能使用,其中各命令的功能如表 1-1-10 所示。

表 1-1-10 Help 的菜单项

菜单项	功能描述
Contents	显示 MSDN 的首页
Search	在 MSDN 中搜索
Index	在 MSDN 中查找某个关键字
Use Extension Help	使用扩展帮助
Keyboard Map	查看键盘快捷键分配
Tip of the Day	显示每日小技巧
Technical Support	微软的技术支持
Microsoft on the Web	连接微软的网站
About Visual C++	关于 Visual C++的版本与版权信息

注意:上述介绍的是一般情况下开发环境所显示的菜单命令。随着集成开发环境当前状态的改变,菜单栏以及菜单选项中的命令项也随之变化。例如,当打开对话框编辑器时,集成开发环境中自动增加了“布局”选项,而当文档窗口中没有任何源文件时,许多菜单命令都是灰色的,不能使用。

1.3 工具栏

工具栏是一种图形化的操作界面,由一系列的工具按钮组成。按钮分别对应某些菜单命令的功能,具有直观和快捷的特点。熟练使用工具栏能提高工作效率。

Visual C++ 6.0 包含十几种工具栏,但默认情况下只显示 3 种工具栏:“Standard(标准)”工具栏、“WizardBar(向导)”工具栏及“Build MiniBar(小型编连)”工具栏。

(1) “Standard”工具栏。如图 1-1-3 所示,其中有新建、保存、撤销、查找等按钮。

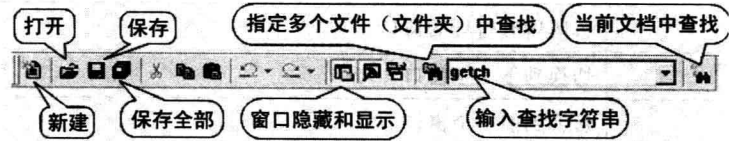


图 1-1-3 “Standard”工具栏

(2) “WizardBar”工具栏。如图 1-1-4 所示,由 3 个相互关联的组合框和 1 个 Actions 控制按钮组成,它是代码导航和类管理工具栏。

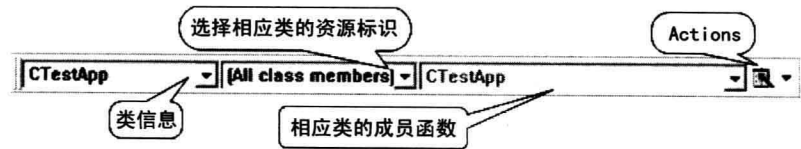


图 1-1-4 “WizardBar”工具栏

① “Class”组合框。用于选择想要找的函数所属的类。如果是全局函数,则选 Globals。

② “Filter”组合框。用于选择相应类的资源标识。如果一个类的成员函数较多,使用过滤条件可将无关函数隐藏起来。

③ “Members”组合框。用于选择相应类的成员函数。在该框中选择的函数会在主窗口中显示出来,且光标就定位在该函数的定义处。

④ “Actions”按钮。单击该按钮右侧的黑三角,可弹出快捷菜单。使用该菜单可将焦点快速定位到函数的定义处、声明处、类的定义处等。

当软件较大时,工程中常包含多个源文件及头文件,而且每个文件中又有多个类,每个类又有多个成员函数。该工具栏可将光标迅速定位到某个类或其成员函数的定义与声明处,也可以添加类,以及在类中添加、修改、删除成员。

(3) “Build MiniBar”工具栏。如图 1-1-5 所示,提供常用的编译、连接等操作命令。

在集成开发环境中,常根据不同的需要选择打开相应的工具栏,或隐藏那些暂时不用的工具栏。显示或隐藏工具栏可使用快捷菜单或“定制”对话框两种方式来实现。

用鼠标右击任何工具栏时会弹出如图 1-1-6 所示的快捷菜单,使用快捷菜单可快速地显示或隐藏所选的工具栏。

执行菜单“Tools”中的“Customize(定制)”命令,打开“Customize”对话框,如图 1-1-7 所示。然后在“Customize”对话框中单击“Toolbars(工具栏)”选项卡,在“Toolbars”框中选择想要打开的工具栏名称即可。

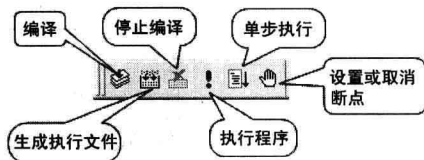


图 1-1-5 “Build MiniBar”工具栏

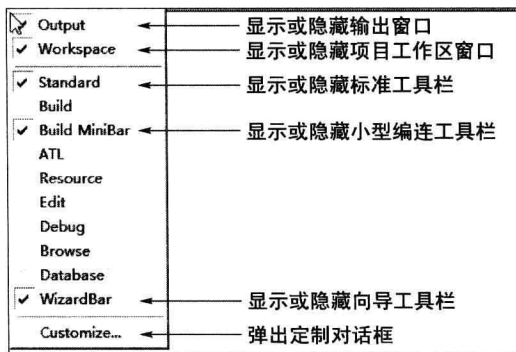


图 1-1-6 工具栏快捷菜单

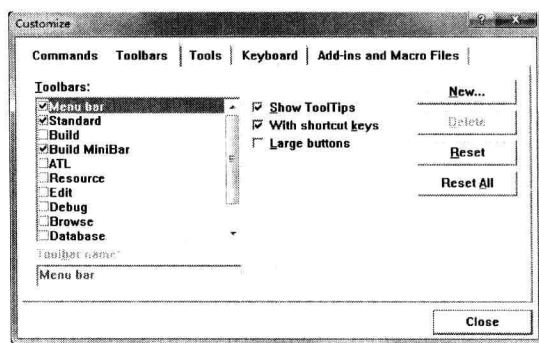


图 1-1-7 Customize 对话框

1.4 一个 C 程序的建立及运行

C 语言是一种编译型语言,用户编写好的源程序(扩展名为 .c 或 .cpp 的文件)不能直接执行,必须先经过编译程序将其编译成扩展名为 .obj 的目标程序,再经过连接程序将目标程序连接成扩展名为 .exe 的可执行程序后才能运行。

例如,在 Visual C++ 6.0 集成开发环境中建立一个 C 程序文件,并对其进行编译、连接及运行。操作步骤如下:

(1) 建立 C 源程序 ex-1. c

① 执行“File”菜单中的“New”命令,打开“New”对话框,如图 1-1-8 所示。

② 在“New”对话框中选择“Files”选项卡,在左边的列表框中选择“C++ Source File(C++源文件)”;在“File”输入框中输入程序名“ex-1. c”;在“Location”输入框中输入保存路径,或单击“...”按钮选择保存路径。

注意:输入文件名的同时也要输入文件扩展名. c,否则扩展名默认为. cpp。c 文件是 C 语言的源程序,. cpp 文件是 C++的源程序,两者是不同的。

③ 按图 1-1-8 所示设置完后,单击“OK”按钮关闭“New”对话框。

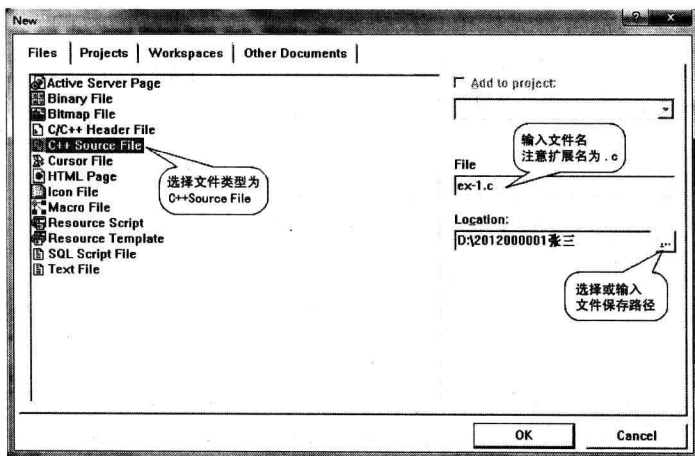


图 1-1-8 “New”对话框

④ 在文档窗口中输入以下程序注:

```
#include <stdio. h>
main( ) {
    double r=0, area=0;
    printf(" \n 请输入圆的半径:");
    scanf(" %lf", &r);
    area=3. 14159 * r * r;
    printf(" \n 输出圆的面积为:%lf\n", area);
}
```

⑤ 执行“File”菜单中的“Save”命令保存程序,文件名为“ex-1. c”。

(2) 编译 C 源程序 ex-1. c

① 执行“Build”菜单中的“Compile”命令,或者单击“Build MiniBar”工具栏上的编译按钮“”,将源程序 ex-1. c 编译成扩展名为. obj 的目标程序。

第一次编译源程序时,系统会弹出如图 1-1-9 所示的对话框,询问是否要建立一个默认的项目工作空间,这时必须单击“是”按钮,因为组建时需要一个项目工作空间。如果建立程序以

注:为与程序代码保持一致,书中的变量和数学公式全部采用正体形式。

前已经建立了项目工作区,就不会弹出该询问对话框。

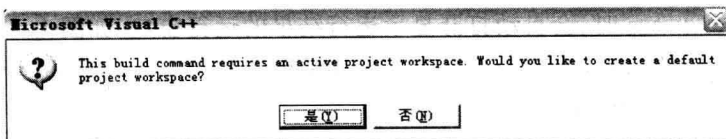


图 1-1-9 建立默认项目工作空间对话框

② 如图 1-1-10 所示,若在输出窗口中显示信息“ex-1.obj-0 error(s),0 warning(s)”,表示目标程序 ex-1.obj 已正确生成,可继续进行“连接”操作。

③ 若在输出窗口中显示信息“ex-1.obj-2 error(s),0 warning(s)”,表示源程序 ex-1.c 中有 2 个语法错误。在输出窗口中查看出错信息的内容,并按出错误信息提示改正源程序中的语法错误。

④ 编译改正后的源程序,若还有错误则再改正、再编译,如此反复,直到在输出窗口中显示信息“ex-1.obj-0 error(s),0 warning(s)”为止。

提示:语法错误分为两类,一类是致命错误,以 Error 表示;另一类是警告错误,以 Warning 表示,这类错误一般不影响生成目标程序,程序仍可以被成功编译、链接,但可能影响程序运行的结果,所以最好将其改正。程序中有 Error 类错误时,.c 源程序就不能编译生成正确的.obj 目标程序。



图 1-1-10 编译 C 源程序

(3) 链接目标程序 ex-1.obj

① 执行“Build”菜单中的“Build”命令,或者单击“Build MiniBar”工具栏上的组建按钮,将目标程序 ex-1.obj 链接成扩展名为.exe 的可执行程序。

② 若在输出窗口中显示信息“ex-1.exe-0 error(s),0 warning(s)”,表示 ex-1.exe 可执行程序已经正确生成,可继续进行“运行”操作。

③ 若不能正确生成可执行文件,则表示链接不成功,应根据输出窗口中显示的出错信息,修改源程序中的错误或修正源程序中的链接错误,然后再进行链接,如此反复直到没有错误为止。

(4) 运行可执行程序 ex-1.exe

① 执行“Build”菜单中的“Execute”命令,或者单击“Build MiniBar”工具栏上的运行按钮